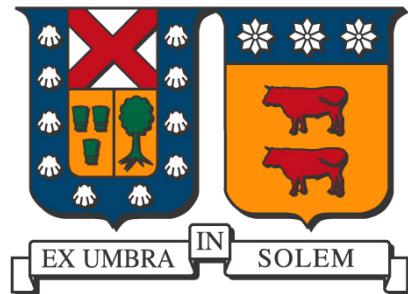


**UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA  
DEPARTAMENTO DE ELECTRÓNICA  
VALPARAÍSO - CHILE**



**“AUTOMATIZACIÓN DE UNA ‘TABLE FOOTBALL’  
MEDIANTE PROCESAMIENTO DE IMÁGENES Y  
ALGORITMOS NEURO-DIFUSOS”**

**RICHARD MICHAEL ATUAN TOLEDO**

**MEMORIA DE TITULACIÓN PARA OPTAR AL TÍTULO DE INGENIERO CIVIL  
ELECTRÓNICO**

**PROFESOR GUÍA:**

**FRANCISCO VARGAS**

**PROFESOR CO-REFERENTE:**

**GONZALO CARVAJAL**

**Agosto - 2018**

## *Agradecimientos*

A lo largo de mi vida, hay personas que han estado siempre allí, apoyando cada decisión que he tomado o cada situación que ha surgido de forma incondicional, Christian Atuan, Catalina Toledo y Jean Pierre Atuan, este trabajo va dedicado a ustedes y a toda mi familia, ya que de otra forma nada de esto hubiera sido posible.

De igual manera agradezco a mis compañeros y amigos que hicieron de este viaje un trayecto agradable, tanto durante el estudio, en celebraciones y momentos difíciles a lo largo de mi carrera universitaria y escolar.

Así también, a todas las personas que de alguna forma fueron parte de mi formación académica y valórica.

## *Resumen*

El estudio que se presenta a continuación, corresponde al primer paso de un proyecto que pretende automatizar el funcionamiento en uno de los equipos de una "Table Football", teniendo como motivación la creación de una plataforma de desarrollo experimental, donde se puedan implementar y probar algoritmos o tecnologías con fines académicos y pedagógicos. El presente trabajo buscar sistematizar, en un ambiente simulado, el movimiento de uno de los arqueros con el fin de bloquear los tiros rivales a portería.

En esta primera etapa del proyecto, se implementa un algoritmo computacional que logra, por medio de una simulación, estimar la posición del arquero dentro del juego. Mediante procesamiento de imágenes se obtiene el lugar donde se encuentra la pelota en juego, mientras que un algoritmo Neuro-Difuso (ANFIS) permite estimar dónde se dirige esta, estableciendo así, la posición donde el arquero debería ubicarse para bloquear el paso de la pelota. Además, se simula el movimiento del portero en cuestión mediante un motor DC cuya posición se controla a través de un controlador PID. Cabe destacar que al ser una primera etapa, la estructura física del proyecto se encuentra en desarrollo, por lo que se opta por un trabajo de simulación con datos teóricos (motor DC) y prácticos (Imágenes/Vídeos).

En este trabajo se plantean soluciones a problemas tales como, la obtención de imágenes y el reconocimiento de la pelota dentro de ellas, predicción de la posición futura del balón y el control del movimiento en un motor DC, especificando las consideraciones técnicas tanto para la obtención de imágenes como para su implementación. Finalmente, se integran dichas soluciones propuestas en un ambiente simulado desarrollado para estudiar el comportamiento del arquero. Esto permitió evaluar, vía simulación, la utilidad de los algoritmos propuestos para ser usados en el sistema real.

## *Abstract*

The study presented in this document, corresponds to the first step of a project that aims to automate the operation on one of the teams of a "Table Football", having as a motivation the creation of a platform for experimental development, where you can implement and test algorithms or technologies with academic and pedagogical purposes. This work seeks to systematize, in a simulated environment, the movement of the goalkeeper, in order to block rivals shots to the goal.

In this first stage of the project, a computational algorithm is implemented that achieves, through a simulation, estimate the position of the goalkeeper within the game. By image processing is obtained the place on the field where the ball is, while a Neuro-Diffuse algorithm (ANFIS) allows to know where the ball is going, establishing the position where the goalkeeper should be placed. In addition, the movement of the goalkeeper is simulated by a DC motor whose position is controlled by a PID controller. It should be noted that, as it is a first stage, the physical structure of the project is in development, so the study opted for a simulation work with theoretical data (DC motor) and practical data (Images / Videos).

In this work there are solutions to problems such as, image acquisition and identification of the ball inside these, prediction of the future position of the ball and control of movement in a DC motor are proposed, specifying the technical considerations both for obtaining images and for their implementation. Finally, these proposed solutions are integrated into a simulated environment developed to study the behavior of the goalkeeper. This allowed to evaluate, via simulation, the usefulness of the algorithms proposed to be used in the real system.

## *Glosario*

Se expresa en esa sección una definición de ciertos conceptos para evitar malentendidos sobre el uso de ciertas palabras:

Jugadores	Personas que se encuentran participando en el juego.
Barra	Estructura metálica alargada unida a los muñecos para permitir su movimiento dentro del juego.
Muñecos	Marionetas generalmente con forma humanoide para golpear a la pelota.
Oclusiones	Perdida del seguimiento de un objeto en una secuencia de imágenes, producto de otros objetos
Obturador	Mecanismo de la cámaras, para regular el paso de luz al sensor.
ANFIS	Adaptive Neuro Fuzzy Inference System (Sistema Adaptativo de Inferencia Neuro-Difusa).
FIS	Fuzzy Inference System (Sistema de Inferencia Difusa).
ANN	Artificial Neuronal Network (Redes Neuronales Artificiales).

## Índice general

1.. <i>Introducción al Problema de Estudio</i> . . . . .	10
1.1. Preliminares . . . . .	11
1.1.1. “Table Football” o Fútbol de mesa . . . . .	11
1.1.2. Dimensiones de la mesa de juego y distribución de los elementos . .	14
1.1.3. Trabajos previos . . . . .	15
1.2. Trabajo Desarrollado . . . . .	19
1.2.1. Objetivo General . . . . .	20
1.2.2. Objetivos Específicos . . . . .	20
2.. <i>Materiales y Métodos</i> . . . . .	22
2.1. Consideraciones acerca de la mesa de juego . . . . .	22
2.1.1. Superficie de la mesa . . . . .	22
2.1.2. Trayectorias y rebotes del balón . . . . .	23
2.1.3. Oclusiones . . . . .	23
2.1.4. Velocidad máxima del balón . . . . .	25
2.2. Sensor de vídeo y Medición . . . . .	27
2.2.1. Cámaras de vídeo y complementos . . . . .	27
2.2.2. Procesamiento Digital de imágenes . . . . .	43
2.3. Predicción . . . . .	54
2.3.1. Algoritmo Neuro-Difuso: ANFIS . . . . .	55
2.4. Actuación . . . . .	71
2.4.1. Motores DC . . . . .	71
2.4.2. Controlador PID . . . . .	78

<i>3.. Metodología</i>	83
3.1. Adquisición y Procesamiento	83
3.1.1. Adquisición de imágenes	84
3.1.2. Procesamiento de las imágenes	85
3.1.3. Seguimiento v/s resolución	88
3.2. Predicción de las condiciones de juego	88
3.2.1. Diseño de la Red ANFIS	90
3.3. Modelado del motor DC y Diseño del controlador PID	93
3.4. Simulación	99
<i>4.. Análisis de Resultados</i>	100
4.1. Adquisición de imágenes	100
4.1.1. Efecto del lente	100
4.1.2. Velocidad de adquisición vs Intensidad	102
4.1.3. Limitaciones por ancho de banda	103
4.1.4. Ganancia ISO e Iluminación	105
4.1.5. Iluminación y sombras	113
4.2. Identificación de la posición de la pelota	114
4.2.1. Comportamiento de la detección según la resolución de la imagen	121
4.3. Predicción de las condiciones de juego	125
4.3.1. Variación del número de entradas	126
4.3.2. Variación del número de reglas	126
4.3.3. Variación del número de datos de entrenamiento	127
4.3.4. Variación del rango en las funciones de membresía	128
4.3.5. Evaluación del error antes y después del entrenamiento	130
4.3.6. Estructura de la red ANFIS	132
4.4. Motor DC y Control de posición	133
4.5. Simulación	140
<i>5.. Conclusiones</i>	147

6.. <i>Posibles Mejoras</i> . . . . .	149
7.. <i>ANEXO A: Implementación del trabajo desarrollado</i> . . . . .	150

## Índice de figuras

1.1.	Mesa de Taca-Taca o “Table Football” . . . . .	10
1.2.	Logo: ITSF Copa del Mundo 2019 . . . . .	12
1.3.	Disposición de las barras . . . . .	12
1.4.	Barra y muñecos . . . . .	13
1.5.	Dimensiones de la cancha . . . . .	14
1.6.	Orden de los muñecos. Equipo A en verde, B en amarillo . . . . .	15
1.7.	Matriz láser . . . . .	16
1.8.	Cámara de vista aérea propuesto por “Brigham Young University” . . . . .	17
1.9.	Cámara posicionada debajo de la mesa propuesto por “Ecole Polytechnique Federale” . . . . .	17
1.10.	Utilización de máscara sobre imagen . . . . .	18
1.11.	Flujo de la información dentro del sistema . . . . .	20
2.1.	Tipos de occlusiones . . . . .	24
2.2.	Pérdida del seguimiento de la pelota por occlusión . . . . .	25
2.3.	Vista superior. A la izquierda la base de un muñeco. A la derecha el balón . .	26
2.4.	Representación de tiros, a la izquierda tiro al frente y a la derecha tiro con ángulo. . . . .	26
2.5.	Cuadros para medir velocidad máxima. Tiempo cuadro a cuadro: 4.16 ms. . .	27
2.6.	Círculo representativo del efecto fotoeléctrico . . . . .	29
2.7.	Sombras de las barras y muñecos sobre el campo de juego. . . . .	30
2.8.	Diafragma de una cámara de vídeo . . . . .	31
2.9.	Valores posibles del Diafragma en la escala f . . . . .	31

2.10. Tiempo de exposición alto . . . . .	32
2.11. Tiempo de exposición bajo . . . . .	33
2.12. Obturador rotatorio . . . . .	34
2.13. Representación de un obturador rotatorio . . . . .	35
2.14. Diversos efectos en imágenes causados por un obturador rotatorio . . . . .	36
2.15. Diferencias al realizar un ajuste del ISO . . . . .	37
2.16. Lentes Basler . . . . .	38
2.17. Foto con lente Ojo de pez . . . . .	39
2.18. Distancias focales . . . . .	40
2.19. Tamaño de sensores en cámaras . . . . .	41
2.20. Píxeles en una pantalla de televisor . . . . .	44
2.21. Valores de píxeles en una imagen en escala de grises . . . . .	45
2.22. Algunos tipos de espacios de colores . . . . .	46
2.23. Comparación entre espacios de colores y pérdida de información . . . . .	46
2.24. Algunos tipos de kernel . . . . .	47
2.25. Aplicación de filtro sobre imagen y generación de una nueva . . . . .	48
2.26. Tipos de ruido en imágenes . . . . .	49
2.27. Uso del algoritmo Canny . . . . .	51
2.28. Distorsión producto de la curvatura en lentes . . . . .	54
2.29. Tipos de funciones de membresía . . . . .	56
2.30. Red Neuronal y sus capas . . . . .	59
2.31. Algunas funciones de activación de las Redes Neuronales . . . . .	60
2.32. Learning rate: Iteraciones hasta llegar al mínimo error posible . . . . .	61
2.33. Estructura de un sistema ANFIS Takagi-Sugeno . . . . .	62
2.34. Método de “Grid Partition” . . . . .	65
2.35. Método “SubClustering” . . . . .	66
2.36. Motor DC con escobillas . . . . .	72
2.37. Rotor y estator de un motor DC con escobillas . . . . .	73
2.38. Sistema de un motor DC con escobillas . . . . .	74

2.39. Motor DC sin escobillas . . . . .	75
2.40. Modelo de un motor DC con escobillas con circuito de campo . . . . .	75
2.41. Modelo de un motor DC expresado en frecuencia . . . . .	77
2.42. Control de proceso SISO mediante un lazo de realimentación . . . . .	78
2.43. Modelo de un control PID . . . . .	79
2.44. Lazo de control de un proceso . . . . .	81
2.45. Lugar de raíces de la función de transferencia $H(s)$ . . . . .	82
 3.1. Modelo con carga . . . . .	94
3.2. Modelo en bloques del motor DC . . . . .	95
3.3. Modelo con lazo de realimentación . . . . .	95
3.4. Respuesta a escalón . . . . .	99
 4.1. Comparación en la distorsión de imagen producto del lente. Las medidas de cada cuadro corresponden a $10 \times 10 [\text{cm}^2]$ , siendo en total $1.0 \times 0.7 [\text{m}^2]$ . . . . .	101
4.2. Corrección del efecto barril . . . . .	102
4.3. Comportamiento de la intensidad de los píxeles al variar la velocidad de adquisición con resolución de $800 \times 600$ píxeles . . . . .	103
4.4. Comportamiento de la intensidad de los píxeles al variar la velocidad de adquisición con resolución de $600 \times 400$ píxeles . . . . .	105
4.5. Comportamiento del ruido al variar la ganancia del sensor a una tasa de 400 fps con resolución de $800 \times 600$ píxeles . . . . .	106
4.6. Cuadrado utilizado para medir ruido en las imágenes . . . . .	107
4.7. Ruido presente en el canal de Matiz de acuerdo a cada imagen . . . . .	108
4.8. Comportamiento del ruido al variar la ganancia del sensor a una tasa de 400 fps con resolución de $800 \times 600$ píxeles, con fondo blanco . . . . .	110
4.9. Cuadrado utilizado para medir ruido en las imágenes . . . . .	111
4.10. Ruido presente en el canal de Matiz de acuerdo a cada imagen con fondo blanco	112
4.11. Efecto del posicionamiento de las luces . . . . .	113
4.12. Imagen de referencia . . . . .	114

4.13. Canales HSV de la imagen . . . . .	115
4.14. Imagen sin filtrar . . . . .	116
4.15. Comparación de filtro con umbrales 3, y 8. . . . .	118
4.16. Posición del objeto (punto rojo en la imagen sobre la pelota verde) . . . . .	119
4.17. Detección color verde . . . . .	120
4.18. Detección color amarillo . . . . .	120
4.19. Detección color azul . . . . .	121
4.20. Detección color rojo . . . . .	121
4.21. Comparación factores de escalados 1, 2 y 3 . . . . .	122
4.22. Comparación factores de escalados 1, 10 y 15 . . . . .	123
4.23. Comparación factores de escalados 1, 20 y 30 . . . . .	124
4.23. Modificación en las funciones de membresía . . . . .	130
4.24. Comparación de las funciones de membresía antes y después de entrenarse (con ruido) . . . . .	131
4.25. Estructura del sistema difuso . . . . .	133
4.26. Mediciones de velocidad y posición del motor . . . . .	134
4.27. Root Locus . . . . .	136
4.28. Respuesta a escalón . . . . .	137
4.29. Respuestas del motor con control PD . . . . .	140
4.30. Respuesta a escalón . . . . .	141
4.31. Instantes durante la simulación. Circulo rojo: Posición de la pelota. Asterisco azul: Referencia entregada por ANFIS, Circulo verde posición de arquero . . . . .	142
4.32. Seguimiento a referencia con tiro de 0.5 m/s. Lineas segmentadas indican el instante en que la pelota llega al extremo del arquero. . . . .	144
4.33. Error con tiro de 0.5 m/s. Lineas segmentadas indican el instante en que la pelota llega al extremo del arquero. . . . .	144
4.34. Seguimiento a referencia con tiro de 1 m/s. Lineas segmentadas indican el instante en que la pelota llega al extremo del arquero. . . . .	145

4.35. Error con tiro de 1 m/s. Lineas segmentadas indican el instante en que la pelota llega al extremo del arquero. . . . .	145
4.36. Seguimiento a referencia con tiro de 5 m/s. Lineas segmentadas indican el instante en que la pelota llega al extremo del arquero. . . . .	146
4.37. Error con tiro de 5 m/s. Lineas segmentadas indican el instante en que la pelota llega al extremo del arquero. . . . .	146
7.1. Tiempos de ejecución de los procesos . . . . .	152
7.2. Programa con ejecución Off-line . . . . .	153
7.3. Programa con ejecución On-line . . . . .	154
7.1. Ejecución del programa. Detección de pelota azul y respuesta de ANFIS en extremo izquierdo (en rojo) . . . . .	158

## Índice de cuadros

2.1. Tabla de lógica difusa . . . . .	58
2.2. Parámetros de un motor DC . . . . .	76
3.1. Proceso de filtrado de un píxel . . . . .	87
3.2. Parámetros y valores del motor DC . . . . .	93
4.1. Resultados de la medición de ruido en fondo negro . . . . .	109
4.2. Resultados de la medición de ruido en fondo blanco . . . . .	112
4.3. Rango de valores para identificación de objetos . . . . .	116
4.4. Comparación de seguimiento, variando la resolución de imagen . . . . .	125
4.5. Comportamiento del error variando el número de entradas . . . . .	126
4.6. Comparación del sistema según el número de reglas . . . . .	127
4.7. Error cuadrático medio para distintas cantidades de datos con y sin ruido, en sistema con 7 pares de entradas . . . . .	128
4.8. Error según la función de membresía . . . . .	129
4.9. Comparación antes y después de entrenarse . . . . .	131
7.1. Tiempos de procesamiento en CPU y GPU para obtención de posición de la pelota, por cuadro de vídeo. . . . .	151

## 1. INTRODUCCIÓN AL PROBLEMA DE ESTUDIO

El trabajo que se presenta a continuación corresponde a la primera etapa de un proyecto universitario, el cual consiste en la automatización de una “Table Football” popularmente conocida en Chile como mesa de Taca-Taca (figura 1.1). Este proyecto busca formar una base donde se pueda realizar experimentación, pruebas de algoritmos, integración de hardware y software, difusión al público y por su puesto, incentivar a las personas por medio de métodos didácticos a involucrarse en desarrollos tecnológicos.



*Fig. 1.1: Mesa de Taca-Taca o “Table Football”*

La secuencia de pasos para lograr automatizar la mesa de juego puede ser expresada en tres etapas distintas:

- Obtención de los datos del juego.

- Predicción de las condiciones y acciones necesarias.
- Actuación y movimiento acorde a las predicciones.

La primera etapa expresa adquirir la información del campo de juego como lo son las posición de los jugadores y de la pelota. La segunda etapa contempla lograr estimar o predecir cuales serán las condiciones futuras, como por ejemplo, a qué altura o posición se encontrará el balón cuando llegue al extremo del arquero. Y por último, mover a los muñecos tanto para evitar como lograr anotar goles al equipo rival.

## 1.1. *Preliminares*

En esta sección, se realiza una descripción detallada del juego “Table Football”, las dimensiones y distribución de sus elementos y por ultimo, se muestran trabajos que han sido realizados por otras instituciones que buscaban el mismo propósito, la automatización de este juego.

### 1.1.1. “Table Football” o Fútbol de mesa

“Table Football” o Fútbol de mesa, es un juego popular en el mundo. Inventada en 1921 en el Reino Unido, ha logrado perdurar hasta nuestros días aunque con distintos nombres dependiendo de la ubicación geográfica donde nos situemos: futbolito, metegol, fulbito, futmesa, taca-taca, fulbatin, tacatocó, futillo, fulbacho, tiragol, fulbote, “foosball”, “table soccer”, “kicker”, son algunos de los nombres que recibe.

Este juego se encuentra basado en el deporte del fútbol, buscando el enfrentamiento entre dos equipos los cuales deben anotar la mayor cantidad de goles en la portería contraria. Al igual que su juego original, este posee competencias internacionales como la Copa del Mundo (figura 1.2), donde la Federación Internacional de Fútbol de Mesa (ITSF [1]) se encarga

de establecer las reglas y normativas de cada competición.



*Fig. 1.2: Logo: ITSF Copa del Mundo 2019*

Dos grupos de competidores compuestos por 1 o 2 jugadores cada uno, cuentan con cuatro barras a su disposición (figura 1.3). Estas barras corresponden a la linea de arquero, defensa, medio campo y delanteros. Cada una de estas poseen adheridos muñecos de plástico, madera o metal de acuerdo al fabricante con los que ambos equipos intentarán golpear la pelota (figura 1.4). Los participantes pueden de esta forma, mover cada una de estas barras además de girarlas sobre su propio eje, el equipo que anote la mayor cantidad o bien una sean los primeros en anotar una determinada cantidad de goles al rival gana el juego.



*Fig. 1.3: Disposición de las barras*



Fig. 1.4: Barra y muñecos

Algunos tiros que se pueden dar durante el juego son:

- Tiro Aereo: Consiste en pasar la pelota por arriba del rival.
- Tiro tic-tac: Los muñecos de una misma barra se dan pases para después tirar a portería.
- Tiro a la banda: Se utilizan los límites físicos de la mesa de juego para hacer rebotar el balón.
- “Spinning”: Se hace girar la barra con la palma de la mano para golpear el balón con mayor fuerza.

Dependiendo de cada competencia, los tiros pueden ser legales o ilegales, como por ejemplo el “Spinning”, que resulta ilegal en ciertas competencias o bien está sujeto a reglas de uso dentro del juego, por ejemplo que sólo el arquero pueda utilizar este movimiento.

Existe una gran variedad de modificaciones a la mesa de juego, como lo son los materiales de fabricación, las porterías sobresalientes al terreno de juego o bien embutidas dentro de los límites de la cancha, la cantidad de muñecos por barra, muñecos con los pies juntos o separados (es la parte inferior de los muñecos con que golpean el balón en general), cantidad de barras, entre otras.

### 1.1.2. Dimensiones de la mesa de juego y distribución de los elementos

Las dimensiones de la mesa de juego (figura 1.5) pueden variar dependiendo del fabricante y competición, pero en competiciones profesionales (utilizadas para este estudio) se ha reglamentado que estas tendrán 1.2 [m] de largo y 0.6 [m] de ancho [2]. Las barras se encuentran espaciadas a lo largo de la cancha, distribuidas a 12 cm de cada una, estando la primera posicionada desde los 12 cm partiendo de un extremo, dando un total de 8 barras.

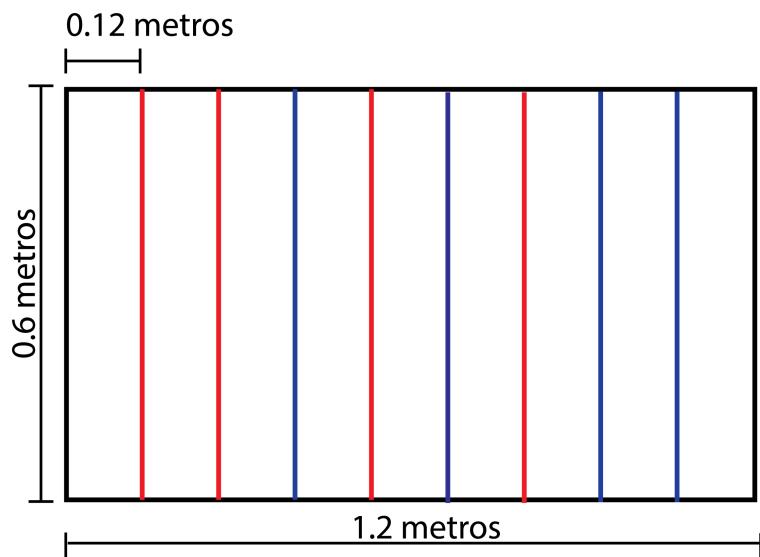
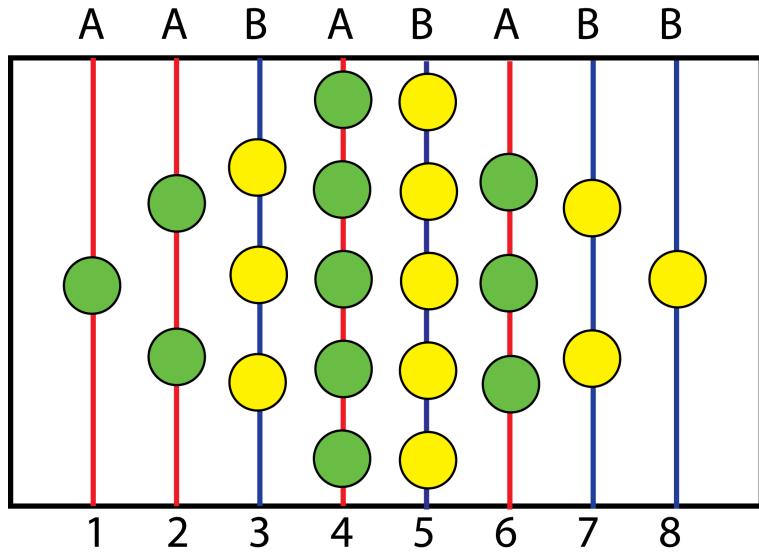


Fig. 1.5: Dimensiones de la cancha

Los equipos poseen un orden específico (figura 1.6), de izquierda a derecha tendremos la linea del arquero A, linea de defensa A, linea de delanteros B, linea de medio campo A, linea de medio campo B, linea de delanteros A, linea de defensas B y la linea de arquero B siendo A y B los correspondientes equipos. La cantidad de muñecos por barra puede variar, pero en competencias se utilizan 1-2-3-5-5-3-2-1 respectivamente.



*Fig. 1.6: Orden de los muñecos. Equipo A en verde, B en amarillo*

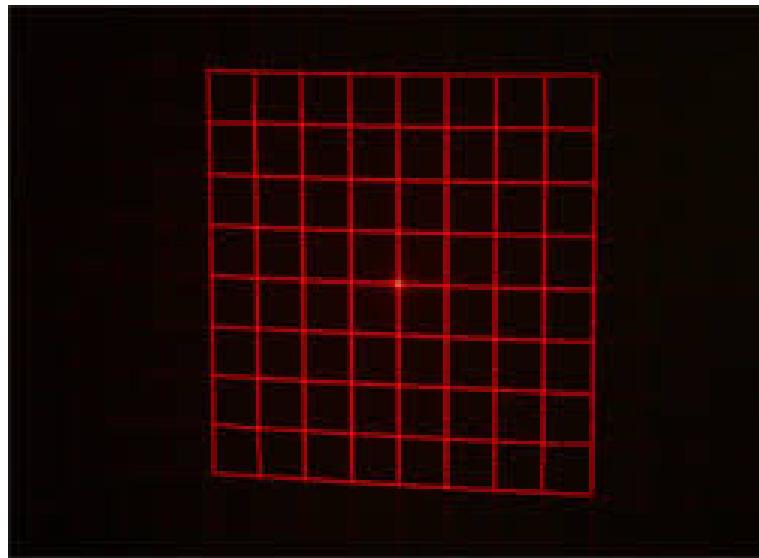
En cuanto a las características de la pelota, estas resultan ser de color naranja en competiciones y de 3 [cm] de diámetro.

#### *1.1.3. Trabajos previos*

En los últimos años se han desarrollado proyectos universitarios y comerciales en donde se ha automatizado este juego mediante distintos métodos, cada uno con ventajas y desventajas.

La Universidad de Adelaide [3] realizó una automatización considerando una matriz láser (figura 1.7) que pasaba por debajo de los muñecos sin tocarlos, con lo cual se permite detectar la posición de la pelota ante la interrupción del haz. La ventaja de este método es que resulta bastante intuitivo de procesar y medir, utilizando un sistema “ON-OFF” de detección se puede establecer la posición, siendo el estado “ON” cuando el haz de luz no está bloqueado y el “OFF” cuando se interrumpe. La desventaja directa es la cantidad de sensores, tanto emisores como receptores que se necesitan utilizar para obtener una buena resolución de la posición del balón, ya que existirá un límite físico del tamaño del sensor el cual no permitirá obtener

una mayor resolución, además existe el inconveniente de que al estar los sensores muy juntos, puede existir interferencias entre ellos.



*Fig. 1.7: Matriz láser*

“Brigham Young University” utilizó una cámara de vídeo la cual se encontraba posicionada encima de la mesa apuntando sobre el campo de juego (figura 1.8), además utilizó colores distintivos para todos los objetos con el fin de utilizar segmentación por color para detectar cada uno de ellos. La ventaja de esto es que permite obtener de una forma práctica y poco invasiva la información del juego. Las desventajas de utilizar una cámara es que ésta necesita tener una gran velocidad de adquisición ya que el balón puede desplazarse a altas velocidades, evitando obtener imágenes borrosas cuando el objeto a enfocar está en movimiento. Otra desventaja de utilizar este tipo de configuración, es la pérdida del seguimiento de la pelota cuando se pierde de vista al pasar por debajo de los objetos como las barras o muñecos.



Fig. 1.8: Cámara de vista aérea propuesto por “Brigham Young University”

StarKick [4] es un proyecto que se volvió comercial en donde la cámara de vídeo a color fue posicionada por debajo de la mesa apuntando hacia arriba, y el suelo o superficie del campo de juego fue reemplazado por un material transparente lo cual permitió eliminar la pérdida de seguimiento por obstaculizaciones. Un acercamiento similar fue realizado por “Ecole Polytechnique Federale” de Lausanne [5] (figura 1.9).

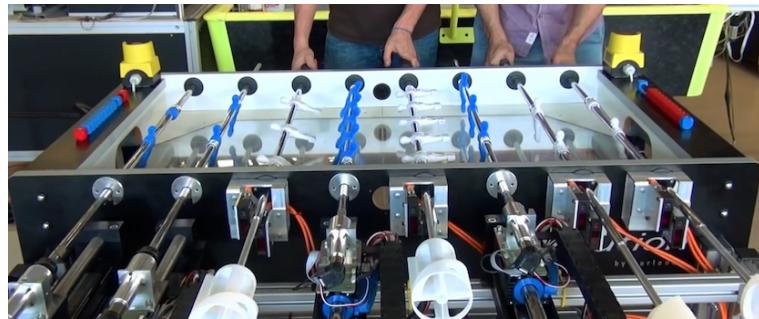


Fig. 1.9: Cámara posicionada debajo de la mesa propuesto por “Ecole Polytechnique Federale”

Existen propuestas de trabajo donde se utilizan cámaras de vídeo monocromáticas [6] es decir que capturan imágenes en escala de grises, lo cual es ventajoso en cuanto a carga computacional, ya que reduce el tiempo de procesamiento en un factor de tres, pasando de 24 bits por pixel a 8 bits, lo cual a su vez puede reducir a un tercio los tiempos de ejecución del sistema siendo un aspecto importante al buscar procesamiento en linea, pero esta mejora en tiempo puede verse perjudicada al disminuir considerablemente la cantidad de información

adquirida, donde los datos se vuelven más susceptibles al ruido ambiental y térmico, producido tanto por la contaminación lumínica como por la electrónica del sensor, lo que resulta en el aumento de procesamiento para obtener correctamente la posición de la pelota y los jugadores en caso de ser necesarios.

Una solución al uso de cámaras monocromáticas, es la utilización de una máscara (superposición de una figura sobre la imagen) (figuras 1.10a a 1.10c) que permita ocultar las barras y muñecos de la imagen adquirida evitando la pérdida de seguimiento [7], de esta forma se puede obtener un alto contraste del balón con la superficie de juego y obtener la posición de la pelota dentro del juego. La dificultad de este proceso radica en la obtención de la máscara, ya que a pesar de que las barras se encuentran fijas en su lugar al observarlas de arriba, los muñecos pueden desplazarse linealmente y rotar sobre su eje, lo que nos da como resultado una máscara que debe ser ajustada en el tiempo, siendo una solución el uso de sensores en las barras que entreguen el ángulo de rotación en que se encuentran y su posición lineal. Conociendo las dimensiones de los muñecos se pueden realizar modelos tridimensionales que al ser proyectados sobre el terreno de juego pueden actuar como máscaras en el procesamiento de imágenes.

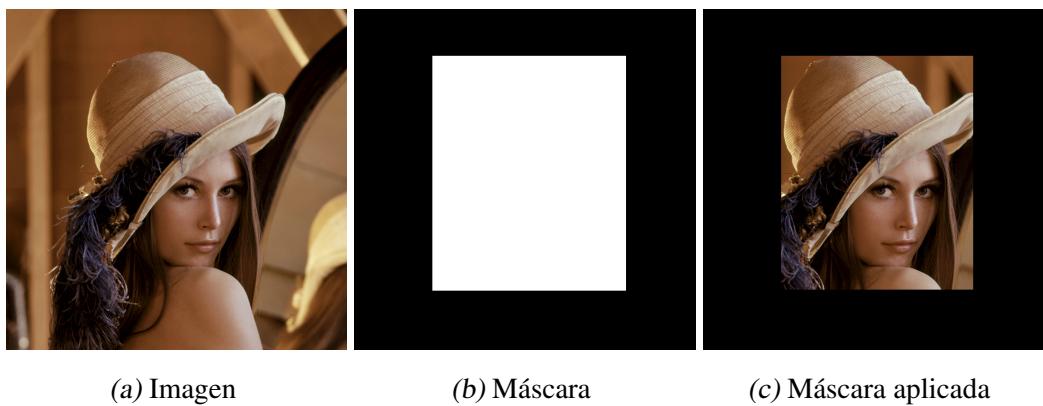


Fig. 1.10: Utilización de máscara sobre imagen

## *1.2. Trabajo Desarrollado*

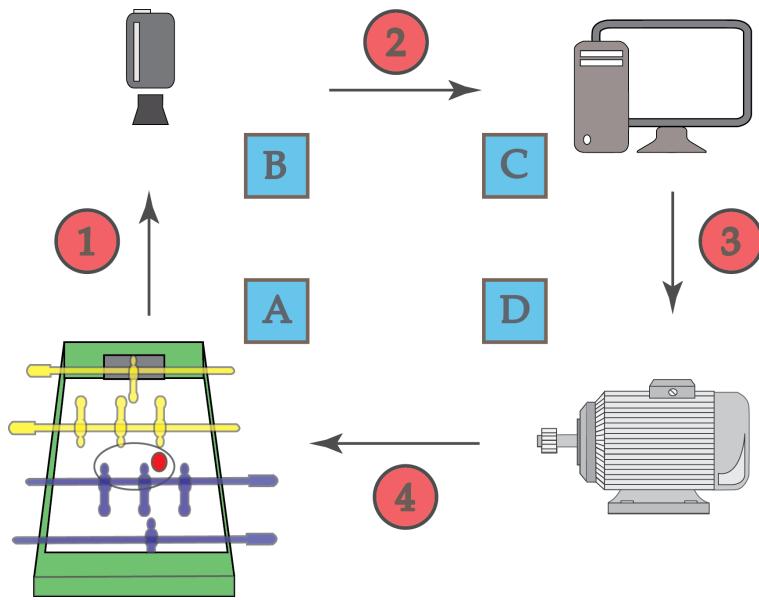
Como fue señalado, este tipo de proyecto en el cual se busca automatizar una “Table-Football” ya ha sido implementado por otras instituciones incluso llegando a ámbitos comerciales. En este sentido, el proyecto del cual forma parte este trabajo, no busca un fin comercial, ni competitivo con estas instituciones. La motivación y la meta a la cual se aspira es contar con una plataforma que permita realizar experimentos sobre ella, probar distintos algoritmos, hardware y software con fines tanto académicos como pedagógicos.

En esta primera etapa se plantea la automatización exclusiva de uno de los arqueros, quitando a los demás muñecos del juego. Este proceso posee tres puntos fundamentales enlistados más abajo, de los cuales sólo será desarrollado el punto 2 durante este estudio.

1. Adaptación física de la mesa de fútbol, tanto para sensores como actuadores.
2. Simulación y elaboración de algoritmos utilizando datos ficticios y reales.
3. Integración de los puntos 1 y 2.

Para ello, se debe tener en cuenta las limitaciones del juego en sí, como los rebotes y velocidad de la pelota durante el juego, ya que serán las características que definen los algoritmos a implementar, así como también los sensores que pueden ser utilizados.

Como propuesta al tema de investigación que busca automatizar el movimiento del arquero en el juego “Table-Football”, se plantea la utilización de procesamiento de imágenes para obtener las condiciones de juego, un algoritmo Neuro-Difuso (ANFIS) para estimar dónde irá a parar la pelota sobre el extremo del arquero, y finalmente, el modelado de un motor DC y diseño de un controlador que logre posicionar correctamente al arquero según una referencia dada, evitando de esta forma las anotaciones en portería.



*Fig. 1.11: Flujo de la información dentro del sistema*

En la figura 1.11 se muestran las cuatro estaciones del sistema A, B, C y D compuestas de la mesa de fútbol, cámara de vídeo, sistema computacional y motores respectivamente. Mientras que el flujo de la información está representada por los números 1, 2, 3 y 4 que significan adquisición de la escena, envío de la imagen al sistema computacional, entrega de referencia al motor y actuación sobre el juego. Se destaca que la estación C, es la encargada de procesar la información.

### 1.2.1. *Objetivo General*

Crear un sistema computacional que sea capaz de posicionar correctamente a un arquero del juego de “Table Football” para bloquear el paso de la pelota ante los tiros rivales, utilizando para ello procesamiento de imágenes y algoritmos Neuro-Difusos.

### 1.2.2. *Objetivos Específicos*

Dentro de los objetivos específicos propuestos, se encuentra:

- Encontrar la posición de la pelota en juego por medio de algoritmos de procesamiento de imágenes.
- Predecir el movimiento a futuro del balón con algoritmos Neuro-Difusos.
- Simular el comportamiento de un motor DC para mover al arquero.
- Diseñar un controlador para el motor DC que permita posicionarse correctamente según alguna referencia.

## 2. MATERIALES Y MÉTODOS

Como fue señalado previamente, este trabajo busca automatizar a uno de los arqueros, por lo que sólo se implementará una plataforma de juego que considere a uno de ellos, eliminando al resto de los muñecos, la superficie de juego para este trabajo será la misma, de  $1.2 \times 0.6$  [m<sup>2</sup>], pero no existirá intervención de los otros muñecos ni de sus respectivas barras, las cuales podrían bloquear el seguimiento de la pelota en la etapa de detección. La teoría y otros métodos serán tratados dentro de este capítulo, haciendo un análisis detallado sobre los factores a tener en cuenta durante este proceso de automatización.

### *2.1. Consideraciones acerca de la mesa de juego*

Se señalan a continuación, una serie de consideraciones que se deben tener en cuenta de la plataforma a automatizar y sus limitaciones.

#### *2.1.1. Superficie de la mesa*

La superficie de la mesa puede no ser plana. Algunos fabricantes construyen la superficie de la mesa de juego con cierta concavidad, lo cual permite que no existan grandes detenciones durante el juego, ya que previene que la bola quede atorada o quieta en un lugar donde los muñecos no puedan alcanzarla, brindando una mayor dinámica a los jugadores.

También se hace enfoque en que al utilizar superficies lisas como el acrílico, la manipulación de la bola como pases o gambetas, puede resultar más complejo que al hacerlo sobre

una plataforma áspera, producto de que el aumento en el roce de ambas superficies mejora el control sobre la pelota.

### *2.1.2. Trayectorias y rebotes del balón*

Uno de los factores importantes a la hora de analizar el sistema es el comportamiento que tendrá el objeto a medir, ya que permitirá establecer un correcto algoritmo de predicción.

En el caso de este juego se tendrán mayoritariamente movimientos rectos, y una menor porción corresponderán a tiros con efecto dado por el giro adicional que agrega movimiento del balón hacia los lados. Todas estas trayectorias, al estar inmersos dentro de un entorno cerrado estarán sujetos a golpes y rebotes, tanto con los límites de la mesa, los arcos o bien los muñecos, transformando el ambiente de juego en un sistema que resulta caótico y difícil de predecir con precisión.

Los muñecos poseen una base cilíndrica de caucho lo cual ayuda a golpear de mejor forma el balón independiente de la dirección. En cuanto a los límites de juego, estos generalmente son fabricados con madera o plástico, pero independiente de su composición, resulta complejo estimar la dirección, velocidad y aceleración del rebote, producto de que dependerá de la energía transferida y cambio en la cantidad de movimiento, necesitando de información como la densidad del material en cada punto para poder estimar correctamente cuanta energía será transferida y el movimiento resultante del balón.

### *2.1.3. Oclusiones*

Las occlusiones corresponden, en procesamiento de imágenes, a la pérdida del seguimiento de un objeto en una imagen producto que ha sido obstruida por alguna otra figura y a medida que aumenta el tiempo en que el objeto de interés se encuentra oculto, también lo hará la

incertidumbre de su posición (figura 2.1).

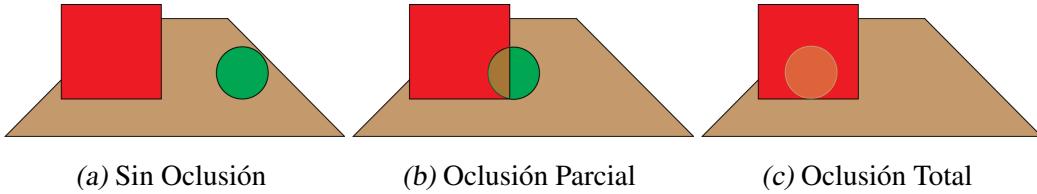
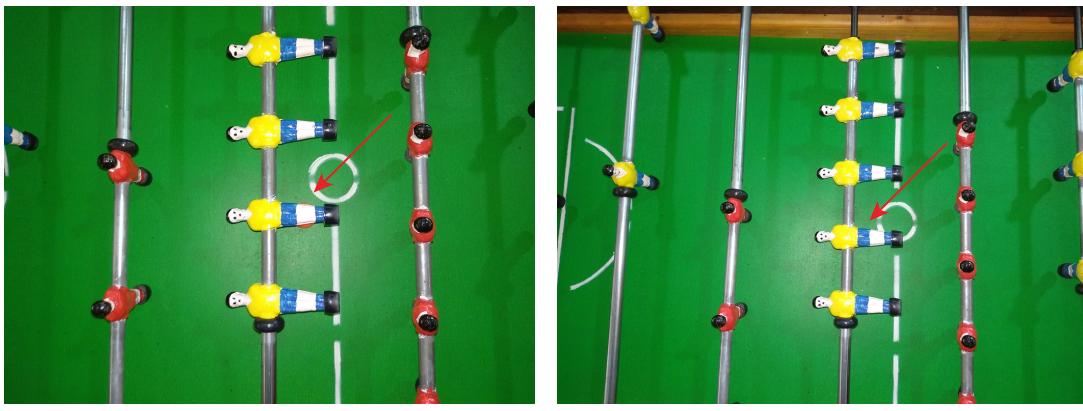


Fig. 2.1: Tipos de oclusiones

En nuestro caso de estudio existirán dos fuentes de oclusión que pueden comprometer el seguimiento del balón en las imágenes, las cuales corresponden a las barras y a los muñecos. En cuanto a las barras estas no poseen el diámetro suficiente para esconder a la pelota de forma total, pero si en una porción considerable. Mientras que los muñecos sólo obstaculizan el seguimiento cuando se encuentren en una posición horizontal con respecto a la mesa, generando una pérdida del seguimiento cuando la bola y el muñeco se encuentren alineados ante la cámara de video, no logrando percibir el balón.

Esta problemática no será tratada en este estudio ya que sólo se implementa el movimiento del arquero, por lo que la obstrucción de los otros jugadores no será tomada en cuenta. Cabe destacar que la existencia de una pérdida de referencia al utilizar a los otros muñecos es altamente probable, ya que la proporción del torso de los muñecos y la pelota, es casi idéntica (figura 2.2).



(a) Oclusión parcial del balón

(b) Oclusión total del balón

*Fig. 2.2: Pérdida del seguimiento de la pelota por oclusión*

#### 2.1.4. Velocidad máxima del balón

Según las reglas de las competencias internacionales, no existe una velocidad máxima a la que una persona pueda golpear la pelota [8], por lo tanto esta limitación será exclusivamente de la persona que se encuentre jugando.

Durante el transcurso del proyecto, no se ha logrado obtener información que cuente con un respaldo adecuado sobre la velocidad máxima alcanzada en competiciones, existen fuentes que señalan una velocidad de 33 m/s lo cual resulta poco creíble por su excesiva velocidad [9]. Otros mencionan competiciones donde se alcanzaron valores de 11 m/s. Por otra parte hay estudios que hablan de una velocidad promedio de tiro de 5 m/s.

Al no existir un claro registro de la velocidad que puede alcanzar, se procede a realizar mediciones que permitan obtener resultados verídicos. Destacando que, la rapidez con que se mueva la pelota, definirá las características tanto de los sensores como actuadores.

Se determina que la velocidad máxima que logró alcanzar la pelota fue de 5 m/s al golpearla hacia delante. Este resultado fue logrado por una persona no profesional en el juego, aunque sí aficionado. Con esta información se puede obtener además la velocidad con que se

mueve hacia los laterales, suponiendo una estructura cilíndrica en la base del muñeco (lugar que golpea el balón), se estima que se lograrán alcanzar valores de 2.5 m/s, producto del movimiento restringido de los muñecos al golpear la pelota rotando sobre las barras (figura 2.4).



Fig. 2.3: Vista superior. A la izquierda la base de un muñeco. A la derecha el balón

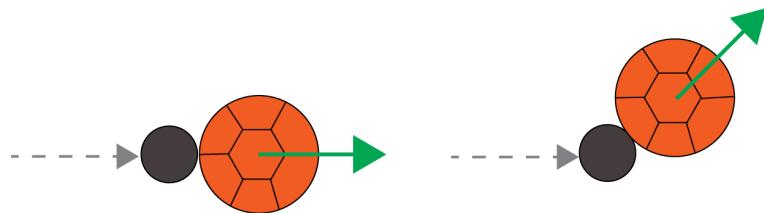
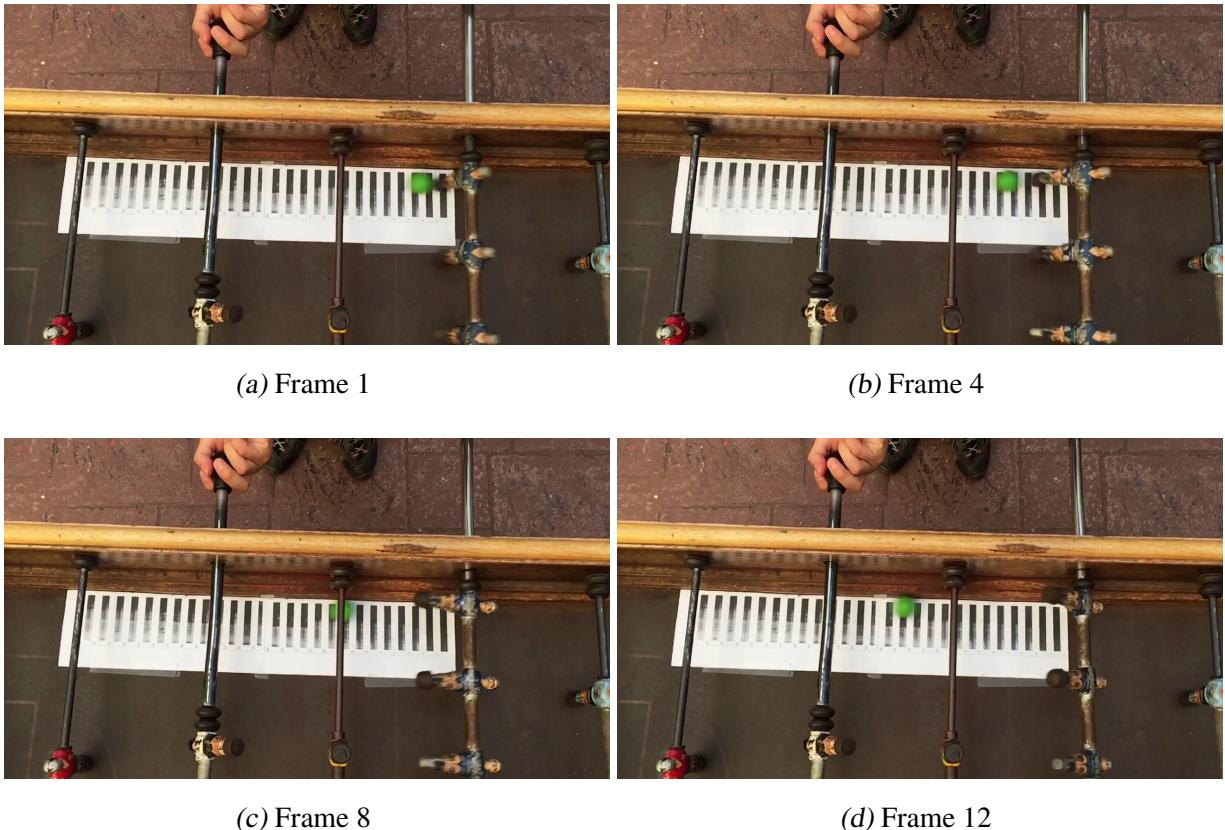


Fig. 2.4: Representación de tiros, a la izquierda tiro al frente y a la derecha tiro con ángulo.

Estas mediciones de velocidad se obtuvieron mediante la utilización de una hoja métrica con intervalos de 1 cm colocada en la mesa de fútbol, y con el uso de una cámara celular que permite grabar a 240 cuadros por segundo utilizando la cámara lenta (figura 2.5). Luego de analizar 4 vídeos diferentes, fue posible visualizar cuadro a cuadro, y considerando la tasa de adquisición de la cámara, se obtuvo que la velocidad máxima de desplazamiento corresponde a 4.8 m/s, la cual fue aproximada a 5 m/s ya que es posible que sea golpeada a una mayor velocidad.



*Fig. 2.5:* Cuadros para medir velocidad máxima. Tiempo cuadro a cuadro: 4.16 ms.

## 2.2. Sensor de vídeo y Medición

En esta sección se detalla la teoría relevante para la adquisición de las imágenes a través de un dispositivo de vídeo, así como también los complementos necesarios de esta. Además se describe el proceso de manipulación de las fotografías por medio del procesamiento de imágenes.

### 2.2.1. Cámaras de vídeo y complementos

Las cámaras de vídeo al igual que las fotográficas, poseen componentes similares en cuanto a los mecanismos utilizados para la adquisición de imágenes los cuales permiten y

ayudan a la obtención de buenos resultados, ya sea mediante la eliminación del ruido de las imágenes, ajuste del enfoque, colores más definidos o bien, efectos visuales que resulten más atractivos al espectador.

Los conceptos más recurrentes para una buena imagen son los referentes a iluminación y exposición de la cámara al momento de tomar las imágenes, pero también hay conceptos como la distancia focal, tamaño del sensor, lentes u objetivos y filtros los cuales también poseen roles importantes.

### *Iluminación*

La iluminación de una escena es necesaria en toda fotografía independiente de su origen natural o artificial, ya que sin ella no existirían rayos de luz que impacten en el sensor de la cámara, obteniendo como resultado una imagen completamente negra.

Cuando los haz de luz son reflejados por los objetos hacia el sensor, estos son adquiridos de acuerdo a su intensidad por medio del efecto fotoeléctrico (figura 2.6) en los sensores de las cámaras ([10]). Esta reacción física, produce un flujo de corriente a medida que las partículas de luz golpean al sensor, a mayor intensidad lumínica, mayor será la corriente generada por el detector fotosensible. Con este tipo de fenómeno es posible ir formando una recreación del ambiente u objeto, pero al igual que en el extremo de que no exista luz entrando al sensor, también existe el caso contrario, lo cual entrega como resultado una imagen con manchas blancas o completamente blanca, llegando a ser apreciado en superficies reflectantes, o bien al observar fuentes lumínicas como lámparas, donde una gran cantidad de luz es devuelta hacia la cámara.

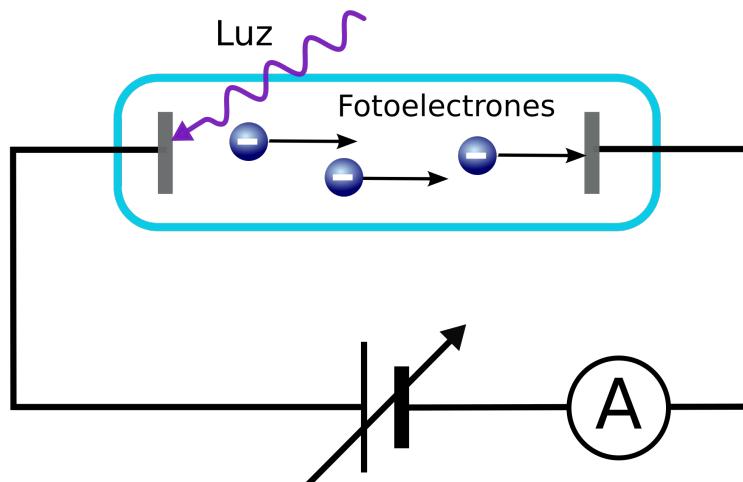


Fig. 2.6: Circuito representativo del efecto fotoeléctrico

Al no tener buenas condiciones de ambiente como la escasez de fuentes lumínicas, se puede pasar de tener una iluminación natural a una artificial, siendo esta última una forma de tener cierto control sobre la escena que se busca retratar ya que se puede modificar la intensidad según se estime conveniente, lo cual no es posible realizar con la luz natural.

Un factor importante a la hora de iluminar la escena, es la generación de sombras (figura 2.7) las cuales pueden resultar perjudiciales en procesamiento de imágenes, ya que resultan en componentes dinámicos en el caso de la luz natural (según la hora del día), o bien cuando existen objetos en movimiento, lo cual dificulta la obtención de información en la imagen.



Fig. 2.7: Sombras de las barras y muñecos sobre el campo de juego.

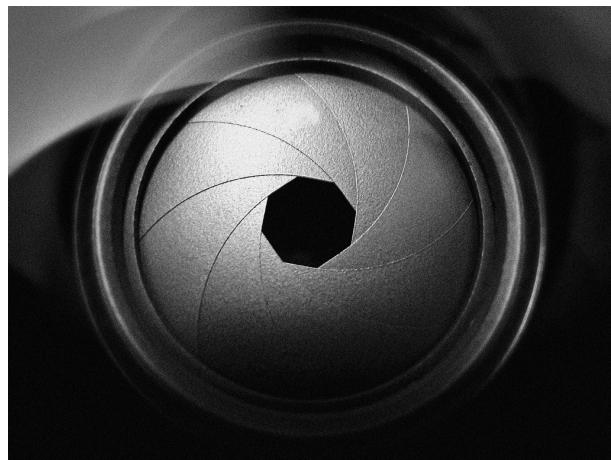
### *Exposición*

El segundo aspecto a tener en cuenta a la hora de realizar fotografías o grabar vídeos es la exposición o configuración de la cámara, la cual posee herramientas mecánicas o digitales según sea el caso. Algunas de estas herramientas son la apertura del diafragma, velocidad de obturación y el ISO o sensibilidad del sensor a la luz. Estos tres componentes (apertura, obturación e ISO) corresponden a un set de recursos que el usuario puede modificar según la necesidad del caso, logrando distintos efectos lo que permite mejorar la calidad de las tomas.

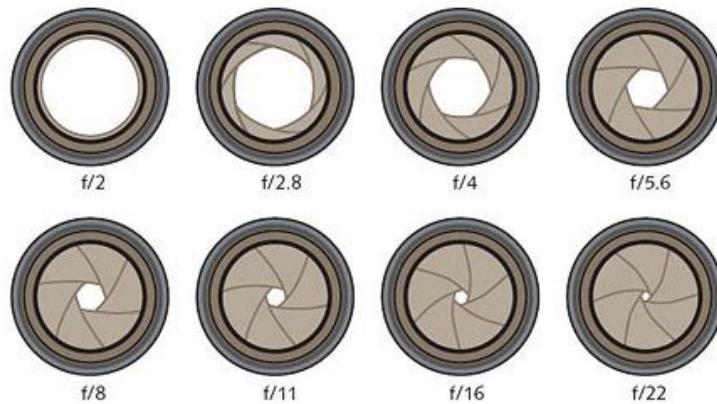
### *Apertura y diafragma*

La apertura corresponde a qué tan abierto se encuentra el diafragma (figura 2.8). El diafragma consiste en una pieza mecánica que permite regular la cantidad de luz que entra al dispositivo mediante la apertura o cierre del mismo. Para su regulación, éste posee una escala de valores “f” medida en pasos (figura 2.9), donde f/1 es una apertura completa (donde se deja entrar la mayor cantidad posible de luz), f/2 reduce a la mitad la cantidad de luz que entra al dispositivo, f/4 la reduce a un cuarto, etc. Por supuesto, no existe una mayor apertura

que f/1 ya que el diafragma no inyecta luz al sistema, sólo la reduce.



*Fig. 2.8:* Diafragma de una cámara de vídeo



*Fig. 2.9:* Valores posibles del Diafragma en la escala f

### *Obturación*

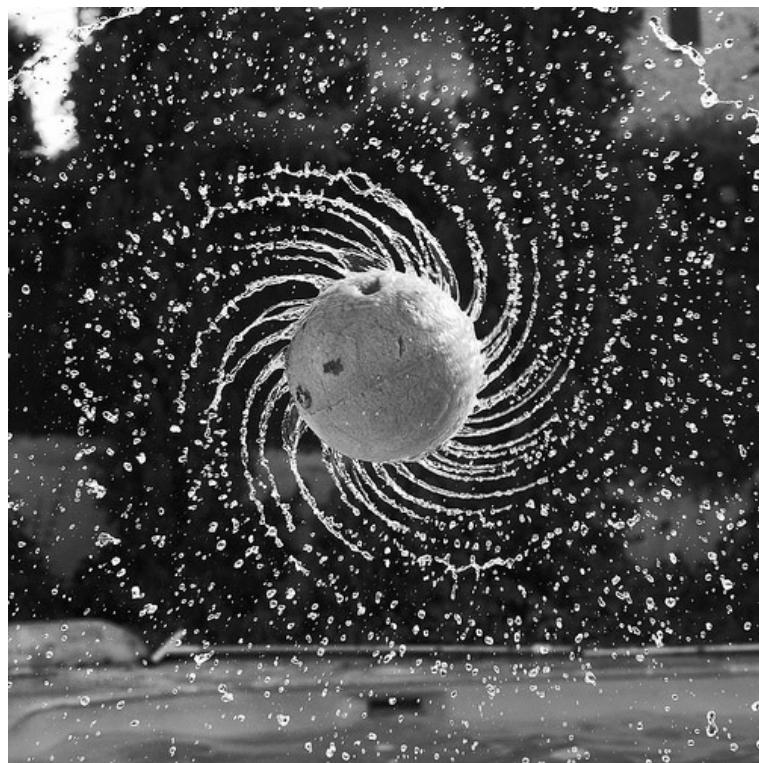
El obturador es un sistema mecánico o digital que permite exponer o bien ocultar al sensor de la luz de la escena para formar la imagen. Muy parecido a una cortina en las cámaras fotográficas, el obturador bloquea el paso de luz al sensor con una cierta frecuencia, esta velocidad de obturación o también llamado tiempo de exposición logra distintos efectos dependiendo si es una alta o baja velocidad.

Cuando el tiempo de exposición es alto, por ejemplo 4 segundos, el sensor estará recolectando la información de la escena durante todo ese tiempo, por lo que si existe movimiento o algún cambio en los objetos que están siendo retratados, éstos dejarán un rastro por donde pasaron sobre la imagen lo cual puede resultar atractivo desde el punto de vista artístico (figure 2.10).



Fig. 2.10: Tiempo de exposición alto

Por el contrario, al tener un tiempo de exposición bajo, por ejemplo 1/30 (0.033 segundos), permitirá retratar de mejor forma los objetos en movimiento, siempre y cuando el desplazamiento del objeto en la imagen, dentro de esos 0.33 segundos sea pequeña, de lo contrario se obtendría un resultado similar al caso anterior o bien con objetos difuminados (figura 2.11).



*Fig. 2.11: Tiempo de exposición bajo*

A pesar de que un aumento de la velocidad de obturación trae un beneficio a la hora de obtener imágenes más definidas, también conlleva a un problema de iluminación. Ésto producto de que al existir una mayor velocidad de obturación, la cantidad de luz que llega al sensor se ve reducida, lo que tiene por efecto una imagen más oscura. En los casos extremos cuando se tienen tiempos de 1/1000 (0.001 segundo) resulta necesario incluir iluminación artificial para incrementar la información que es adquirida.

En cuanto a la configuración de este parámetro, éste debe ser ajustado de acuerdo al objeto o condiciones que se buscan retratar, siendo la nitidez del objeto o efecto de permanencia en la imagen.

Una característica de las cámaras de vídeo, es que poseen dos tipos de obturadores, uno giratorio o rodante y otro global, siendo un sistema mecánico y digital respectivamente. El obturador rodante consiste en un dispositivo que permite el paso de luz al sensor mediante el

giro de una lámina con un hueco descentrado, el cual al girar, permite que la luz entre o sea bloqueada dependiendo de la alineación entre la apertura y el sensor (figura 2.12). El inconveniente de este tipo de obturación, es que no toda el área del sensor es expuesta al mismo tiempo, lo cual provoca objetos distorsionados ante la alta velocidad de éstos.

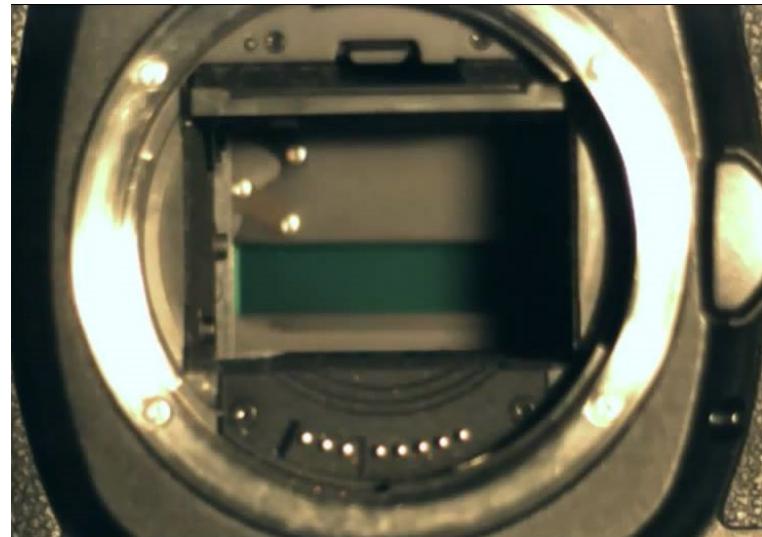
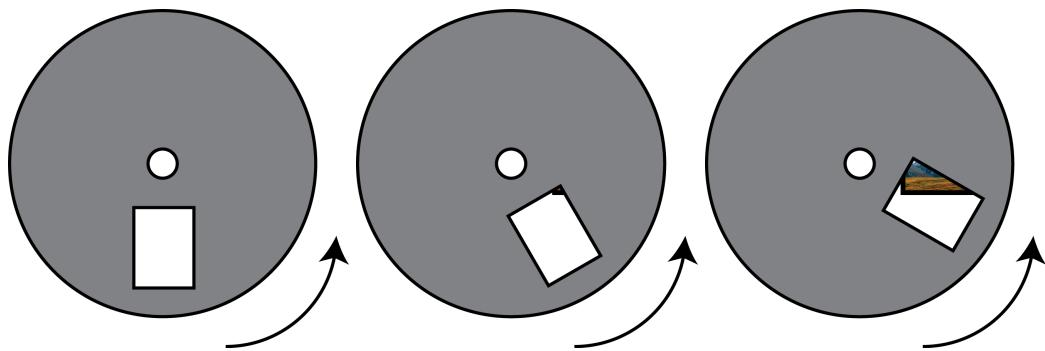
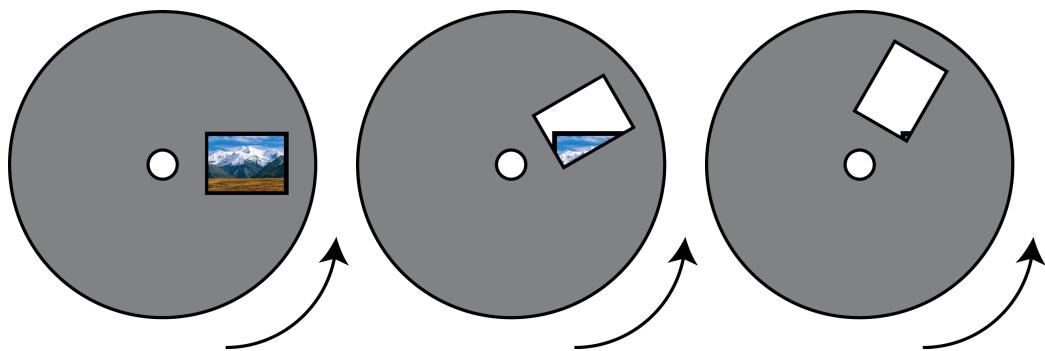


Fig. 2.12: Obturador rotatorio



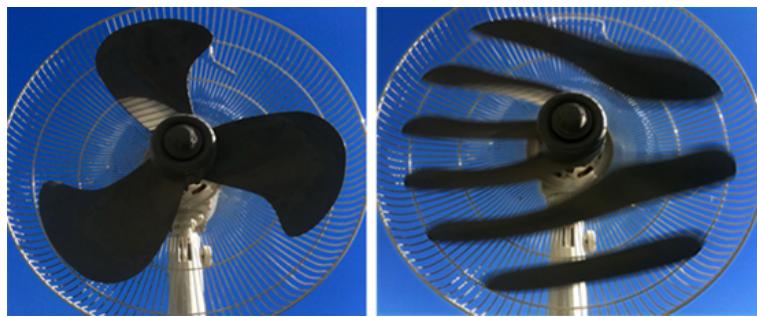
(a) Obturador en posición 1 (b) Obturador en posición 2 (c) Obturador en posición 3



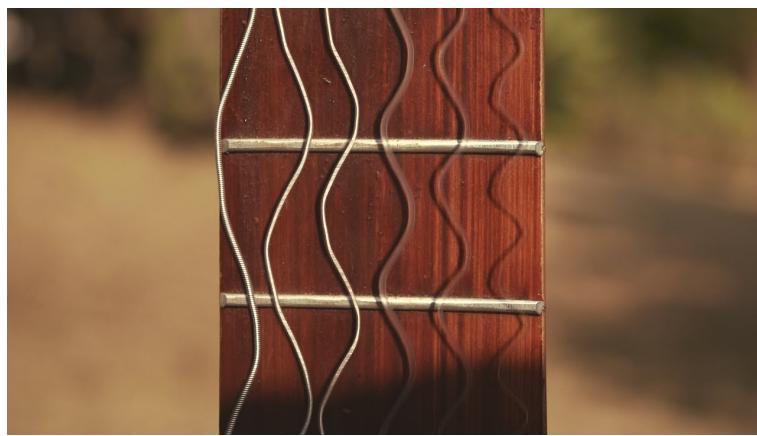
(d) Obturador en posición 4 (e) Obturador en posición 5 (f) Obturador en posición 6

*Fig. 2.13: Representación de un obturador rotatorio*

En la figura 2.13, se aprecia el proceso de exposición de luz al sensor con un sistema rotatorio, donde se aprecia que al ir girando, no toda la superficie es expuesta al mismo tiempo, produciendo que ciertos píxeles capturen la luz entrante antes que otros. Siendo los efectos reflejados en la figura 2.14, donde las aspas se deforman en la imagen producto de el obturador rodante. Las cuerdas de guitarra, toman un efecto oscilatorio estacionario, cuando la razón es que la cámara toma distintos instantes del movimiento de la cuerda como si fueran uno sólo. Y finalmente, los trenes o vehículos capturados por este tipo de obturación, genera una deformación de los objetos.



(a) Efecto en las aspas de ventiladores (y aviones)



(b) Efecto en cuerdas de instrumentos musicales



(c) Efecto en la imagen de un tren

*Fig. 2.14: Diversos efectos en imágenes causados por un obturador rotatorio*

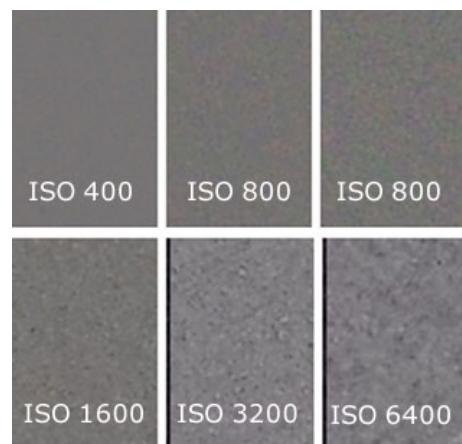
Por otro lado el obturador global es un sistema digital, el cual mediante instrucciones eléctricas indica al sensor cuando medir o dejar de hacerlo, lo cual evita problemas como los

antes señalados ya que toda la superficie del sensor actúa de la misma manera, en el mismo instante y por igual cantidad de tiempo.

### *ISO*

El último término de exposición corresponde al ISO, el cual hace referencia a la sensibilidad con la que el sensor logra medir la luz que llega sobre su superficie. Una mayor sensibilidad permite capturar la información en ambientes oscuros. Antiguamente se utilizaban películas (o rollos) fotográficos los cuales poseían una determinada y única sensibilidad, con la tecnología este recurso dejó de existir, dando paso a una sensibilidad que puede ser ajustada de forma digital por medio de una amplificación de la señal eléctrica.

Resulta oportuno mantener esta ganancia lo menor posible, ya que al ser una amplificación de la señal eléctrica que es capturada, el aumento de los valores es realizado de igual forma a toda la imagen, lo que produce que el ruido también sea amplificado (figura 2.15). Debido a esto, se prefiere obtener buenos resultados modificando la velocidad de obturación y apertura del diafragma antes de subir el valor del ISO.



*Fig. 2.15:* Diferencias al realizar un ajuste del ISO

## *Lentes y Objetivos*

Los objetivos son un conjunto de lentes (figura 2.16) que convergen la luz entrante a un punto en concreto llamado foco. Este conjunto de lentes cóncavos, convexos, bicóncavos o biconvexos, permiten ir modificando la trayectoria a medida que la luz atraviesa estos objetos, los cuales a diferencia de un prisma, no separan los distintos componentes (frecuencias) presentes en ella, sino que sólo cambian la trayectoria de esta.



*Fig. 2.16: Lentes Basler*

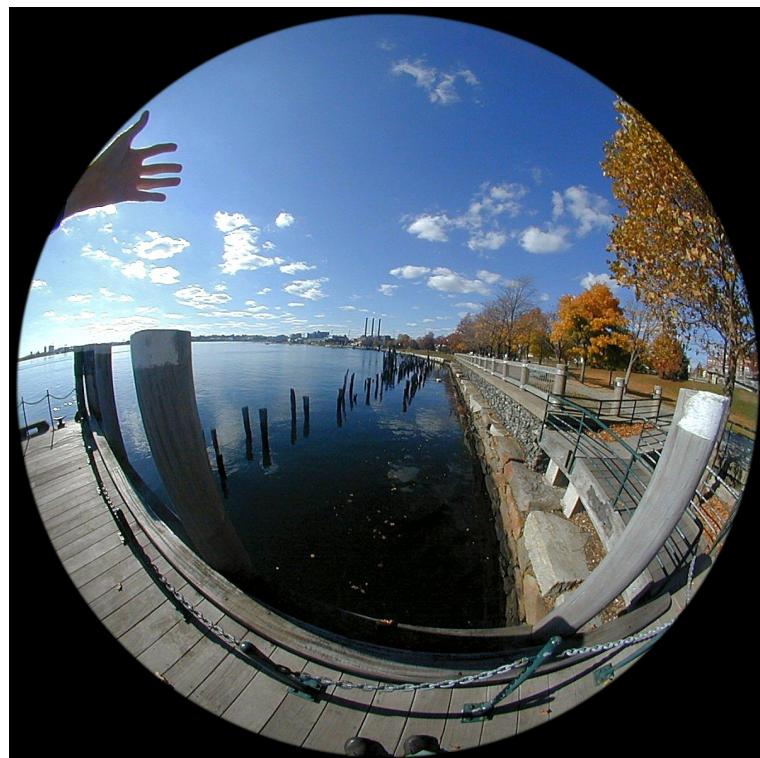
Sin un objetivo, el sensor de la cámara no puede construir una imagen de la escena, producto de que los haz de luz provenientes de diversas fuentes, estarían llegando a un mismo punto en el área del dispositivo, entregando como resultado una mancha borrosa. Para evitar este problema, la utilización de un lente o conjunto de lentes resulta indispensable, ya que elimina los rayos de luz de objetos colindantes y provocando la convergencia de la luz proveniente de un punto de la escena a un punto del sensor. Esto último, es producto de que la emisión de rayos de luz es efectuada en todas direcciones y pueden ser vistos como líneas paralelas a la hora de llegar al objetivo.

Existe una diversa cantidad de objetivos:

- Ojo de pez: Gran ángulo de visión pero con alta distorsión.
- Gran angular: Alto campo de vista, sin mucha distorsión.

- Fijos: Con distancia focal fija, no permitiendo realizar zoom. Campo de visión normal.
- Zoom: Amplio rango focal. Permiten realizar zoom en un amplio rango.
- Tele-objetivos/Super-tele-objetivos: Poseen un largo focal muy alto, logrando enfocar objetos a gran distancia.
- Macro: Logran realizar enfoques a distancias cortas, magnificando la imagen.
- Tilt-Shift: Realiza rotaciones sobre el eje del lente, permitiendo corregir perspectivas en fotografía de arquitectura.

Finalmente la selección y utilización de ellos, queda determinada por las condiciones del ambiente y resultados que se esperan obtener.

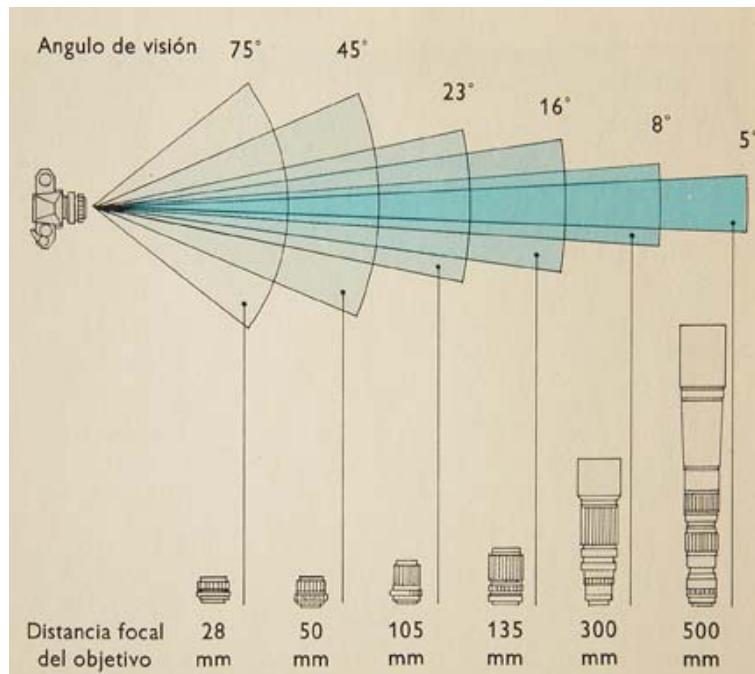


*Fig. 2.17: Foto con lente Ojo de pez*

## *Distancia Focal*

La distancia focal hace referencia al punto donde convergen los rayos de luz luego de pasar por los lentes convergentes. El sensor de la cámara, ubicado delante del punto de convergencia será el encargado de adquirir la información de los rayos luminosos.

Un cambio en la distancia focal modifica la superficie de escena que cubrirá el sensor, logrando capturar más información del ambiente cuando la distancia focal es pequeña y una menor y concentrada superficie cuando la distancia focal es grande (figura 2.18). Esto se debe a que cuando aumentamos o disminuimos la distancia focal el campo de visión varía, al igual que lo que ocurre cuando miramos por una ventana, si nos acercamos a ésta observamos cómo nuestro campo de visión hacia fuera de esta comienza a aumentar.



*Fig. 2.18: Distancias focales*

Un efecto utilizado en películas haciendo uso de la modificación de la distancia focal es el efecto "Dolly"([11]), donde se cambia la distancia focal a medida que se mueve la cámara,

lo que entrega una sensación de distanciamiento entre los objetos en escena.

### *Tamaño del sensor*

El tamaño del sensor de una cámara fotográfica o de vídeo no se relaciona con el tamaño en píxeles, sino que en el largo y ancho que este posee, siendo un factor del campo de visión que pueda abarcar, ya que mayores dimensiones significan más espacio donde proyectar la luz entrante (figura 2.19), lo cual provocará que el fotógrafo pueda cubrir mayores superficies sin tener que alejarse de la escena.

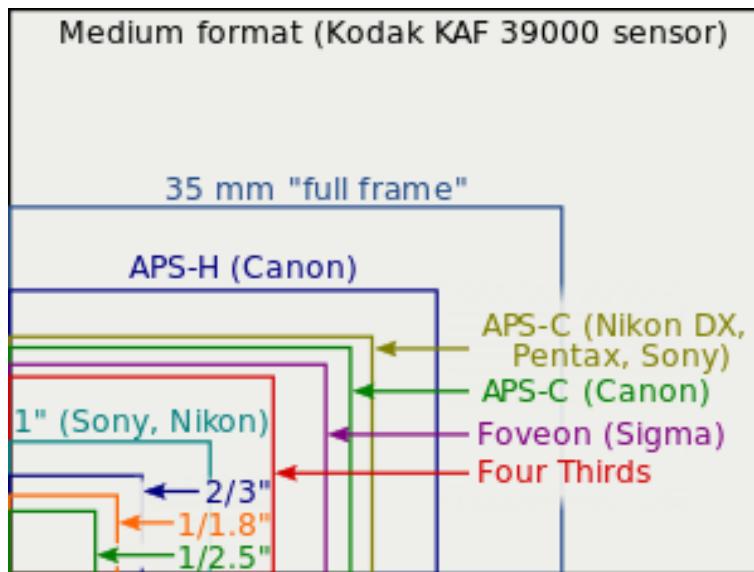


Fig. 2.19: Tamaño de sensores en cámaras

### *Selección del dispositivo de Vídeo*

Uno de los objetivos del proyecto es encontrar la posición de la pelota en juego por medio de algoritmos de procesamiento de imágenes, para lograr esto se debe contar con un dispositivo que capture las imágenes de la escena de forma continua, con la suficiente rapidez para permitir actuar de acuerdo a las limitaciones del propio juego y con la capacidad de entregar las imágenes capturadas en linea, sin necesidad de almacenarlas. Para ello se tomó en

consideración las velocidades máximas a la cual puede viajar el balón, así como también las dimensiones efectivas del campo de juego.

Como fue señalado, las dimensiones del campo de juego que se utilizarán en este estudio son de  $1.2 \times 0.6$  [m<sup>2</sup>], pero los arqueros de ambos equipos se encuentran a 12 [cm] de sus extremos. Esto significa que las dimensiones efectivas que la cámara de vídeo debe capturar son de  $0.96 \times 0.6$  [m<sup>2</sup>], dado que no interesa lo que ocurre más allá de las posiciones de los arqueros.

Finalmente, se considera el peor caso para que un arquero logre detener un tiro rival. Esto sucede cuando un muñeco de la linea delantera del equipo rival es el que golpea la pelota a máxima velocidad y rebota en un defensa cambiando su dirección, ya que correspondería a la menor distancia recorrida y la mayor velocidad posible (5 [m/s]). En este caso se tiene que la pelota recorrería 0.12 [m] al cambiar de dirección en alguno de los defensas (figuras 1.5 y 1.6). Ante esta situación, el tiempo que demora la bola en llegar a la linea del arquero es de 24 [ms] al considerar una velocidad constante ( $Velocidad = distancia/tiempo$ ).

Esta información permite estimar la velocidad con que la cámara deberá adquirir los datos de la escena, recordando que para lograr estimar la dirección de la pelota se necesitan al menos dos puntos y además que el sistema tendrá un retardo computacional ya que debe ser capaz de adquirir los datos, procesarlos y actuar acorde a estos. Estas consideraciones llevan a que el dispositivo de vídeo, así como el sistema en general, deba actuar con mayor rapidez que los 24 [ms]. Como limitación se consideró que el dispositivo debe ser capaz de capturar la información de la escena en un 10 % del tiempo límite estimado, lo que nos dá una velocidad mínima de adquisición de 416 cuadros por segundo, aunque dado que es un estimado se prefiere la utilización de velocidades superiores.

Con los resultados obtenidos de la velocidad máxima que puede alcanzar la pelota dentro del juego, el proyecto seleccionó la cámara Basler acA800-511uc [12], la cual permite capturar hasta 511 cuadros por segundo según sus especificaciones, con una resolución máxima

de 800 x 600 píxeles, entregando imágenes a color y con una interfaz de conexión USB 3.0.

### *2.2.2. Procesamiento Digital de imágenes*

Procesamiento digital de imágenes se describe como todas las técnicas computacionales para la manipulación de una imagen, siendo hoy en día una herramienta extensamente utilizada por casi todo dispositivo telefónico móvil en mayor o menor medida. Este procesamiento parte de la base de que la imagen ha sido digitalizada, permitiendo su manipulación a nivel de píxeles.

Los píxeles, son la unidad básica de toda imagen digital, los cuales están compuestos por tres indicadores cuantizados que reflejan la intensidad o brillo de luz roja, verde y azul (RGB por sus siglas en inglés) presentes en un punto de la imagen (figura 2.20). Éstos son ordenados en una estructura cuadriculada donde la resolución de la imagen, nos indica la cantidad de píxeles en horizontal y vertical que posee. Una mayor resolución, permite tener imágenes mejor definidas, lo que a su vez equivale a una mayor cantidad de información proveniente de la escena retratada.



*Fig. 2.20: Píxeles en una pantalla de televisor*

La obtención de información, análisis o modificación de una imagen digitalizada, contempla la manipulación o lectura de los valores contenidos en cada uno de estos píxeles. Destacando que estos corresponden a valores numéricos en un rango 0 a 255 por canal (R, G o B) indicando la cantidad de luz que estos poseen. Siendo ordenados de forma matricial (figura 2.21), la implementación de algoritmos en este tipo de casos, casi siempre contempla la manipulación y uso de matrices, ya que esto facilita de cierta forma el procesamiento computacional.

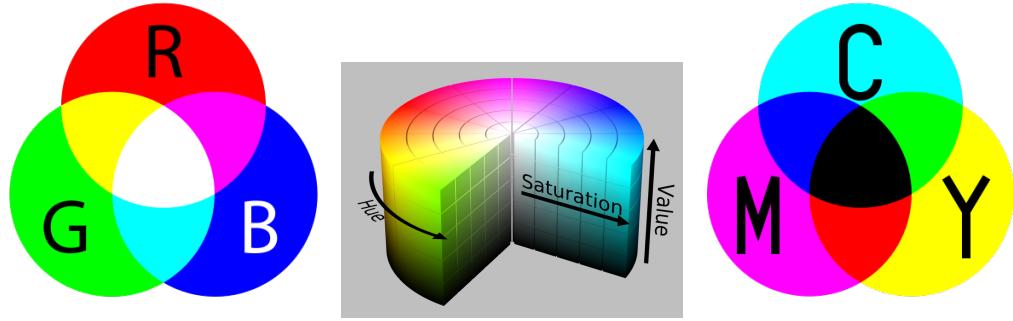
1	0	1	2	3	4	5	6	7	8	9
2	1	2	3	4	5	6	7	8	9	10
3	2	3	4	5	6	7	8	9	10	11
4	3	4	5	6	7	8	9	10	11	12
5	4	5	6	7	8	9	10	11	12	13
6	5	6	7	8	9	10	11	12	13	14
7	6	7	8	9	10	11	12	13	14	15
8	7	8	9	10	11	12	13	14	15	16
9	8	9	10	11	12	13	14	15	16	17
10	9	10	11	12	13	14	15	16	17	18
	1	2	3	4	5	6	7	8	9	10

Fig. 2.21: Valores de píxeles en una imagen en escala de grises

### Espacio de colores

Los espacios de colores son formas que permiten representar el color. Es muy común escuchar el término RGB, el cual hace mención a que cada píxel tendrá tres valores, comúnmente de 8 bits cada uno, los cuales representaran las tonalidades de rojo, verde y azul por sus siglas en inglés.

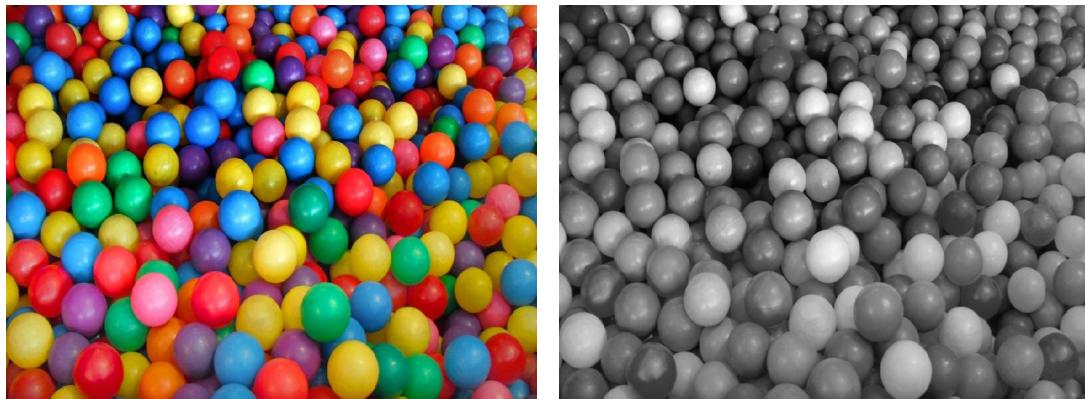
Existen muchas formas de representar los colores, donde RGB, HSV, CMYK (figura 2.22) son los más comunes, utilizados para facilitar la manipulación según sea el caso, por ejemplo el sistema CMYK es utilizado en sistemas de impresión, donde su nombre proviene de los colores cian (C), magenta (M), amarillo (Y) y negro (K). El sistema RGB es utilizado por televisores o pantallas ya que la información va directa a los diodos emisores de luz de cada píxel ya que utilizan la misma configuración de colores. En cuanto al sistema HSV, esta representa el matiz (Hue), nivel de saturación (Saturation) y brillo (Value) que cada pixel posee, el cual es frecuentemente utilizado cuando se necesita analizar la composición de colores en una imagen, ya que esta información es obtenida por medio de la lectura de un sólo canal.



(a) Espacio de colores RGB (b) Espacio de colores HSV (c) Espacio de colores CMYK

*Fig. 2.22: Algunos tipos de espacios de colores*

Otro sistema muy utilizado es la escala de grises, en donde la información se ve reducida a sólo un canal el cual muestra la intensidad de la luz. En esta escala no se obtiene información de los colores por lo que aunque tengan distintos matices, estos pueden tener igual valor de intensidad pareciendo que fueran del mismo color (figura 2.23).



(a) Imagen a color

(b) Imagen a escala de grises

*Fig. 2.23: Comparación entre espacios de colores y pérdida de información*

## Filtros Digitales

Los filtros digitales son utilizados para modificar la imagen de distintas formas, disminuyendo cambios abruptos de intensidad (filtro pasa-bajos), intensificando los cambios de color (filtros pasa-altos), promediando las intensidades (filtros suavizantes), eliminando el ruido, etc. El resultado de estas operaciones, es una nueva imagen modificada.

Para utilizar y aplicar estos filtros, es necesario formar un kernel o matriz (figura 2.24), que contendrá los coeficientes cuyos valores dependerán del resultado que se quiera lograr. El kernel irá haciendo un barrido por cada pixel de la imagen, donde irá realizando multiplicaciones entre el kernel y los valores de la imagen, donde el resultado de esta operación será reflejada en una nueva imagen (figura 2.25). La forma y tamaño de la matriz del filtro es determinada por el usuario, aunque es común que las dimensiones sean impares, superponiendo el elemento central del kernel con el pixel donde va a ser aplicado el filtro.

1	1	1
1	1	1
1	1	1

3x3 Mean kernel

1	4	7	4	1
4	16	26	16	4
7	26	41	26	7
4	16	26	16	4
1	4	7	4	1

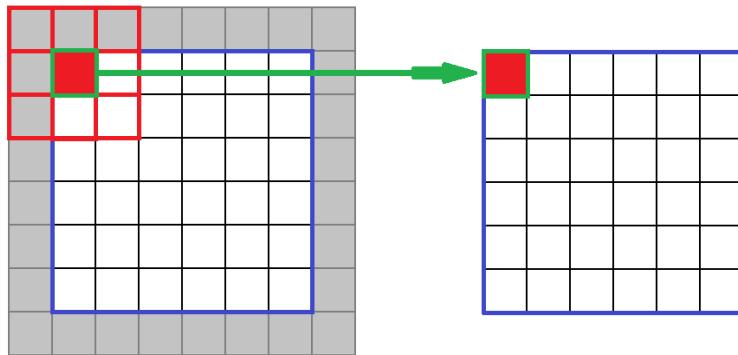
5x5 Gaussian kernel

2	1	0
1	0	-1
0	-1	-2

3x3 Sobel kernel

Fig. 2.24: Algunos tipos de kernel

Los kernels deben tener ser normalizados luego de convolucionar las matrices, de esta forma mantener los valores entre los rangos 0 a 255, correspondiente a las intensidades de los píxeles.



*Fig. 2.25: Aplicación de filtro sobre imagen y generación de una nueva*

### *Ruido en imágenes*

El ruido es causado en la etapa de adquisición, cuando la imagen es formada en el sensor mediante la medición del flujo eléctrico proporcional a la intensidad lumínica que es recibida. El sensor al ser un circuito eléctrico, es propenso a ser alterado por la temperatura y dada la dificultad para realizar cada celda fotosensible exactamente igual a las demás existirán diferencias en su comportamiento, por mínimas que sean estarán siempre presentes desde su fabricación. Además la estructura reducida, produce que cada celda se encuentre muy próxima a las demás haciendo que los píxeles interactúen unos con otros al aumentar la temperatura.

Al aumentar la sensibilidad ISO de una cámara, también se está incrementando estas pequeñas diferencias en los valores eléctricos de cada foto-sensor, lo cual trae como consecuencia un aumento desigual, llevando a la aparición de más ruido en la imagen.

Los dos tipos más comunes corresponden al ruido Gaussiano (figura 2.26a) y sal y pimienta (figura 2.26b).



(a) Ruido Gaussiano

(b) Ruido Sal y Pimienta

Fig. 2.26: Tipos de ruido en imágenes

### *Segmentación y contornos*

El identificar objetos dentro de las imágenes es un proceso muy utilizado por las aplicaciones actuales, ya que permite extraer información del ambiente como la posición de los objetos, identificar movimientos, formas, etc. La segmentación es una forma de separar objetos dentro de una imagen por medio de criterios, como el color, cambios de intensidad, distancias, por nombrar algunos.

Un método simple de segmentar una imagen, es a través de establecer rangos de colores y separar los datos de acuerdo a estos, el problema de realizar este tipo de segmentación es que si en el ambiente existen otros objetos aparte del de interés que comparten colores similares, resultará muy difícil sólo obtener componente deseado. Por el mismo motivo, resulta necesario tener cierto conocimiento de los objetos que estarán involucrados en la escena, o bien tener control sobre ésta.

Entre los criterios para segmentar una imagen, destacamos para este estudio la utilización de los cambios de intensidad que puedan existir a medida que realizamos un barrido por los

píxeles. Si existe un cambio significativo, es probable que estemos pasando de un objeto a otro, si este patrón tiene un comportamiento continuo y no aislado dentro de su sector, la probabilidad de que sea un objeto de interés aumenta. Esta forma de buscar cambios de intensidad sirve como una primera idea a la hora de buscar bordes dentro de una imagen.

### *Identificación de Objetos*

En cuanto al algoritmo de identificación, se han analizado cuatro métodos, cada uno con ventajas y desventajas:

1. Según las características del objeto (“Features Matching”).
2. Por cambios bruscos de intensidad (Encontrando bordes y clasificándolos).
3. Según los colores de los objetos.
4. Por comparación de imágenes.

### *Detección por “Features Matching”*

“Features Matching” es un algoritmo de detección de objetos que utiliza la fuerza bruta para lograr identificarlos en las imágenes. Esto quiere decir que el objeto es buscado por cada zona de la imagen, recorriendo cada uno de los píxeles [13].

Para el proceso de detección, se carga una imagen del objeto a buscar y se le extraen características de este, como por ejemplo, la existencia de esquinas, bordes, etc. Luego estas características del objeto, son buscadas por toda la imagen.

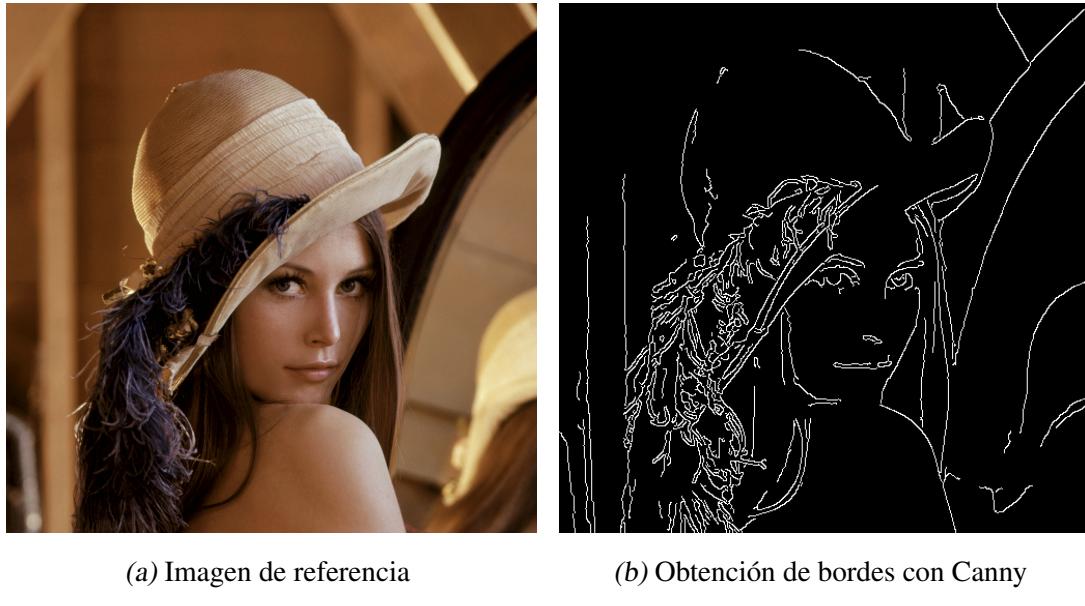
Una ventaja con que cuenta algunos algoritmos de este estilo (ej. SURF: “Speeded-Up Robust Features” [14]) es la no incidencia en la orientación del objeto, ni la escala de este. Es decir que no importa si posee un distinto tamaño y se encuentra rotado, el algoritmo será

capaz de identificarlo de igual forma.

Una desventaja, es que el tiempo requerido para encontrar la ubicación del objeto aumenta considerablemente, ya que resulta en un proceso iterativo con búsquedas en varias escalas para lograr identificarlo.

#### *Detección por cambios bruscos de intensidad*

La búsqueda de objetos en una imagen puede ser resuelta al encontrar cambios bruscos, ya que el objeto generalmente no tendrá la misma tonalidad de fondo produciendo una diferencia en los valores de los píxeles considerable. Esta característica es explotada utilizando algoritmos como Canny ([15]) el cual entrega como resultado los bordes de una imagen (figura 2.27), para posteriormente se utilizados para encontrar la posición de los objetos.



(a) Imagen de referencia

(b) Obtención de bordes con Canny

*Fig. 2.27: Uso del algoritmo Canny*

La gran desventaja que posee este tipo de algoritmos, es la dependencia de la calidad de la imagen, que en la presencia de ruido, se detectan contornos donde no los hay, introduciendo

etapas de identificación para lograr determinar si el borde detectado es realmente un objeto o no, ya sea por su tamaño o forma, resultando de esta manera en un proceso lento de ejecutar ya que se deben comparar todos los contornos que fueron encontrados en la imagen, resultando la eliminación de ruido una etapa fundamental.

### *Detección por color*

Resulta intuitivo la utilización de un detector por color en el procesamiento de imágenes, ya que es generalmente lo que hacen las personas en el día a día, aun así, esta forma de procesar la información no siempre tiene buenos resultados, siendo afecta a ruido, iluminación de la imagen, colores distintivos para cada objeto y lo principal, la existencia de color en la imagen. Si bien pueden ser utilizados detectores por intensidad en imágenes monocromáticas, la dificultad de que el detector funcione de una forma robusta aumentan, dada la disminución de información que se percibida.

La ventaja directa de utilizar detección por color es la sencillez con que puede ser aplicado, identificando el rango de color por cada pixel ya entrega la información necesaria para lograr ubicar al objeto. La desventaja es que otros objetos de similar tonalidad puede estar presente en la imagen y ser detectados erróneamente.

### *Detección por comparación de imágenes*

Al igual que en el caso anterior, el echo de comparar imágenes para señalar si un objeto que estaba presente ya no lo está resulta intuitivo. Esta forma de identificar a los cuerpos puede resultar idónea cuando solamente el objeto de interés posee movimiento y la cámara se encuentra fija en su lugar.

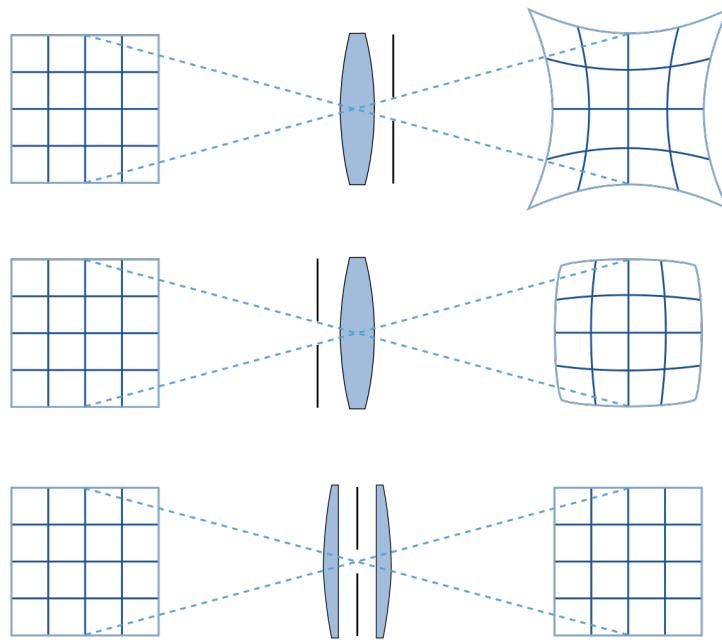
Existen 2 formas de implementar este tipo de algoritmos, utilizando una comparación

dinámica con el instante previo, o bien comparar con el instante inicial. Ambos métodos poseen cierta ventaja y desventaja, en el primero si el objeto se mueve rápidamente existirán dos cambios percibidos, la posición anterior y la actual ya que en ambos lugares existe un cambio de valor en los píxeles, la ventaja que este método posee es que si se añaden objetos a la imagen, estos dejarán de ser percibidos una vez se encuentren quietos ya que aparecerán de igual forma en la imagen previa y actual. El segundo método, tiene como ventaja la detección del objeto de forma única, sin duplicar los cambios, pero al contrario del anterior es propenso a realizar la detección continua de nuevos objetos que se incorporen a la escena.

Si bien es un algoritmo de rápida ejecución, ya que sólo contempla la utilización de una resta entre imágenes, en nuestro caso de estudio existirán comportamientos dinámicos aparte del movimiento de la pelota en juego. Los muñecos de ambos equipos estarán en continuo movimiento durante el juego, con lo que no se tendrá certeza de cual es el balón y cuales los muñecos. Además, al utilizar una cámara de alta velocidad los cambios entre el momento previo y actual serán casi imperceptibles a bajas velocidades de la pelota.

### *Geometría Proyectiva*

La geometría proyectiva es comúnmente utilizada en procesamiento de imágenes, como una herramienta de corrección [16]. Anteriormente se ha hablado que los objetivos pueden introducir distorsión sobre la imagen (figura 2.28), como lo es en el caso de un lente "fish-eye", el cual curva la imagen. Este tipo de distorsiones pueden ser resueltas mediante geometría proyectiva, donde se deforma la imagen para quitar este tipo de distorsiones, aunque cabe destacar que este proceso puede ser realizado con el sentido de deformar una imagen bien proyectada para que tenga el efecto de un lente "fish-eye".



*Fig. 2.28: Distorsión producto de la curvatura en lentes*

Existen otros casos donde se necesita estirar, rotar o girar la imagen, para poder llegar al resultado deseado, en estos casos se utiliza también geometría proyectiva, donde se cambia el punto de vista del observador. Cabe destacar que la información que uno posee para realizar este tipo de algoritmos se encuentra en una dimensión y corresponde a intensidades o colores, por lo que las proyecciones modifican al plano de la imagen de forma global y no exclusivamente a un objeto dentro de la imagen.

### 2.3. Predicción

En esta sección se detalla la teoría sobre el algoritmo ANFIS, utilizado para estimar o predecir la posición a la cual posicionar el arquero para bloquear los tiros.

### 2.3.1. Algoritmo Neuro-Difuso: ANFIS

ANFIS o Sistema Adaptativo de Inferencia Neuro Difusa por sus siglas en inglés (Adaptive Neuro Fuzzy Inference System) es un sistema de procesamiento de información que busca emular el proceso de razonamiento humano por medio de la integración de dos métodos computacionales: FIS y ANN.

Por un lado FIS (Fuzzy Inference System) o Sistema de Inferencia Difusa, trata de representar los sistemas en base a reglas Si-Entonces que pueden ser impuestas de forma subjetiva, mientras que por otra parte las ANN (Artificial Neuronal Network) o Redes Neuronales Artificiales buscan recrear el aprendizaje mediante una gran cantidad de iteraciones que permiten ir realizando ajustes de los coeficientes en una compleja red de nodos. Al combinar estos dos sistemas, podemos ir modificando las reglas y parámetros del sistema difuso en base al aprendizaje que se va logrando a medida de múltiples ajustes.

Entre las ventajas de utilizar ANFIS se señala la posibilidad de representar modelos altamente no lineales, traspasar el conocimiento de los operarios de planta al sistema computacional de forma casi directa, además gracias a su estructura posee cierto grado de transparencia en su interpretación y análisis lo cual contrasta fuertemente con el modelo de caja-negra (donde no se puede saber lo que hay dentro o su composición) cuando se utiliza ANN.

#### *FIS: Fuzzy Inference System*

Los Sistemas de Inferencia Difusa o FIS [17], utilizan la lógica difusa para poder obtener una correspondencia entrada-salida en un sistema dado. La lógica difusa se basa en asignar funciones de probabilidad a conjuntos de expresiones o condiciones que no están bien cuantificadas, por ejemplo expresiones como 'mucho', 'poco', 'alto' o 'bajo'. Por ejemplo, podemos asignar funciones de probabilidad a la temperatura del agua, en donde el sistema quedará definido por variables lingüísticas que serán etiquetas de cada grupo, en este caso los

grupos pueden ser “fría”, “caliente”, además podemos incluir una mayor cantidad de grupos que definen de una forma más exacta al sistema, agregando los prefijos ’muy’, ’poco’. Cabe destacar que al ser simplemente etiquetas, no afectan en ninguna medida al funcionamiento del sistema, pero resulta auto explicativo el utilizar nombres que concuerden con el proceso que se está realizando.

Cada uno de los conjuntos tiene asociado una función de pertenencia, la cual indica en qué medida la señal de entrada forma parte del grupo (figura 2.29). Estas funciones entregan un valor entre 0 y 1, siendo 0 el indicativo de que la señal evaluada no pertenece al conjunto y 1 la completa pertenencia. Estas funciones pueden tomar distintas formas, pero todas cumplen con la condición antes mencionada.

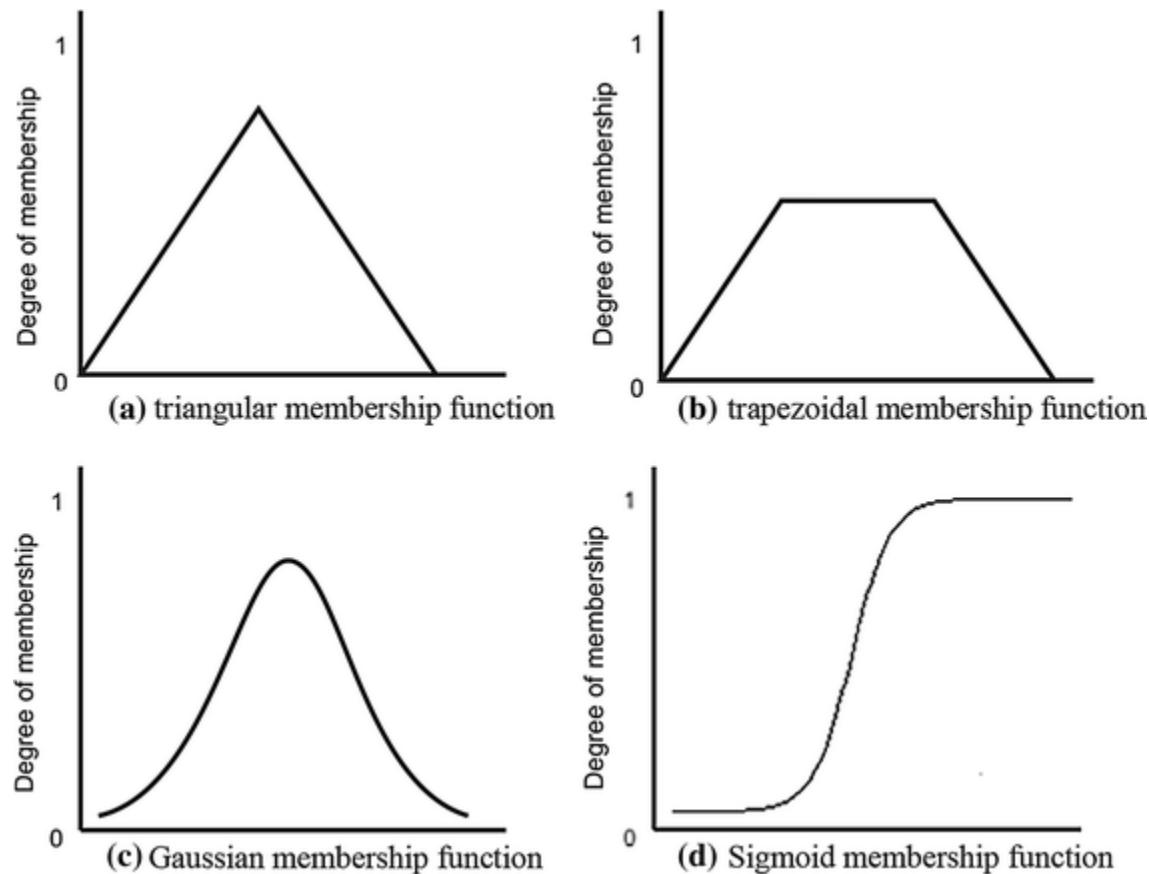


Fig. 2.29: Tipos de funciones de membresía

Para obtener una salida en base al resultado de los conjuntos, se utilizan reglas condicionales SI-ENTONCES, en donde se define el procedimiento a seguir cuando se pertenece a un determinado grupo. Por ejemplo, considerando el caso de la temperatura del agua, donde se tiene una entrada (temperatura del agua) y dos grupos o conjuntos posibles (fría y caliente), se podrían establecer distintos tipos de reglas ante un sistema de regulación de la temperatura del agua por medio de la combustión de gas:

- Si el agua está caliente, entonces apagar el gas.
- Si el agua está fría, entonces prender el gas.

El método anterior de expresar el sistema en base a conceptos lingüísticos corresponde al modelo de Mandani, pero también existen otros métodos donde se utilizan expresiones matemáticas como lo es el modelo Takagi-Sugeno, que busca dar un valor exacto a las expresiones evaluadas por las reglas. Si consideramos a 'y1' e 'y2' como el nivel de gas que debe tener el sistema a una cierta temperatura, se logran dos comportamientos diferentes, y dependientes del valor de entrada:

- Si el agua está caliente, entonces  $y1 = a*T^\circ + c$
- Si el agua está fría, entonces  $y2 = b*T^\circ + d$

La lógica difusa utiliza distintos operadores lógicos cuando se tienen múltiples entradas. Por ejemplo, la regla puede hacer referencia a dos situaciones que deben ocurrir simultáneamente (operador AND de conjunción), o bien que no necesariamente ocurran ambas en el mismo instante (operador OR de disyunción). Además se señala el operador de negación (operador NOT) el cual es utilizado para señalar la no inclusión del conjunto. Existen distintas formas de utilizar estos operadores como se muestra en la tabla 2.1, donde se muestran los métodos de Zadeh y probabilístico:

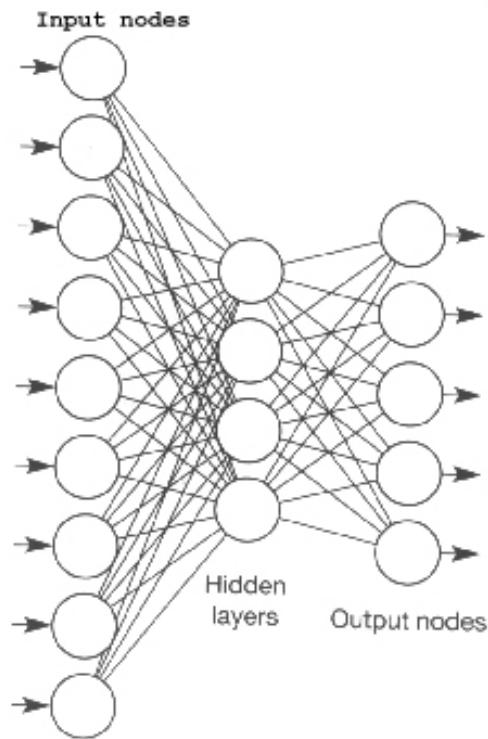
*Tab. 2.1: Tabla de lógica difusa*

Método	A and B	A or B	not A
Zadeh	$\text{Min}(\mu_A, \mu_B)$	$\text{Max}(\mu_A, \mu_B)$	$1 - \mu_A$
Probabilístico	$\mu_A * \mu_B$	$\mu_A + \mu_B - \mu_A * \mu_B$	$1 - \mu_A$

Donde  $\mu_A$  y  $\mu_B$  corresponden a los valores de las funciones de membresía en las entradas A y B.

#### *ANN: Artificial Neuronal Network*

Las Redes Neuronales Artificiales o ANN [18], son una herramienta computacional que en un principio buscaba imitar al cerebro humano, actualmente están siendo aplicadas en distintas áreas como visión por computador, diagnóstico médico, video juegos, entre otros. Siendo utilizadas por su habilidad de aprendizaje, el cual se basa en una etapa de entrenamiento de la red. Este proceso de aprendizaje toma una señal de entrada, se procesa dentro de la red y entrega un resultado, si este valor final no es el esperado, se modifican los valores internos de la red para mejorar el sistema. Este procedimiento es realizado con una gran cantidad de datos los cuales deben ser correctamente etiquetados con su entrada y valor deseado para poder evaluar la mejora de la red.



*Fig. 2.30: Red Neuronal y sus capas*

Las redes están compuestas por tres capas principales, una de entrada, capa oculta y la de salida (figura 2.30), cada capa puede poseer múltiples nodos que se interconectan con los nodos de la siguiente capa. En la conexión de nodos existen pesos que amplifican la señal transportada. En las distintas capas los nodos poseen una función de activación (figura 2.31) que modifica la señal entrante y la entrega en su salida, estas funciones de activación mantienen su identidad o forma a medida que el sistema se entrena, pero los pesos entre nodos así como también la velocidad de aprendizaje van siendo modificados con el fin de mejorar el sistema.

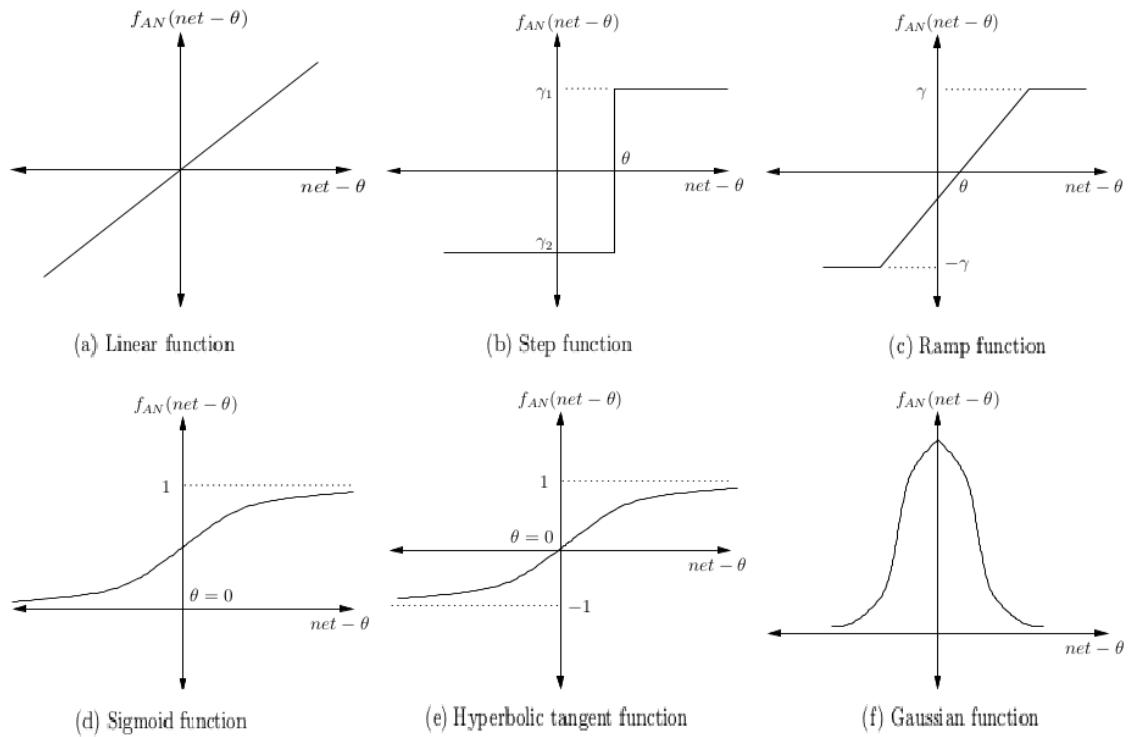


Fig. 2.31: Algunas funciones de activación de las Redes Neuronales

La velocidad de aprendizaje o "Learning Rate" de una Red Neuronal, hace referencia a la proporción en que serán modificados los pesos a medida que el sistema aprende. Como en un comienzo la red no posee una estructura ideal, se espera que el error entre la entrada y su salida deseada sea considerable, por lo que la proporción en que se modificarán los pesos de la red será grande, no obstante, a medida que el sistema aprenda los resultados irán mejorando, por lo cual, cambios grandes en la estructura interna pueden producir oscilaciones en torno al punto óptimo. Es por esta razón que el "learning rate" debe ser modificado e ir disminuyendo en el transcurso del proceso (figura 2.32).

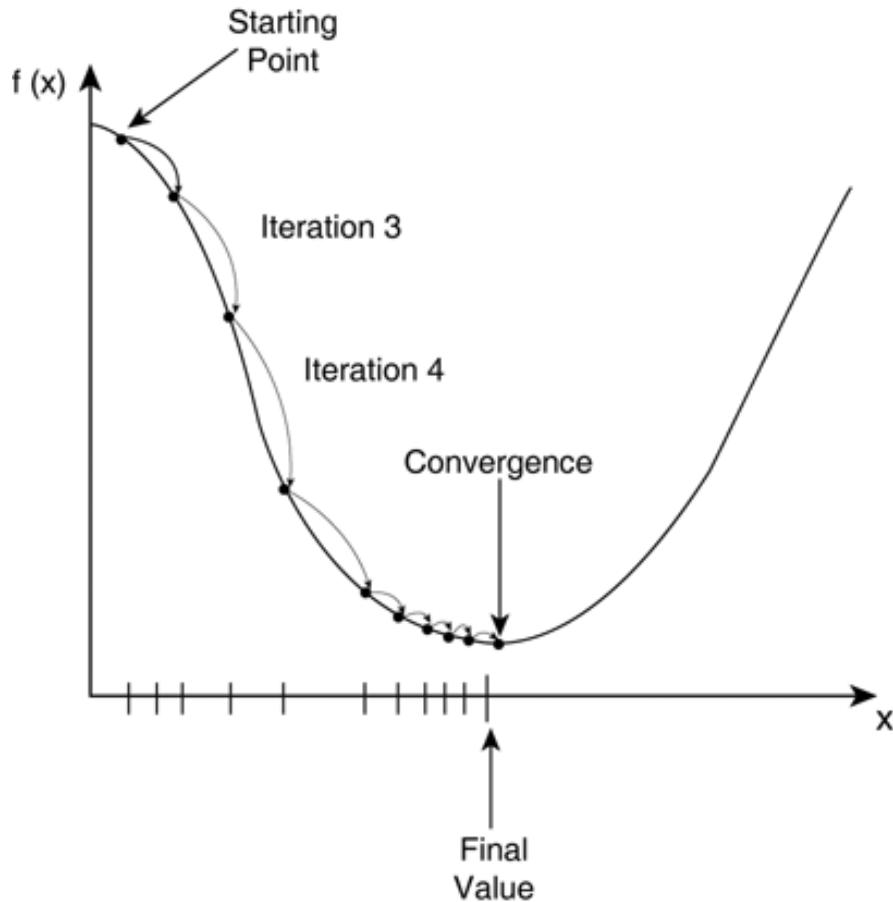


Fig. 2.32: Learning rate: Iteraciones hasta llegar al mínimo error posible

La forma de actualizar los pesos de la red para que el sistema mejore su desempeño, es mediante la utilización del gradiente del error sobre el parámetro a modificar, permitiendo saber la dirección en la cual el error disminuye, e ir ponderando por una ganancia que va disminuyendo a medida que se acerca al valor de mínimo error, método conocido por gradiente descendente.

$$\theta_1 = \theta_1 - \alpha * \nabla J(\theta_1) \quad (2.1)$$

Donde  $\theta_1$  corresponde a la variable de peso a modificar,  $\alpha$  al factor de ajuste que irá disminuyendo a medida que la función minimiza su error.  $J(.)$  corresponde a la función de error con respecto a la salida deseada.  $\nabla J(\theta_1)$  es el gradiente de la función, y su valor apuntará

en la dirección del mayor cambio, por lo que el ajuste realizado al peso, debe ir en dirección contraria para lograr reducir el error, lo cual corresponde al signo menos de la ecuación.

### *ANFIS: Adaptive Neuro Fuzzy Inference System*

Como se mencionó anteriormente, ANFIS [19] mezcla las herramientas previamente descritas. Por una parte toma la metodología FIS para plantear sistemas en base a reglas y por otro lado, utiliza los métodos de aprendizaje de las ANN para mejorar el desempeño ajustando su estructura. Como resultado, se obtiene que los sistemas basados en reglas que propone la lógica difusa, se pueden ir adaptando y mejorando mediante la utilización de sets de datos que establecen cómo debería ser la respuesta del sistema ante una entrada específica.

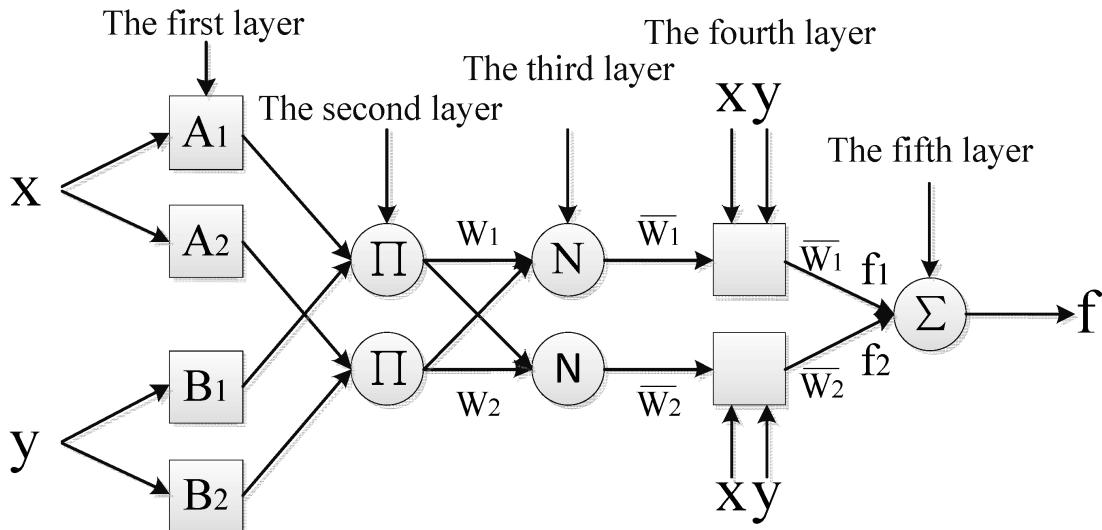


Fig. 2.33: Estructura de un sistema ANFIS Takagi-Sugeno

La arquitectura ANFIS (figura 2.33) comúnmente utiliza el modelo difuso de Takagi-Sugeno la cual posee 5 capas con distintos tipos de nodos cada una. Los nodos toman la información de las entradas, la procesan y luego la entregan a las siguientes capas. A diferencia de una red neuronal, ANFIS no posee pesos o ganancias en las conexiones entre nodos.

- Capa 1: En esta capa se encuentran los nodos que poseen las funciones de membresía, estas determinan el sentido de pertenencia de las entradas sobre los conjuntos. Cuando las señales de entrada llegan a esta capa, estas se propagan a los nodos de sus conjuntos correspondientes, y son evaluadas con un valor entre 0 y 1 dependiendo del valor de pertenencia. Cabe destacar que esta capa está sujeta al aprendizaje lo que significa que los valores de las funciones se irán modificando a medida que se entrena la red. Los parámetros de la funciones de membresía de esta capa son llamados parámetros de la premisa, los cuales definen las formas de estas funciones.
- Capa 2: Esta capa posee nodos fijos, es decir, no cambian su forma de comportamiento una vez establecidos. La función que desempeñan estos nodos es dependiente de las reglas del sistema. Como se ha mencionado antes, pueden ser utilizadas distintas reglas lógicas en los sistemas difusos (AND y OR) que relacionan las salidas de la capa anterior. El propósito de esta capa es la de evaluar las señales en el conjunto de reglas, obteniendo mayor o menor valor dependiendo de la fuerza con la que participan. El número de nodos de esta capa es definida por la cantidad de reglas del sistema.
- Capa 3: Los nodos de esta capa, al igual que antes corresponden a nodos fijos, los cuales poseen la función de ponderar los grados de cumplimiento de cada una de las reglas, según sea el resultado de los nodos de la capa anterior.
- Capa 4: En esta capa pueden o no utilizarse parámetros consecuentes. Estos son valores que añaden un comportamiento matemático a cada regla. Dependiendo de la expresión, se puede tratar de un modelo Takagi-Sugeno de orden 0, 1, etc. Cada nodo posee una función matemática que puede tomar las variables de entrada del sistema para su evaluación, el objetivo es tener un comportamiento diferente dependiendo del grado con que se cumplan las reglas.

- Capa 5: En esta capa los nodos realizan una sumatoria de los valores previos. El resultado corresponde a la salida del sistema ANFIS, el cual puede tener una o múltiples salidas.

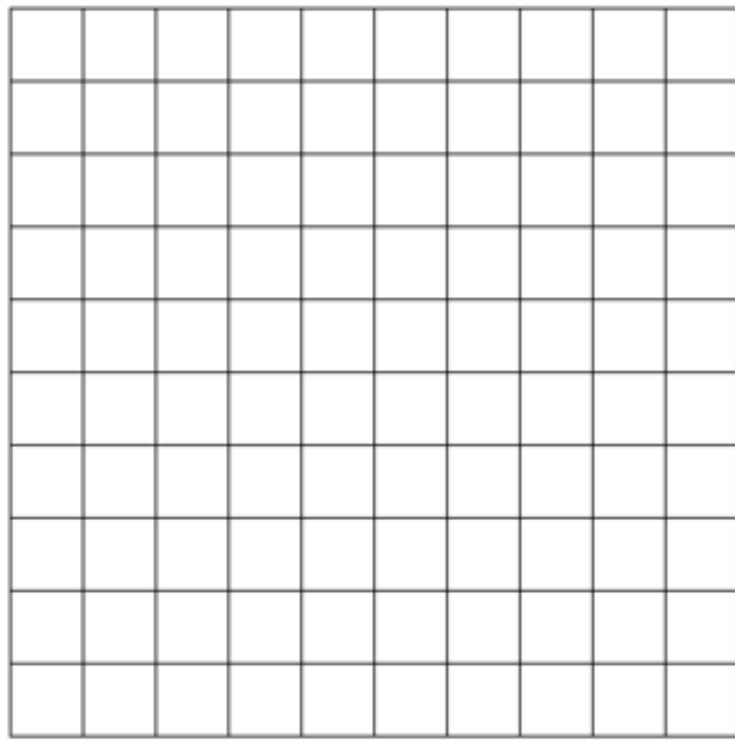
Se destaca que sólo las capas 1 y 4 poseen parámetros que pueden ser modificados en la etapa de aprendizaje. Esto sólo afecta a los valores numéricos que poseen las funciones de membresía y las expresiones matemáticas de los parámetros consecuentes, por lo que no son modificadas las estructuras, ni el orden de estas expresiones respectivamente.

En cuanto a las funciones de membresía, estas pueden tener distintos tipos al igual que los sistemas FIS (figura 2.29), donde las más comunes corresponden a las gaussianas, trapezoidales, triangulares, y sigmoidales. Cada una de ellas presenta ventajas y desventajas, siendo su uso definido por el programador y las condiciones del sistema.

### *Iniciación de ANFIS*

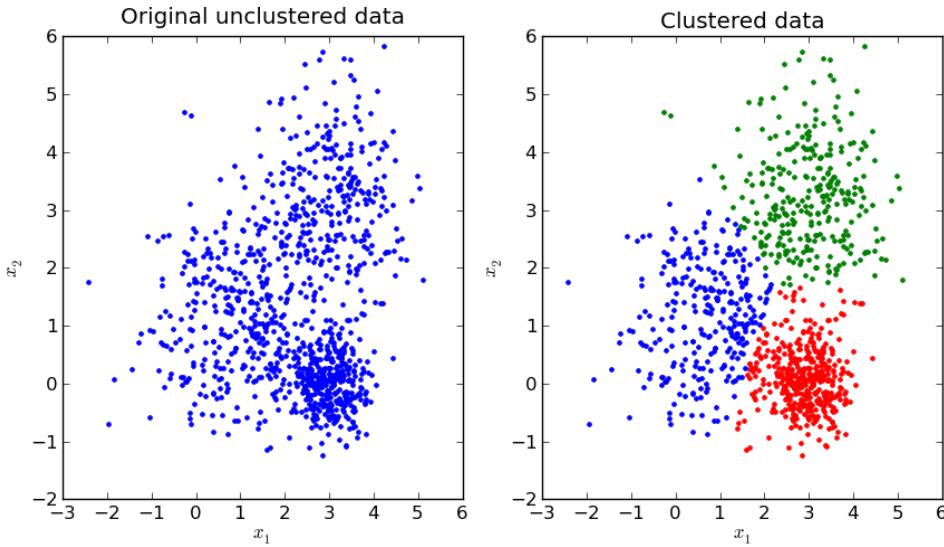
Existen dos métodos para definir la arquitectura de la red ANFIS, generando la cantidad de reglas y funciones de membresía que tendrá la red, lo cual a su vez, definirá el comportamiento de esta.

- Partición de red (“Grid Partition”): El algoritmo divide el rango de valores de entrada de manera uniforme, creando grupos hasta que cierta cantidad de funciones de membresía es alcanzado. Dado que necesita de la evaluación del rango de las entradas, este método debe contar previamente con los datos de entrenamiento. Un inconveniente de este método es que la cantidad de reglas que se genera, corresponde a la combinación entre todos los grupos, generando de forma exponencial  $2^n$  reglas (figura 2.34).



*Fig. 2.34: Método de “Grid Partition”*

- Agrupación sustractiva (“SubClustering o Subtractive Clustering”): Este método a diferencia del anterior, toma los valores de las señales de entrada y los va agrupando de acuerdo a ciertos parámetros, uno de ellos es el rango de influencia, el cual determina el porcentaje del rango de la señal de entrada a ser cubierta por la función de membresía, un mayor valor genera grupos más anchos y con menos reglas mientras que un menor valor, establece funciones angostas con una mayor cantidad de reglas por entrada. A diferencia del método anterior, este genera la misma cantidad de reglas que de grupos reduciendo la carga computacional para una gran cantidad de entradas. (figura 2.35).



*Fig. 2.35: Método “SubClustering”*

### *Error Backpropagation*

Para que el sistema mejore (y al igual que en las redes neuronales) son necesarios métodos de aprendizaje, el algoritmo de “Error Backpropagation” utiliza un set de datos etiquetados, donde se especifica una entrada y la salida deseada de la red, luego mediante métodos matemáticos se modifican las cualidades internas de la red que van logrando disminuir el error entre la salida obtenida y la deseada.

Este método señala que el error existente en la salida, proviene las etapas previas del sistema. Se utiliza el error entre, la salida que ha sido recién procesada ante una entrada y la salida deseada, esta diferencia es utilizada para ajustar las funciones de membresía que posee la red ANFIS mediante pequeños cambios en los valores de los parámetros internos. Los ajustes realizados son en base al método del gradiente descendente, donde en cada iteración, el ajuste de parámetros lleva al sistema en dirección del mínimo valor de error. Cabe destacar que para la utilización de éste método, se deben dejar los parámetros consecuentes (capa 4) con valores fijos durante el proceso.

El error en la salida del sistema puede ser expresado de la siguiente forma:

$$J = \frac{1}{2} \sum_1^N e_k^2 \quad \text{con} \quad e_k = y_k^* - y_k \quad (2.2)$$

Donde  $J$  corresponde al error acumulado en la salida del sistema,  $e_k$  es el error por cada set de datos, es decir, la diferencia entre la salida deseada  $y_k^*$  y la salida del sistema  $y_k$ ,  $k$  el número del par entrada-salida ( $k = 1, 2, 3, \dots, N$ ),  $N$  la cantidad total de datos utilizados para el entrenamiento.

Para actualizar los parámetros del sistema, se utiliza como ejemplo el método del gradiente de primer orden, con  $w$  como el vector de variables que se espera ir mejorando con el entrenamiento, el parámetro  $\alpha$  un ajuste de velocidad para mejorar el sistema y  $\nabla J(w)$  el jacobiano de la red.

$$w(n+1) = w(n) - \alpha \nabla J(w(n)) \quad (2.3)$$

Siendo el Jacobiano, las derivadas parciales del error acumulado con respecto a las variables asociadas. Dado que el sistema puede contener múltiples salidas, es preciso utilizar el Jacobiano en lugar del Gradiente.

$$\nabla J(w(n)) = \left[ \frac{\partial J(w)}{\partial w_1}, \frac{\partial J(w)}{\partial w_2} \dots \frac{\partial J(w)}{\partial w_M} \right]^T \quad (2.4)$$

Utilizando  $\mu_{ij}(x)$  como representación de la función de membresía para el caso de una función Gaussiana:

$$\mu_{ij}(x_j, c_{ij}, \sigma_{ij}) = \exp\left(-\frac{(x_j - c_{ij})^2}{2\sigma_{ij}^2}\right) \quad (2.5)$$

Con  $j$  el valor de entrada e  $i$  el grupo al que hace referencia de la entrada  $j$ .

Dado que en la capa 2 de la red y capa 3, se multiplican los resultados de las funciones

de membresía y luego se realiza un promedio, esto se puede expresar de la forma:

$$\gamma_i(x) = \frac{\prod_{j=1}^P \exp\left(-\frac{(x_i - c_{ij})^2}{2\sigma_{ij}^2}\right)}{\sum_{i=1}^K \prod_{j=1}^P \exp\left(-\frac{(x_i - c_{ij})^2}{2\sigma_{ij}^2}\right)} \quad (2.6)$$

Siendo  $P$  la cantidad total de funciones de membresía para la entrada  $j$  y  $K$  la cantidad de reglas dentro de la red.

Con esto en cuenta, se puede expresar la salida de la red como una suma por producto para el caso donde los parámetros consecuentes poseen una estructura lineal.

$$y = \sum_{i=1}^K \gamma_i(x)(a_i^T x + b_i) \quad (2.7)$$

Pero dado que para el método de “Error Backpropagation” los parámetros consecuentes se mantienen fijos su valor corresponderá a una constante, resultando:

$$y = \sum_{i=1}^K \gamma_i(x)d_i \quad (2.8)$$

Finalmente con las derivadas parciales y el uso de la regla de la cadena:

$$\frac{\partial J}{\partial d_i} = \frac{\partial J}{\partial e} \frac{\partial e}{\partial d_i} = -\gamma_i e, \quad i = 1, 2, 3 \dots K \quad (2.9)$$

Con la ecuación de actualización:

$$b_i(n+1) = b_i(n) + \alpha \gamma_i e, \quad i = 1, 2 \dots K \quad (2.10)$$

De igual forma para los demás:

$$c_{ij}(n+1) = c_{ij} + 2\alpha(n)\gamma_i e [b_i - y] \frac{x_j - c_{ij}}{\sigma_{ij}^2}, \quad i = 1, 2 \dots K, \quad j = 1, 2 \dots P \quad (2.11)$$

$$\sigma_{ij}(n+1) = \sigma_{ij}(n) + 2\alpha(n)\gamma_i e[d_i - y] \frac{(x_i - c_{ij})^2}{\sigma_{ij}^2}, \quad i = 1, 2, \dots, K, \quad j = 1, 2, \dots, P \quad (2.12)$$

### Mínimos Cuadrados Recursivo

Para lograr ajustar los valores de los parámetros consecuentes de la capa 4 de la red ANFIS, se mantienen fijos los valores de las funciones de membresía, con lo cual se procede a utilizar el método matemático de Mínimos Cuadrados Recursivos. Dado que en este caso el error es propagado hacia delante, también se le llama un método “Error Forwardpropagation”.

Dado que las funciones de membresía son consideradas invariantes, estas son modeladas como pesos en el sistema. Considerando una red con dos reglas ( $f_1$  y  $f_2$ ), cada regla proporciona una función diferente, quedando con una forma:

$$Q_1 = \bar{w}_1 f_1 + \bar{w}_2 f_2 \quad (2.13)$$

Donde  $Q_1$  corresponde a la salida para el primer set de datos,  $\bar{w}_i$  al peso entregado por la regla  $i$  al realizar un promedio de las funciones de membresía y  $f_i$  la función de parámetros consecuentes que entrega cada regla. al expresar el sistema de forma expandida tenemos:

$$f_1 = a_1 x + b_1 y + r_1 \quad (2.14a)$$

$$f_2 = a_2 x + q_2 y + r_2 \quad (2.14b)$$

Quedando para el primer set de datos, y luego de forma general:

$$Q_1 = \bar{w}_1(a_1 x_1 + b_1 y_1 + r_1) + \bar{w}_2(a_2 x_1 + q_2 y_1 + r_2) \quad (2.15a)$$

$$Q_1 = (\bar{w}_1 x_1)a_1 + (\bar{w}_1 y_1)b_1 + (\bar{w}_1)r_1 + (\bar{w}_2 x_1)a_2 + (\bar{w}_2 y_1)q_2 + (\bar{w}_2)r_2 \quad (2.15b)$$

$$Q_i = (\bar{w}_1 x_i)a_1 + (\bar{w}_1 y_i)b_1 + (\bar{w}_1)r_1 + (\bar{w}_2 x_i)a_2 + (\bar{w}_2 y_i)q_2 + (\bar{w}_2)r_2 \quad (2.15c)$$

Con el valor de  $i$  desde 1 a  $P$ , siendo  $P$  el largo del set de datos de entrenamiento a la red.

Cabe decir que se necesitan igual cantidad de funciones que de variables para que el sistema de ecuaciones tenga solución, aunque generalmente se cuenta con una mayor cantidad de datos que de ecuaciones. El sistema puede ser expresado de forma matricial como se muestra a continuación.

$$A\theta = y \quad (2.16)$$

Donde  $\theta$  corresponde a datos ya conocidos, siendo un vector  $M \times 1$  donde  $M$  es el número de parámetros consecuentes (6 en este ejemplo),  $A$  son las variables a buscar, siendo una matriz  $P \times M$  con  $P$  el largo del set de datos de entrenamiento e  $y$  el vector de salida deseados del sistema de tamaño  $P \times 1$ :

$$A = \begin{bmatrix} (\bar{w}_1x)_1 & (\bar{w}_1y)_1 & (\bar{w}_1)_1 & (\bar{w}_2x)_1 & (\bar{w}_2y)_1 & (\bar{w}_2)_1 \\ (\bar{w}_1x)_2 & (\bar{w}_1y)_2 & (\bar{w}_1)_2 & (\bar{w}_2x)_2 & (\bar{w}_2y)_2 & (\bar{w}_2)_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ (\bar{w}_1x)_p & (\bar{w}_1y)_p & (\bar{w}_1)_p & (\bar{w}_2x)_p & (\bar{w}_2y)_p & (\bar{w}_2)_p \end{bmatrix}, \theta = \begin{bmatrix} a_1 \\ b_1 \\ r_1 \\ a_2 \\ b_2 \\ r_2 \end{bmatrix}, y = \begin{bmatrix} Q_1 \\ Q_2 \\ \vdots \\ \vdots \\ Q_p \end{bmatrix} \quad (2.17a)$$

Buscando los parámetros de la matriz  $\theta$ , se busca minimizar el error de la expresión  $\|A\theta - y^2\|$ . Utilizando una pseudo-inversa de  $\theta$ :

$$\theta^* = (A^T A)^{-1} A^T y \quad (2.18)$$

Donde  $A^T$  es la traspuesta de  $A$  y si no es singular (matriz cuadrada con determinante igual a cero), se tiene que  $(A^T A)^{-1}$  corresponde a la pseudo-inversa de  $A$ . Finalmente el

método Recursivo de Mínimos Cuadrados queda expresado como el sistema:

$$\theta_{i+1} = \theta_i + P_{i+1}a_{i+1}(y_{i+1}^T - a_{i+1}^T\theta_i) \quad (2.19a)$$

$$P_{i+1} = P_i - \frac{P_i + a_{i+1}a_{i+1}^T P_i}{1 + a_{i+1}P_i a_{i+1}}, \quad i = 1, 2, \dots, P-1 \quad (2.19b)$$

Donde  $a_i^T$  es el vector fila  $i$  de la matriz  $A$ ,  $y_i$  es el elemento  $i$  del vector columna  $y$ ,  $P_i$  es la matriz de covarianza definida por  $P_i = (A^T A)^{-1}$ .

## 2.4. Actuación

En esta sección se describe la teoría detrás de los motores DC, y de los controladores PID que serán utilizados para lograr manipular el movimiento del arquero.

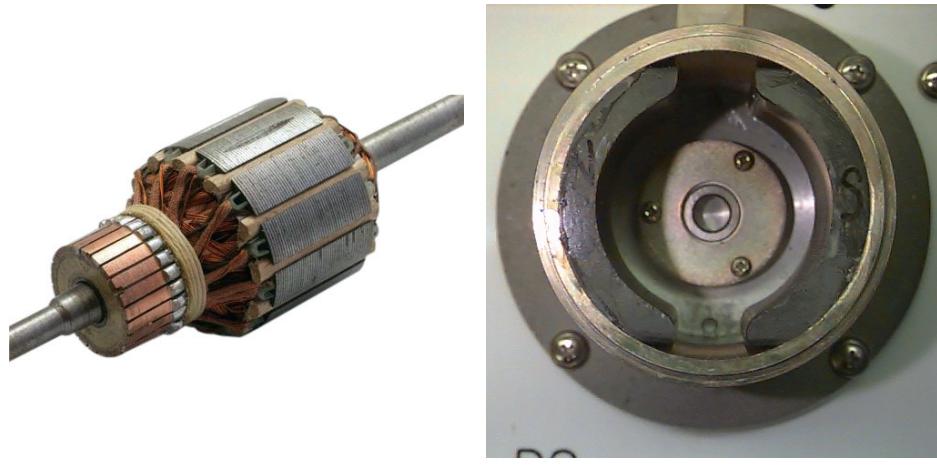
### 2.4.1. Motores DC

Los motores de corriente directa o motores DC (Direct Current) [20], también llamados de corriente continua, han sido utilizados en una gran cantidad de aplicaciones que requieran de la conversión eléctrica-mecánica y viceversa, generando un movimiento rotatorio producto del campo magnético.



*Fig. 2.36:* Motor DC con escobillas

La estructura de un motor DC con escobillas consta de dos partes fundamentales, el estator que corresponde a una sección inmóvil y el rotor que rota sobre su eje (figura 2.37). El estator compuesto de imanes o bien devanados de cobre con núcleos de hierro, establecen polos magnéticos constantes. El rotor generalmente posee múltiples devanados de cobre con núcleo de hierro, en estos se hace circular una corriente que produce un campo magnético a medida que van siendo activados.



(a) Rotor

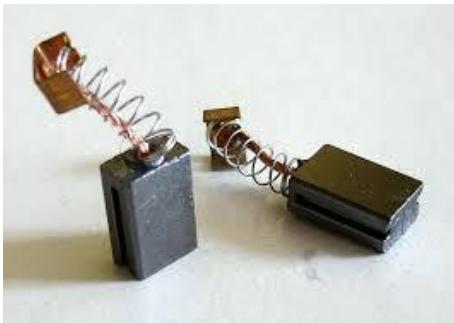
(b) Estator

Fig. 2.37: Rotor y estator de un motor DC con escobillas

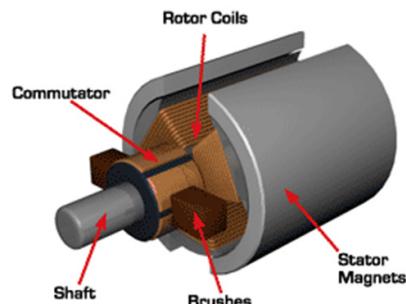
El movimiento es generado por la atracción y repulsión de los campos magnéticos generados por el rotor y estátor. Cuando se hace circular una corriente por el rotor, se genera un campo que trata de alinearse con el existente en el estátor, el cual permanece siempre constante en su sitio. Al aumentar la corriente circulante, la fuerza con que el rotor gira también aumentará lo que se explica con la ley de Lorentz. Por otra parte, la velocidad que puede alcanzar el motor está determinada por el voltaje aplicado al sistema, y su dirección de giro será dependiente de las direcciones de los campos generados, lo cual a su vez se asocia con la dirección de la corriente circulante.

El nombre que se les dá a estos motores “con escobillas”, es producto de la conexión directa que tiene el circuito eléctrico para inyectar corriente a los devanados del rotor. Antiguamente se utilizaban escobillas metálicas, pero producto de su bajo nivel de contacto entre los terminales, se procedió a utilizar carboncillo como material conductor, que mediante un sistema de resortes logran dar presión con el rotor, lo cual permite mantener en contacto las superficies (figura 2.38). El principal inconveniente del carboncillo es el desgaste que este tiene al estar en constante fricción con una superficie rotatoria, produciendo que los restos del material se acumulen al interior de los motores generando chispas que pueden dañar los

componentes. Producto de lo anterior, los motores DC con escobillas requieren de mantenimiento periódico y se encuentran propensos a fallas por desgaste en el tiempo.



(a) Escobillas



(b) Conjunto del motor DC

Fig. 2.38: Sistema de un motor DC con escobillas

Una alternativa a los motores DC con escobillas (Brushed DC Motor) son los motores DC sin escobillas (Brushless DC Motor), los cuales poseen un sistema diferente de funcionamiento. Es el rotor el que posee imanes permanentes y es el estator el que cuenta con devanados y núcleos de hierro (figura 2.39). Mediante la aplicación de corriente a un devanado del estator, el rotor tratará de alinear los polos magnéticos produciendo la rotación esperada, al no circular corriente por el rotor, es posible alimentar al sistema sin necesidad de recurrir a las escobillas. La condición que este tipo de estructura posee, es que se necesita saber la orientación del rotor con los imanes permanentes en todo momento, ya que se necesita energizar al devanado correcto para producir el movimiento deseado, es por esto que se incurre a sensores de efecto Hall que miden el campo magnético, lo cual permite tener control sobre la rotación del motor.

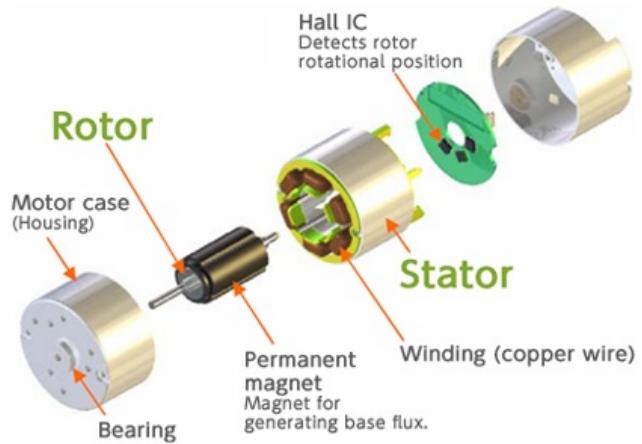


Fig. 2.39: Motor DC sin escobillas

Para poder hacer un análisis comparativo o bien, lograr simular un motor DC con escobillas, es necesario conocer el modelo matemático de estos (figura 2.40), a continuación se presenta una descripción de los componentes:

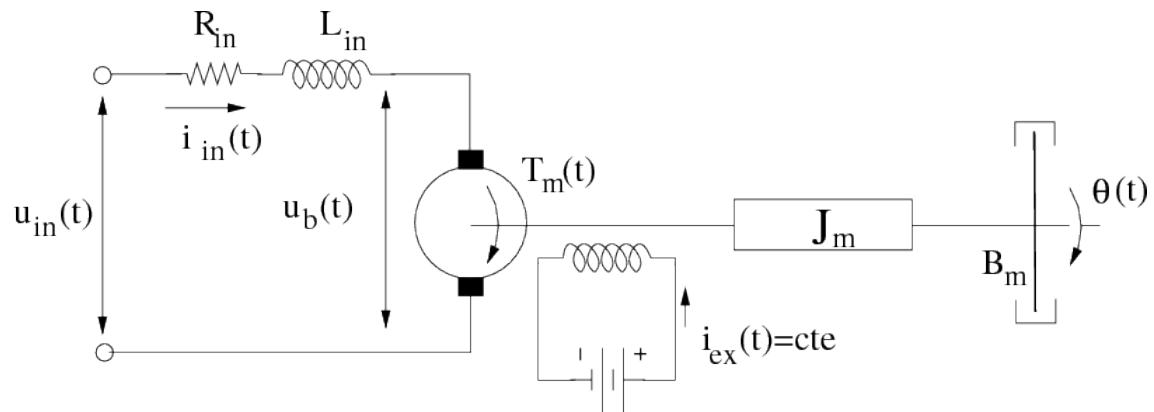


Fig. 2.40: Modelo de un motor DC con escobillas con circuito de campo

Tab. 2.2: Parámetros de un motor DC

Variable	Significado
$u_{in}(t)$	Voltaje aplicado
$i_{in}(t)$	Corriente de armadura
$R_{in}$	Resistencia de armadura
$L_{in}$	Inductancia de armadura
$u_b(t)$	Fuerza contra-electromotriz
$\omega(t)$	Velocidad angular del motor
$\theta(t)$	Posición angular del motor
$K_b$	Constante de la fuerza contra-electromotriz
$K_i$	Constante de torque
$T_m(t)$	Torque del motor
$T_L(t)$	Torque de carga
$J_m$	Inercia del rotor
$B_m$	Coeficiente de fricción

Los parámetros de armadura corresponden a los que se encuentran dentro del rotor, la resistencia e inductancia provienen del embobinado de cobre. La variable de fuerza contra-electromotriz es proporcional a la velocidad de rotación del motor produciendo un voltaje en los terminales. El circuito de excitación de campo puede ser modelado por una inductancia ( $L_{ex}$ ) y una resistencia ( $R_{ex}$ ), este circuito modela el campo generado por los imanes permanentes en el estator o bien el embobinado externo para producir un campo magnético. El momento de inercia del rotor, puede ser considerado conjuntamente con la carga, siendo sumado a este ya que ambos generan resistencia al movimiento, siempre y cuando se tenga un sistema rígido, sin elasticidad en los mecanismos que alteren el comportamiento.

Finalmente las expresiones matemáticas que modelan al sistema son:

$$u_{in}(t) = R_{in} * i_{in}(t) + L_{in} \frac{di_{in}(t)}{dt} + u_b(t) \quad (2.20a)$$

$$T_m(t) = k_i * i_{in}(t) \quad (2.20b)$$

$$u_b(t) = \frac{1}{K_b} \omega(t) \quad (2.20c)$$

$$T_m(t) - T_L(t) = J_m \frac{d\omega(t)}{dt} + B_m * \omega(t) \quad (2.20d)$$

Donde aplicando la transformada de Laplace se obtiene:

$$U_{in}(s) = (R_{in} + L_{in} * s) * I_{in}(s) + U_b(s) \quad (2.21a)$$

$$T_m(s) = k_i * I_{in}(s) \quad (2.21b)$$

$$U_b(s) = \frac{1}{K_b} \Omega(s) \quad (2.21c)$$

$$T_m(s) - T_L(s) = (J_m * s + B_m) \Omega(s) \quad (2.21d)$$

Y finalmente se obtiene el siguiente modelo de bloques, expresado en el dominio de frecuencia

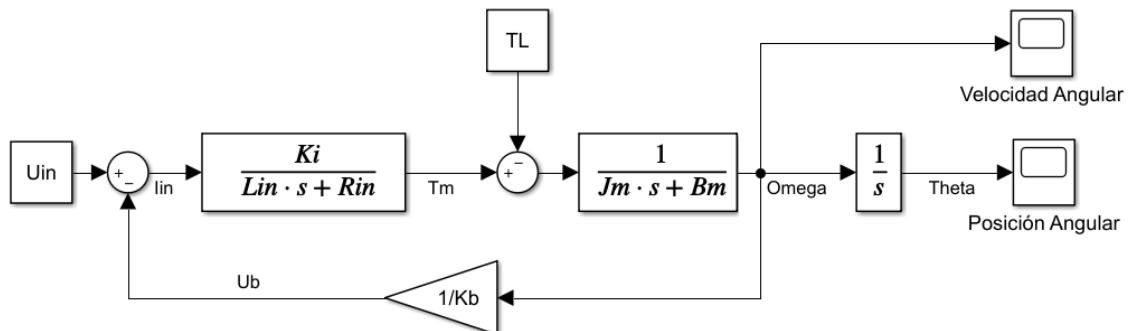


Fig. 2.41: Modelo de un motor DC expresado en frecuencia

Se menciona además que las constantes  $K_i$  y  $K_b$  son el inverso la una de la otra.

#### 2.4.2. Controlador PID

Los sistemas de control pueden ser estructurados de distintas formas, dadas las necesidades de este estudio sólo se implementará un sistema de una entrada y una salida SISO (“Single Input Single Output”) con una lazo de realimentación. Este sistema será el encargado de controlar la posición del motor dada alguna referencia, como se muestra en la figura 2.42, el sistema posee un controlador  $C(s)$ , una planta o proceso  $G(s)$  y un sistema de realimentación  $F(s)$ , el cual puede ser un sensor que mide la señal de salida.

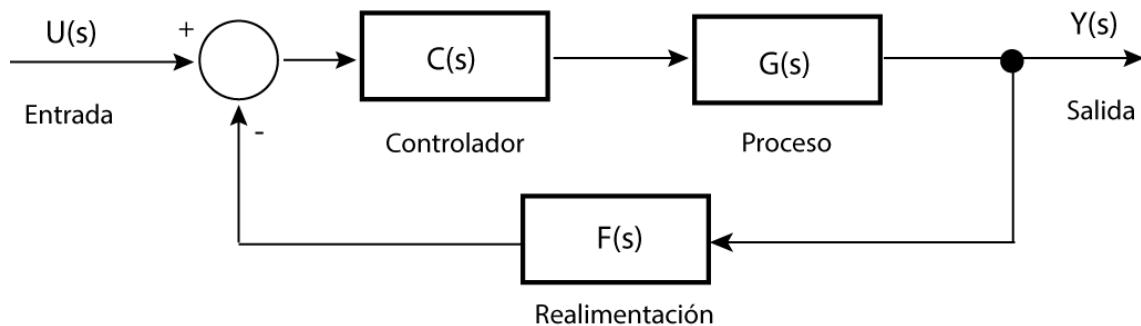


Fig. 2.42: Control de proceso SISO mediante un lazo de realimentación

El controlador PID [21] por sus siglas en inglés Proportional, Integral y Derivative (Proporcional, Integrativo y Derivativo), es un tipo de control de lazo cerrado ampliamente utilizado en la industria por diversos procesos. Permite mantener un control de las funciones que realizan las plantas de operación, mediante el ajuste de ciertos parámetros.

Este controlador utiliza el error entre la señal de referencia (resultado esperado) y la salida actual de la planta para realizar los cálculos matemáticos que llevarán al sistema al punto requerido. Utilizando tres componentes, el controlador irá ajustando la actuación (señal de entrada a la planta), estos componentes poseen su propia función que determinarán el comportamiento de la planta (figura 2.43).

El proceso o también llamada planta, es la operación o mecanismo que queremos controlar. Para efectos de este estudio la planta será el motor DC.

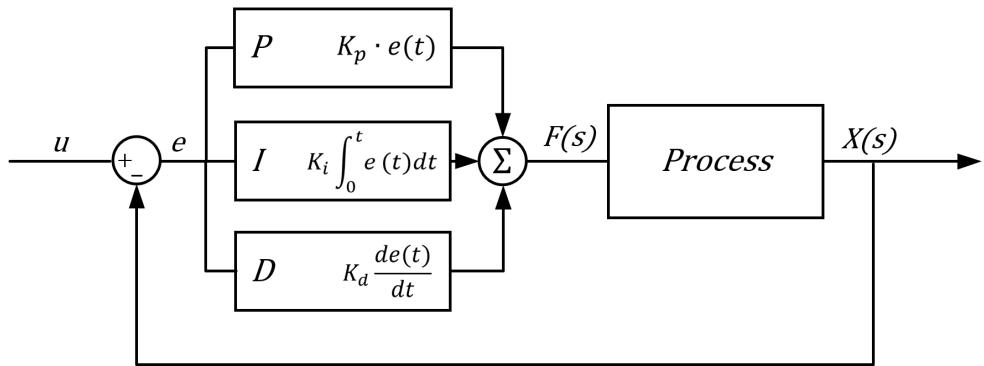


Fig. 2.43: Modelo de un control PID

- Proporcional (P): Este componente añade a la actuación un valor proporcional al error actual. Un valor bajo para el sistema producirá una actuación que no logrará tener mayor repercusión en el comportamiento de la planta. Por el contrario, un valor alto para el sistema puede suponer una actuación exagerada, donde la salida de la planta comience a oscilar bruscamente en torno al punto deseado. Finalmente, un correcto valor para el sistema permite que la salida sea igual o cercana al valor de referencia. Siendo el error pequeño, el componente proporcional no logrará eliminarlo, ya que la actuación también se verá reducida, es por esto que el componente proporcional por sí solo, posee error en estado estacionario.
- Integrativo (I): La parte integrativa del controlador actua de acuerdo a los errores previos del sistema, almacenando los errores anteriores y actuando acorde a ello. Es por esto que la parte integrativa permite la eliminación del error en estado estacionario, ya que por muy pequeño que sea el error, este se irá acumulando en el tiempo y su peso en la actuación también irá creciendo. Uno de los problemas con este tipo de acumulación en el tiempo, son las oscilaciones que se producen en la salida cuando la parte integrativa es significativa, esto es producto de que el valor acumulado no desaparece inmediatamente cuando se llega a la salida deseada y se produce una sobre actuación sobrepasando la referencia. El error al cambiar de signo, comienza a reducir la acu-

mulación almacenada hasta llegar al valor correcto, este fenómeno se le conoce como enrollamiento del lazo.

- Derivativo (D): La componente derivativa es considerado un elemento que predice el comportamiento de la señal, utilizando la derivada o gradiente permite obtener la tendencia de la señal de error y de esta forma prever el cómo actuar a continuación. La desventaja de este componente es que los cambios abruptos en la señal de error producen gradientes de gran valor, lo cual ante sistemas que posean señales con ruido considerable de alta frecuencia se transformará en oscilaciones que impacten negativamente al controlador.

El lazo de control en un controlador PID puede ser estable o inestable, lo que significa que se debe tener cierto conocimiento del sistema cuando se está implementando. La inestabilidad del lazo significa que el sistema, ante una señal acotada tendrá una respuesta no acotada, esto quiere decir que la señal de salida o bien alguna señal interna, aumenten su amplitud rápidamente a medida que pasa el tiempo sin poder recuperar el control, es decir diverge. Este comportamiento puede traer graves problemas dependiendo de la planta, por ejemplo, si es un sistema que controla la intensidad de corriente, la inestabilidad puede provocar fallas en la planta de operación produciendo pérdidas materiales o humanas.

### *Lugar geométrico de raíces o “Root Locus”*

En la teoría de control, el método de lugar geométrico de raíces, entrega información sobre el comportamiento de polos y ceros de una función de transferencia de algún proceso en lazo cerrado al variar la ganancia. El lugar de raíces es una herramienta visual para analizar sistemas dinámicos lineales tipo SISO (Una entrada, una salida. Por sus siglas en inglés: “Single Input Single Output”) y su estabilidad, la cual se logra cuando todos los polos de lazo

cerrado se encuentran en semiplano izquierdo del plano complejo.

En la figura 2.44 se muestra el aspecto de un lazo de control de un proceso SISO, con una planta  $P(s)$ , controlador  $C(s)$  y la ganancia del sensor  $F(s)$ . Las letras  $r$ ,  $y$ ,  $e$ ,  $u$ , corresponden a las señales de referencia, salida, error (entre referencia y salida) y la señal de actuación respectivamente.

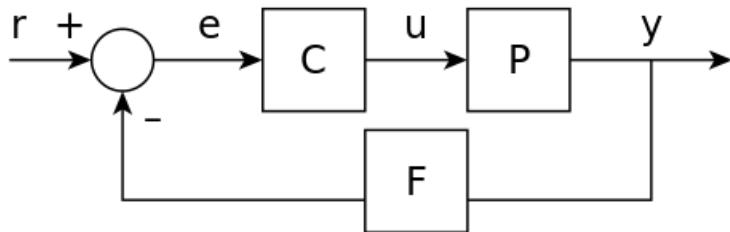


Fig. 2.44: Lazo de control de un proceso

El uso del lugar geométrico de raíces utiliza la ganancia del controlador como única variable a modificar. Siendo este el caso, el controlador puede ser expresado únicamente como una ganancia y el resto de este puede ser incorporado a la planta  $P(s)$ . Si bien este concepto es físicamente incorrecto ya que existe causalidad en cada uno de los bloques, matemáticamente no expresa errores. Dando como resultado la función de transferencia:

$$H(s) = \frac{y(s)}{r(s)} = \frac{KC(s)P(s)}{1 + KF(s)C(s)P(s)} \quad (2.22)$$

Definido el controlador  $C(s)$ , se procede a ver el comportamiento de los polos de lazo cerrado del sistema al aumentar y disminuir la ganancia  $K$ . Los cuales comenzarán a moverse por el plano complejo cambiando de valor, y entregando información del comportamiento que tendrá el sistema ante las señales de entrada. Si los polos se encuentran en el semiplano derecho el sistema será inestable, si existen polos complejos existirán oscilaciones dentro del sistema, entre otras.

En la figura 2.45, se muestra un ejemplo del sistema de transferencia expresado en 2.23,

con ganancia de lazo 2, polos (se muestran con una equis X) en  $-0.5000 + 1.1180i$  y  $-0.5000 - 1.1180i$ , y ceros (se muestran con un circulo O) en  $2.281$  y  $0.2192$ . Los polos se moverán hacia los ceros en este caso, recorriendo las curvas verde y azul.

$$H(s) = 2 \frac{(s + 2,281)(s + 0,2192)}{(s^2 + 2s + 3)} \quad (2.23a)$$

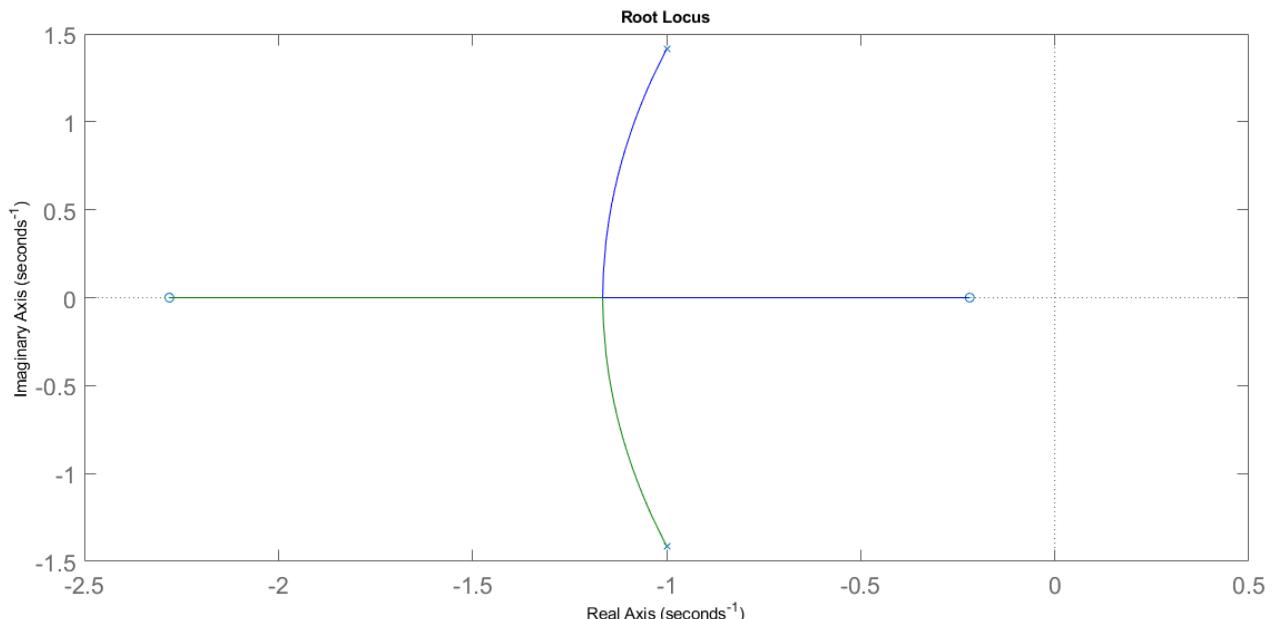


Fig. 2.45: Lugar de raíces de la función de transferencia  $H(s)$

Un sistema que es estable a cierta ganancia, puede dejar de serlo al aumentar o disminuir esta. Pero no solo su estabilidad se verá afectada, sino también la rapidez u oscilaciones según sea el caso.

### 3. METODOLOGÍA

Como se logra apreciar de los capítulos previos, el desarrollo de este estudio posee distintas formas en que puede ser llevado a cabo. En esta sección se menciona la estrategia de solución que se adapta a las condiciones del problema junto con las consideraciones que se deben tomar en cuenta a la hora de implementarlo.

Si bien el proyecto cuenta con una cámara de vídeo, se deben realizar pruebas para determinar el comportamiento de la cámara, corroborar la velocidad de adquisición, deformación de la imagen producto del objetivo, entre otras cosas. Luego, definir la estrategia con que se abordará la confección del algoritmo Neuro-Difuso, así como también la metodología de aprendizaje de este. Y por último, las consideraciones que se utilizarán para desarrollar el diseño del controlador y modelado del motor DC.

#### 3.1. *Adquisición y Procesamiento*

En esta sección, se comenzará detallando el proceso de la adquisición de imágenes así como también su procesamiento, donde se explicarán además las pruebas realizadas sobre cómo afecta la resolución de las imágenes sobre el seguimiento de los objetos de interés para este estudio.

### *3.1.1. Adquisición de imágenes*

Como fue señalado, el proyecto cuenta con una cámara Basler acA800-511uc, con 800 x 600 píxeles para obtener la información de la escena, pero se debe realizar una serie de pruebas para lograr comprobar el funcionamiento del dispositivo, máximas tasas de adquisición, medición del ruido en las imágenes entre otras características.

El primer paso es medir el efecto de utilizar un lente de 4 mm y 6 mm, lo cual entregará información sobre el efecto barril que deforma la imagen en los bordes producto de la distancia focal, además permite establecer la distancia a la que tendrá que ser posicionada la cámara para que el campo de juego efectivo de 0.96 x 0.6 [m<sup>2</sup>] sea totalmente visible, con la consideración de que un lente con una mayor distancia focal, necesitará posicionar a la cámara más lejos del campo de juego, pero se tendrá una menor distorsión del lente.

Luego, se obtendrán imágenes variando los distintos parámetros que posee la cámara comercial haciendo uso del software Pylon Viewer que provee Basler. Se modifica la velocidad de obturación para adquirir las imágenes a una distintas velocidades y ver cómo afecta la iluminación sobre este parámetro. Como punto de partida, se adquieren imágenes a la máxima velocidad de adquisición posible por parte de la cámara, otra a 200 fps, finalmente, una intermedia para lograr realizar una comparación entre ellas, para estas imágenes se tomará en cuenta la máxima resolución, sin luz artificial.

Dado que el manejo de una mayor cantidad de datos puede traer consigo una mayor velocidad de procesamiento, se realizará una prueba bajo las mismas condiciones, cambiando la resolución a 600 x 400 píxeles para ver el comportamiento del dispositivo, y comprobar la tasa máxima que puede ser alcanzada al disminuir la cantidad de píxeles adquiridos a la mitad.

El siguiente paso es comprobar el efecto de la ganancia o sensibilidad ISO que posee la cámara Basler. Para ello se adquieren tres imágenes distintas, a ganancia máxima, mínima y

media, considerando para ellas el uso de iluminación led de un foco de 10 [W] de potencia, dado que a mayores velocidad de adquisición más oscurecidas serán las imágenes. Además se obtendrá una imagen comparativa con el uso de la ganancia máxima pero con iluminación natural.

Estas pruebas permitirán determinar el ruido presente en las imágenes al variar la ganancia, así como también observar la mejor configuración de acuerdo al color observado. Ya que gran parte del ruido que es adquirido proviene del fondo de las imágenes, se considerará el efecto sobre una superficie negra y una blanca, lo cual permitirá corroborar el efecto de las sombras e iluminación sobre estas.

Para la medición del ruido en las imágenes, se considera un sector de 100 x 100 píxeles que debiera presentar un color uniforme sobre la superficie. Esto ayudará a determinar que tanto varían los colores de la imagen ante la variación de los parámetros de ganancia e iluminación. Se utilizará la media y varianza del canal de matiz del espacio de colores HSV con el fin de establecer las condiciones de menor ruido según las respectivas ecuaciones:

$$\bar{X} = \frac{\sum_1^N X_i}{N} \quad (3.1a)$$

$$\sigma^2 = \frac{\sum_1^N (\bar{X} - X_i)^2}{N} \quad (3.1b)$$

### *3.1.2. Procesamiento de las imágenes*

Para obtener la posición de la pelota en juego se mencionaron 4 métodos distintos que contemplan procesar las imágenes adquiridas por la cámara. Uno de los factores principales que se toma en cuenta es el efecto del ruido en la imagen, el cual produce que el tiempo de procesamiento varíe, ya que se debe iterar una mayor cantidad de veces sobre la imagen para lograr asegurar que el objeto encontrado corresponde con el que está siendo buscado.

Tomando en cuenta el efecto del ruido con la velocidad de procesamiento de la imagen, el método de detección de los objetos según el color que estos posean, resulta inmune al ruido siempre y cuando el elemento a buscar posea un color distintivo de los demás objetos en la imagen, ya que aun con la presencia de ruido, el algoritmo de selección no varía su tiempo de ejecución, sea rápido o lento su procesamiento varía de acuerdo al tamaño de la imagen y no del contenido en esta, ya que todos los píxeles son tratados por igual. No así los otros métodos, ya que su tiempo de procesamiento es afectado por el ruido en la imagen, un ejemplo de esto es el algoritmo de detección de bordes, el cual al existir ruido, se detectará una mayor cantidad de objetos de acuerdo a la intensidad de este.

Con el método de obtención del objeto por medio del color, se obtiene como resultado una imagen que muestra los píxeles que se encuentran dentro del rango de colores seleccionado, pero para obtener la posición de la pelota se utiliza el centro de masa con el fin de establecer las coordenadas dentro de la imagen.

$$X_{cm} = \frac{m_1x_1 + m_2x_2 + \dots + m_nx_n}{m_1 + m_2 + \dots + m_n} = \frac{1}{M} \sum_1^N m_i x_i \quad (3.2a)$$

$$Y_{cm} = \frac{m_1y_1 + m_2y_2 + \dots + m_ny_n}{m_1 + m_2 + \dots + m_n} = \frac{1}{M} \sum_1^N m_i y_i \quad (3.2b)$$

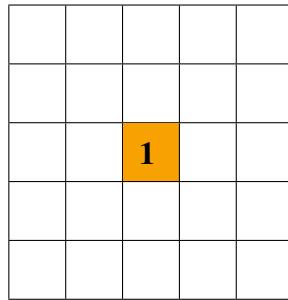
Si bien puede existir ruido en la imagen, este puede ser reducido utilizando un filtro mediana o similar. Para este caso de estudio, se decide utilizar un filtro que se ajusta a la aplicación, utilizando un kernel que suma sus componentes y luego entrega un valor 1 o 0 dependiendo si se supera el umbral, como puede ser apreciado a continuación, donde se utiliza un kernel de 3x3 y un umbral de 7, el cual al ser superado, resulta en un pixel de valor 1.

0	0	1	0	0
1	1	1	1	1
1	1	<b>1</b>	0	0
1	1	0	1	1
0	1	1	1	0

1	1	1	1	1	0	1	0	1
---	---	---	---	---	---	---	---	---

(b) Vector de datos

(a) Elemento central (naranja) y datos colindantes



(c) Resultado del filtro

Tab. 3.1: Proceso de filtrado de un píxel

Para la selección del filtro se utilizarán distintos umbrales considerando un kernel de 3x3 dado el tamaño de la pelota en comparación a las dimensiones del campo de juego, ya que un kernel de mayores dimensiones a pesar de reducir el ruido en una mayor proporción, también afecta al objeto que se está buscando, reduciendo el tamaño y partes de éste. El umbral será determinado al comparar el efecto de la reducción de ruido y tamaño del objeto.

Este procesamiento será llevado a cabo bajo las condiciones que entregaron menor ruido de las pruebas anteriores y considerando máxima resolución de la cámara.

### *3.1.3. Seguimiento v/s resolución*

Un aspecto interesante para el desarrollo del estudio, es el efecto de reducir la resolución de la cámara y comprobar hasta que valores se puede tener seguimiento del balón sin pérdida de su posición. Para ello se realiza seguimiento utilizando un vídeo a distintas resoluciones, del cual se utiliza un análisis comparativo del error de cada una con respecto a la posición que se obtuvo a máxima resolución.

Se comprobará si existe seguimiento continuo durante el vídeo, además de obtener el error promedio y varianza con respecto a las posiciones obtenidas ante la resolución de 800 x 600 píxeles. Los factores de escalamiento serán 1, 2, 3, 10, 15, 20 y 30, donde el factor indica la reducción que se hace en ancho y largo de la imagen.

El resultado de esta prueba indicará hasta cuánto podemos reducir la cantidad de información que es procesada y en qué medida es afectada la posición que se adquiere de la pelota.

## *3.2. Predicción de las condiciones de juego*

Para lograr predecir la posición a futuro de la pelota, se utiliza el algoritmo Neuro-Difuso ANFIS, pero para poder implementarlo, primero se deben generar los datos con los que se entrenará el algoritmo.

Para ello se crea un ambiente simulado que permite tener control sobre las trayectorias del balón, utilizando movimientos rectilíneos uniformemente acelerados (MRUA) se van evaluando las posiciones a medida que transcurre el tiempo. Las expresiones que logran predecir

las posiciones son:

$$Pos_{i+1} = Pos_i + Vel_i * dt + \frac{1}{2} Acel_i * dt^2 \quad (3.3a)$$

$$Vel_{i+1} = Vel_i + Accel_i * dt \quad (3.3b)$$

$$Acel_{i+1} = \alpha * Acel_i \quad (3.3c)$$

Siendo el valor de  $\alpha$ , un factor que modifica la aceleración, como por ejemplo al rebotar sobre los bordes del campo de juego. Además se debe tomar en cuenta que la aceleración, siempre debe ser negativa ya que esta representa al roce que la cancha ejerce sobre el balón. Siendo la única instancia de aceleración positiva, el golpe de uno de los muñecos, pero al ser únicamente un instante al comienzo del movimiento, será la velocidad inicial que contenga esa información.

En cuanto a los límites del campo de juego, se utilizan las dimensiones máximas de 800 x 600, correspondientes a la resolución máxima entregada por la cámara Basler, lo que significa que la posición de la pelota también será medida en píxeles.

Con la simulación de la trayectoria de la pelota se pueden generar los datos requeridos para el set de entrenamiento. Como entrada a la red, se utiliza la posición de la pelota a medida que se mueve por una trayectoria, esto equivale a las coordenadas previas “x” e “y” que ha descrito el movimiento del balón, y como salida deseada, la posición cuando alcance el extremo donde se ubica el arquero.

Ya que el movimiento a controlar del portero es exclusivamente a lo ancho de la coordenada “y” en un rango 0-600 por el tamaño de resolución de la cámara, la posición en x se mantendrá fija con el valor en  $x = 0$ . Se debe señalar que el set de datos para el entrenamiento de la red son generados de forma aleatoria, posicionando la pelota dentro de los márgenes de la cancha de forma al azar (probabilidad uniforme sobre todo el campo de juego), y teniendo

en cuenta una velocidad inicial que puede llegar hasta los 5 [m/s] equivalente a 4166 [píxeles/s]. Además sólo se utilizan trayectorias en dirección hacia el arquero, por lo que no se especifica el comportamiento deseado de la salida cuando la pelota se aleja del arquero.

Existe además, un comportamiento que la simulación debe tener en cuenta y es, que la adquisición de posiciones, por parte del algoritmo que identifica la posición de la pelota por medio de imágenes, es propenso a perturbaciones y ruido, por lo tanto los datos entregados sobre la posición de la pelota contendrán variaciones que difieren respecto a la ubicación real de esta. Es por esto que el sistema de aprendizaje debe tener en cuenta datos con cierta intensidad de ruido en sus posiciones, acercándose a un ambiente no idealizado que se obtiene en la realidad.

Se toma en consideración 40 % del radio del balón, como error máximo para ser añadido a la ubicación del balón, siendo considerado como el peor caso. Cabe destacar que este error será añadido luego de simular las trayectorias, lo cual permitirá comparar el comportamiento de la red ANFIS con y sin ruido de medición.

### *3.2.1. Diseño de la Red ANFIS*

Para el diseño de la red ANFIS, se utiliza la herramienta computacional MATLAB, el cual permite generar la estructura difusa, además de realizar el entrenamiento de la red.

Para establecer la arquitectura de la red ANFIS, se utiliza el método “SubClustering” dado que los demás métodos investigados sólo pueden ser implementados bajo un limitado número de entradas. Además, la cantidad de reglas y funciones de membresías que son generadas por este método crecen de forma lineal y no exponencial, permitiendo que el sistema no se sobrecargue al iniciar el proceso de computación y entrenamiento. Cabe destacar que este método sólo permite su construcción con funciones de membresía del tipo Gaussiana.

Para determinar el sistema a utilizar, se realizarán una serie de pruebas con el fin de obtener las condiciones donde el algoritmo Neuro-Difuso entregue el menor error. Este será medido según el EMC o Error Cuadrático Medio según la expresión:

$$ECM = \frac{1}{n} \sum_{i=1}^n (\hat{y}_i - y_i)^2 \quad (3.4a)$$

Donde  $n$  corresponde al número de datos,  $\hat{y}_i$  la salida deseada e  $y_i$  el valor de salida para la trayectoria  $i$ .

Como entradas a la red, se propuso utilizar las últimas posiciones de la pelota, considerando las coordenadas x e y, provenientes del procesamiento de imágenes. Necesitando como mínimo, dos mediciones para poder determinar la posición actual y hacia donde se dirige.

Por otro lado, la salida de la red corresponderá a la ubicación donde el arquero deberá posicionarse para bloquear el paso de la pelota. En otras palabras, corresponderá a la posición donde el balón alcanza la portería.

Las pruebas a realizar contemplan la variación del número de entradas a la red, cantidad de reglas y trayectorias de entrenamiento, siendo estas última la cantidad de datos de entradas y salidas deseadas con que se entrenará la red, así como también el comportamiento ante perturbaciones en la entrada. Además, se modifica el rango de las funciones de pertenencia conjuntamente con el número de reglas. Y finalmente, se realizará un análisis comparativo del error de la red antes y después de entrenarse. Se utiliza un análisis exhaustivo, esto con el fin de establecer el comportamiento del sistema ante la variación de los parámetros que definen la estructura, ya que no se posee certeza de la metodología a seguir para que la red tenga un buen desempeño.

Se utilizará ruido normal en las mediciones de entrada, con una desviación estándar de

5 píxeles y valor promedio 0, siendo el diámetro de la pelota equivalente a 26 píxeles, se obtiene un error máximo del 40 % del radio de la pelota.

La primera prueba se determina el efecto de la variación del número de entradas a la red, donde se considerarán 2, 3, 5, 7, 9, 11 y 13 pares de coordenadas, correspondientes a las posiciones previas del balón, considerando para este caso 23 reglas. En esta situación se contempla la utilización de 2000 datos de entrenamiento y otros 500 para validar el comportamiento ante nuevas entradas.

La segunda prueba varía la cantidad de reglas, manteniendo el número de entradas a la red que dio el menor EMC con ruido de la prueba anterior. Para este caso se utilizan 2, 5, 10, 20, 30 y 40 reglas, con 2000 datos de entrenamiento y 500 para su validación.

La tercera prueba contempla la variación en la cantidad de datos de entrenamiento, donde se utilizan 100, 500, 1000, 1500, 2000, 2500, 3000, 3500, 4000, 4500 y 5000 trayectorias distintas considerando 500 datos de validación, la cantidad de entradas y reglas que dieron el menor error con ruido de la primera y segunda prueba.

En la cuarta prueba se modifica el rango que abarcan las funciones de membresía, variando el número de reglas acorde a este, se utilizan funciones de membresía con el 10 %, 25 % y 50 % del rango de la señal de entrada con 30, 20 y 10 reglas respectivamente. Donde se utilizan las condiciones que dieron el menor error de las pruebas anteriores.

Una quinta prueba comprueba el comportamiento de la red antes y después de ser entrenada, con los resultados que entregaron el menor error de las pruebas anteriores.

### 3.3. Modelado del motor DC y Diseño del controlador PID

Para modelar el motor DC es necesario tener los valores de los parámetros, como la resistencia e inductancia de armadura o el voltaje y corriente de entrada para su funcionamiento. En este caso se utilizan los parámetros de un motor comercial tipo DC con escobillas de Maxon Motors modelo DCX 35 L  $\phi$  35 mm, por las características similares al que se utilizará en el proyecto universitario de par y velocidad nominal que alcanza, correspondiente a un motor Servo-Brushless DC modelo CPM-SCSK-2310P-RLNB [22]. Las características del motor Maxon se muestran en la tabla 3.2.

Tab. 3.2: Parámetros y valores del motor DC

Parámetro	Valor
Voltaje Nominal	24 [V]
Velocidad Nominal	7160 [RPM]
Torque Nominal	121 [mNm]
Corriente Nominal	4.26 [A]
Corriente de Arranque	69.3 [A]
Resistencia	0.346 [ $\Omega$ ]
Inductancia	0.121 [mH]
Constante de Torque	29.3 [mNm/A]
Constante de fuerza electro-motriz	326 [RPM/V]
Relación velocidad/torque	2.86 [RPM/mNm]
Momento de inercia del rotor	$9.66 \times 10^{-6}$ [Kgm <sup>2</sup> ]
Velocidad Límite	12300 [RPM]
Diámetro del Motor	35 [mm]

Con el fin de simular el comportamiento, se utiliza el software computacional Matlab & Simulink, que permitirá realizar tanto el sistema de control como el modelado del motor DC.

En cuanto al comportamiento del par de carga, dado que no se cuenta con la curva característica de ella, se realiza un supuesto de su dinámica, se considera como un sistema rígido, donde no existe elasticidad por parte de los mecanismos que componen al sistema, por lo que la carga puede ser añadida a la inercia del motor y considerar que el sistema no tiene una perturbación [23]. Además, se considerará una masa de 0.5 [Kg] para la barra del arquero a desplazar junto con un radio de giro del eje del motor de 2 [cm], el cual contempla un sistema de correa transportadora para su movimiento.

Con estos datos es posible obtener la inercia de la barra y dada su magnitud con respecto a la inercia del motor, este ultimo es despreciado en el sistema resultante.

$$J_{barra} = mR^2 \quad \text{Kgm}^2 \quad (3.5a)$$

$$J_{barra} = 0,5 * (2 * 10^{-2})^2 \quad (3.5b)$$

$$J_{barra} = 2 * 10^{-4} \quad \text{Kgm}^2 \quad (3.5c)$$

Con los parámetros y valores se obtiene el modelo en Simulink (figura 3.1) que representa el comportamiento dinámico del motor DC con carga:

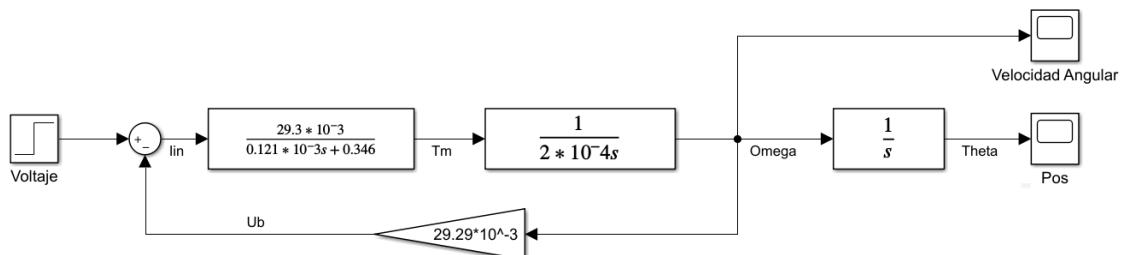


Fig. 3.1: Modelo con carga

Este modelo puede ser expresado simplemente como un bloque, el que posee como entrada un voltaje y una velocidad angular de salida, que al ser integrada se obtiene la posición angular, como es expresado en la figura 3.2.



Fig. 3.2: Modelo en bloques del motor DC

Con el modelo del motor, y los datos característicos del mismo, se obtiene la función de transferencia del motor DC. La función de transferencia del motor puede ser expresada como:

$$G(s) = \frac{1,2107 * 10^6}{s(s + 2847)(s + 12,46)} \quad (3.6a)$$

$$G(s) = \frac{Kg}{s(s + P1)(s + P2)} \quad (3.6b)$$

$$(3.6c)$$

Donde  $Kg$  corresponde a la ganancia de la planta  $G(s)$ ,  $P1$  corresponde al primer polo (-2847) y  $P2$  al segundo polo (-12.46) del modelo del motor  $G(s)$ .

Con este modelo, se realiza una realimentación con el controlador PID como se muestra en la figura 3.3.

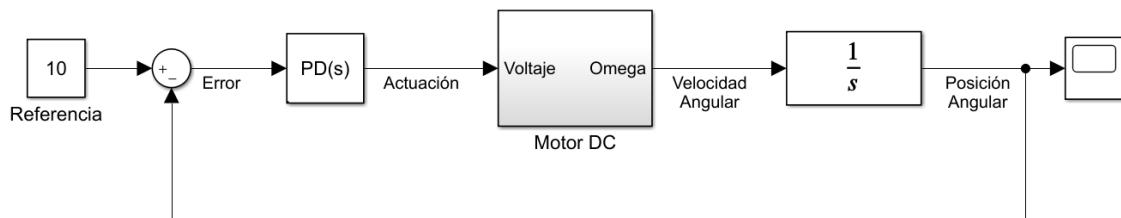


Fig. 3.3: Modelo con lazo de realimentación

Por otra parte, se busca posicionar correctamente al arquero y para ello se debe seleccionar el tipo de controlador a utilizar, siendo los controladores P, PI, PD y PID las alternativas que

se poseen, se comprueba el comportamiento ante el lazo cerrado según la expresión 3.7, que indica la función de transferencia  $H(s)$  del sistema ante una planta  $G(s)$ , un controlador  $C(s)$  y una realimentación unitaria como se mostró en la figura 2.42.

$$H(s) = \frac{C(s)G(s)}{1 + C(s)G(s)} \quad (3.7a)$$

Con las función de transferencia de los controladores:

$$C_P(s) = K_p \quad (3.8a)$$

$$C_{PI}(s) = K_p + \frac{K_i}{s} \quad (3.8b)$$

$$C_{PD}(s) = sK_d + K_p \quad (3.8c)$$

$$C_{PID}(s) = sK_d + K_p + \frac{K_i}{s} \quad (3.8d)$$

El controlador P, entrega una función de transferencia de lazo cerrado de grado relativo 3, que no permite mayor manipulación de los polos del sistema, ya que estos siguen siendo dependientes de los polos de la planta como se muestra en la ecuación 3.9.

$$H_P(s) = \frac{K_p K_g}{s(s + P_1)(s + P_2) + K_p K_g} \quad (3.9a)$$

$$H_P(s) = \frac{K_p K_g}{s^3 + s^2(P_1 + P_2) + sP_1P_2 + K_p K_g} \quad (3.9b)$$

El controlador PI, entrega una función de lazo cerrado de grado relativo 3, donde dependiendo de la ganancia  $K_p$  utilizada, el sistema puede tener inestabilidad, al tener polos en el semi-plano derecho. Además, para simplificar la expresión se utiliza  $\frac{K_g}{K_p} = A$  y  $\frac{K_i}{K_p} = P_2$  como se aprecia en la ecuación 3.10:

$$H_{PI}(s) = \frac{Kg(sKp + Ki)}{s^2(s + P1)(s + P2) + Kg(sKp + Ki)} \quad (3.10a)$$

$$H_{PI}(s) = \frac{A}{s^3 + s^2P1 + A} \quad (3.10b)$$

El controlador PD, entrega una función de lazo cerrado de grado relativo 2, con una ecuación de segundo orden de la forma estándar. Además, para simplificar la expresión, se utiliza  $\frac{Kg}{Kd} = A$  y  $\frac{Kp}{Kd} = P2$  como se aprecia en la ecuación 3.11:

$$H_{PD}(s) = \frac{Kg(sKd + Kp)}{s(s + P1)(s + P2) + Kg(sKd + Kp)} \quad (3.11a)$$

$$H_{PD}(s) = \frac{A}{s^2 + sP1 + A} \quad (3.11b)$$

Finalmente el controlador PID, entrega una función de transferencia de grado relativo 2, pero con polos complejos y sin parte real, lo cual indica inestabilidad. Además para simplificar la expresión, se utiliza  $\frac{Kg}{Kd} = A$ ,  $\frac{Kp}{Kd} = (P1 + P2)$  y  $\frac{Ki}{Kd} = P1P2$  como se aprecia en la ecuación 3.12

$$H_{PID}(s) = \frac{Kg(s^2Kd + sKp + Ki)}{s^2(s + P1)(s + P2) + Kg(s^2Kd + sKp + Ki)} \quad (3.12a)$$

$$H_{PID}(s) = \frac{A}{s^2 + A} \quad (3.12b)$$

Como se puede observar, al simplificar las expresiones se obtienen distintos comportamientos. En el caso del controlador PID, este posee un polos complejos conjugados con parte real cero, lo que provoca oscilaciones sinusoidales permanentes en el sistema.

En los casos de controlador PI y P, la función de transferencia posee grado relativo 3, y sólo un parámetro para lograr manipular los polos del sistema. Para identificar la estabilidad

del lazo se utiliza el teorema de Routh, el cual indica la cantidad de polos en el semi plano de recho según los cambios de signos en su algortimo de ejecución. Para el caso del controlador PI, el lazo es inestable, ya que contará con polos positivos independiente del valor de  $A$ . Por otro lado el controlador P, muestra que al aumentar el valor del factor  $KpKg$ , existirán polos que se irán desplazando cada vez más a la derecha terminando finalmente en polos dentro del semi plano derecho, volviendo inestable el lazo, lo que indica que la ganancia con la que se puede actuar estará limitada.

Finalmente, el controlador PD logra manipular los polos de lazo cerrado aumentando su ganancia  $A$  de su sistema, lo cual provoca que la función de transferencia de segundo orden genere polos conjugados en algún momento, aunque no provoca la inestabilidad del lazo, sí agrega oscilaciones al comportamiento.

Con estos resultados, se emplea un lazo de control PD con un comportamiento críticamente amortiguado, esto es, que los polos de lazo cerrado sean iguales y sin parte imaginaria para evitar introducir oscilaciones dentro del sistema, y además dado que se está realizando un control de posición, el portero no puede sobrepasar los límites de la cancha ya que se estrellaría con los bordes del campo de juego, considerando esta situación y que el comportamiento críticamente amortiguado entrega una respuesta más rápida en comparación al sobre amortiguado, se utiliza esta primera, donde se utilizará la herramienta computacional SISOTOOL de Matlab para escoger los valores del controlador mediante lugar geométrico de raíces. Cabe destacar que la simplificación siempre fue realizada con el polo más lento (o de menor magnitud) de la planta para eliminar la dinámica más lenta del sistema y que reaccionara de forma más rápida ante la señal de referencia.

### 3.4. Simulación

En última instancia, se realizará la simulación integrando los resultados de cada sección, donde se evalúa el comportamiento de la predicción por parte del algoritmo ANFIS y el seguimiento a referencia por parte del motor. Si bien no será implementado el uso de la cámara para la adquisición de los datos de las posiciones, se utilizará el sistema de generación de trayectorias para generar las posiciones de la pelota según una posición, velocidad y aceleración inicial, con el cual se medirá el comportamiento del sistema ante el movimiento de la pelota, corroborando que ANFIS entregue una referencia acorde al desplazamiento de esta y que el motor siga a este valor.

Como se muestra en la figura 3.4, el primer bloque generará nuevas posiciones del balón, ANFIS entregará una referencia y el motor DC se moverá acorde a ella. Por otro lado el bloque del campo de juego, permitirá observar las distintas señales y cambios que ocurren dentro.

Para este propósito, se utilizará Matlab & Simulink, utilizando los bloques de programación “sFunction” para programar el código necesario de cada bloque en concreto.

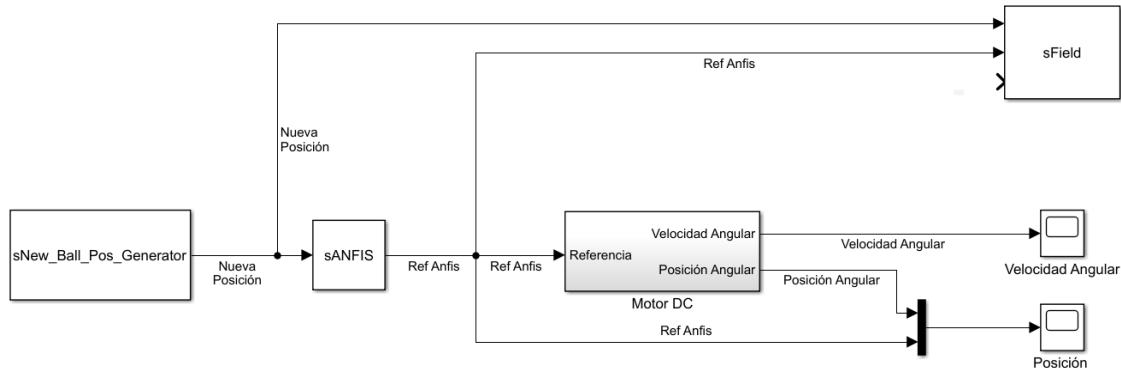


Fig. 3.4: Respuesta a escalón

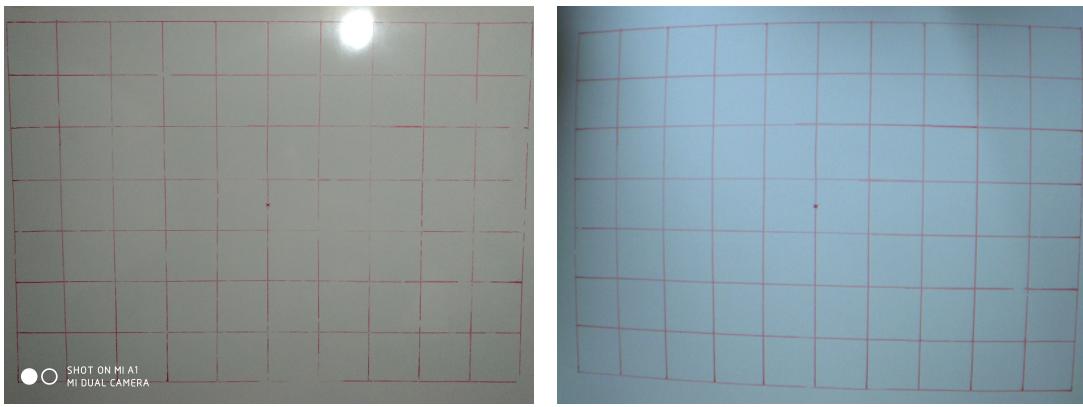
## 4. ANÁLISIS DE RESULTADOS

### 4.1. *Adquisición de imágenes*

En primera instancia se utilizó el software “Pylon Viewer” para la adquisición de imágenes, en donde se corroboró tanto el funcionamiento de la cámara, como también los efectos producidos por la manipulación de los parámetros que ésta posee. En este procedimiento, se realizan diversas pruebas que permiten establecer las condiciones para un correcto funcionamiento del sistema, ya sea por medio de la reducción de ruido, el aumento de la iluminación en escena, o bien la distancia a la cual deba estar la cámara.

#### 4.1.1. *Efecto del lente*

En cuanto al objetivo de la cámara, se posee un lente de 4mm (distancia focal) el cual permite posicionar el dispositivo a 1.25 [m] de la mesa de juego según la página web de Basler. En comparación, el lente de 6mm coloca a la cámara a una distancia de 1.88 [m] sobre la mesa, lo cual dificulta la implementación en un ambiente reducido, decidiendo que el lente de 4mm se ajusta mejor para este tipo de aplicación. Un aspecto contraproducente es la deformación de los objetos producto de una baja distancia focal (figura 4.1), la cual tiene como resultado que las esquinas de la imagen se curven al tratar de contener un mayor campo visual.

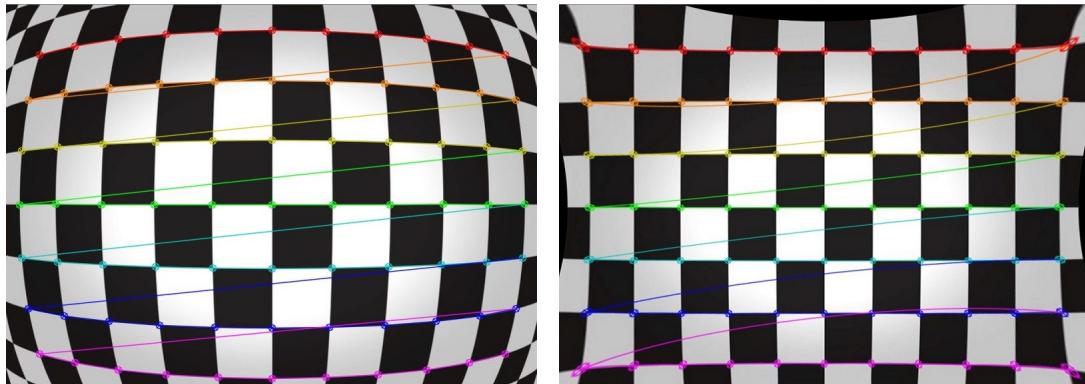


(a) Imagen normal

(b) Imagen distorsionada por lente de 4mm

*Fig. 4.1: Comparación en la distorsión de imagen producto del lente. Las medidas de cada cuadro corresponden a  $10 \times 10 \text{ [cm}^2\text{]}$ , siendo en total  $1.0 \times 0.7 \text{ [m}^2\text{]}$*

Se aprecia de las figuras 4.1 dos consideraciones que se deben tener en cuenta, la primera es la deformación que sufren las dimensiones del campo de juego al tener cierta inclinación en la cámara de vídeo. El segundo aspecto, es la distorsión de la imagen producto del lente agregando el efecto barril el cual se puede apreciar de mejor forma al observar las esquinas de la grilla. Si bien este efecto puede ser eliminado mediante procesamiento proyectivo como se puede apreciar en las figuras 4.2, este proceso añade un retardo al procesamiento al sistema, correspondiente a 38 [ms] para este caso. Dado el tiempo necesario para eliminar esta distorsión, se decide dejar de lado esta etapa ya que de ser necesario, sólo la posición de la pelota deberá ser modificada y no la imagen completa.



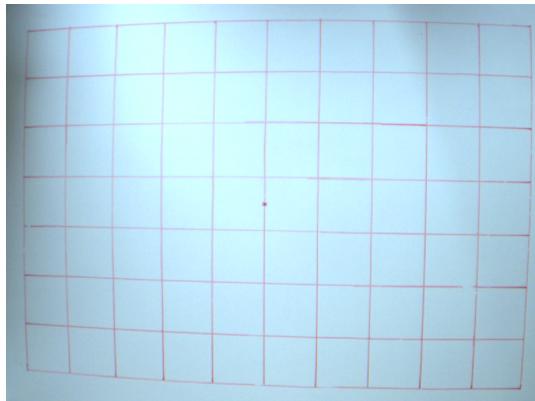
(a) Imagen normal

(b) Imagen distorsionada por lente de 4mm

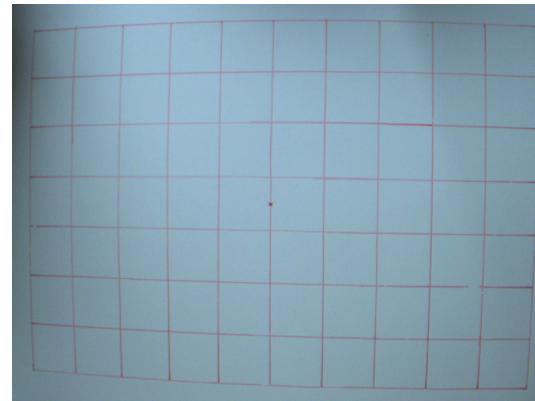
Fig. 4.2: Corrección del efecto barril

#### 4.1.2. Velocidad de adquisición vs Intensidad

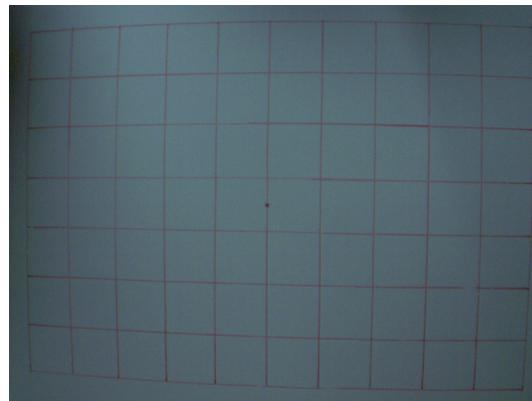
Observamos en las figuras 4.3a, 4.3b, 4.3c, que al aumentar la velocidad de obturación (equivalente a disminuir el tiempo de exposición) tiene como consecuencia una bajada en la intensidad (o valor) de los píxeles de la imagen, la cual decae al tener una menor cantidad de luz recibida en el sensor. Estas imágenes fueron adquiridas con un lente de distancia focal de 4mm y una resolución de 800x600 píxeles.



(a) Imagen a 200 fps



(b) Imagen a 300 fps



(c) Imagen a 400 fps

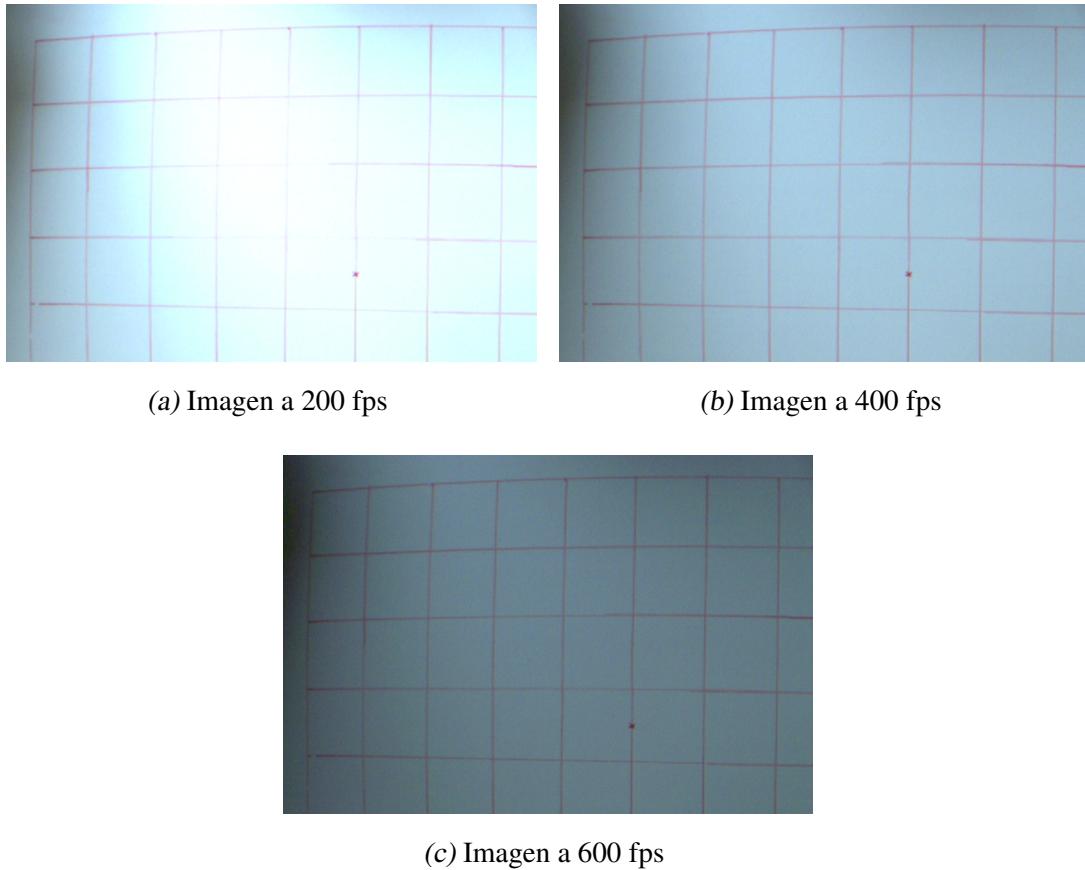
Fig. 4.3: Comportamiento de la intensidad de los píxeles al variar la velocidad de adquisición con resolución de 800 x 600 píxeles

#### **4.1.3. Limitaciones por ancho de banda**

Un aspecto importante a señalar es que existe una limitación por parte del dispositivo en cuanto a la tasa de datos que puede enviar por el conector USB 3.0 (5 Giga bits por segundo). Si consideramos la resolución de 800 x 600 píxeles, con 3 canales (RGB) los cuales cada uno de ellos posee 8 bits. tendremos una carga de datos de 11.52 Mega bits por imagen. Luego, podemos calcular el valor máximo teórico de cuadros por segundo que permite el conector, dividiendo la capacidad por la carga, dando un total de 434 cuadros por segundo. Este valor no es del todo cierto en la práctica, ya que debe existir una comunicación entre el dispositivo

y el computador, por lo que es necesario transmitir una mayor cantidad de información que sólo las imágenes. Es por esto que la cantidad de cuadros por segundo será menor al valor teórico. En las pruebas realizadas, se alcanzaron valores máximos cercanos a los 411 fps a una resolución de 800x600 del dispositivo.

Como consecuencia, si se disminuye la resolución de las imágenes adquiridas se logrará aumentar la velocidad de adquisición. Mostrando los resultados en las figuras 4.4a, 4.4b, 4.4c, donde se llegaron a tasas de hasta 600 fps al reducir la cantidad de píxeles a la mitad. Además, se puede observar que las imágenes poseen mayor intensidad (siendo más luminosas) que al utilizar una resolución completa, al comparar los casos de 200 y 400 fps. Esto es producto de que el tiempo de exposición cambió, dado que se necesita de un menor periodo para transmitir la imagen, se puede utilizar la diferencia de tiempo en captar una mayor cantidad de luz.



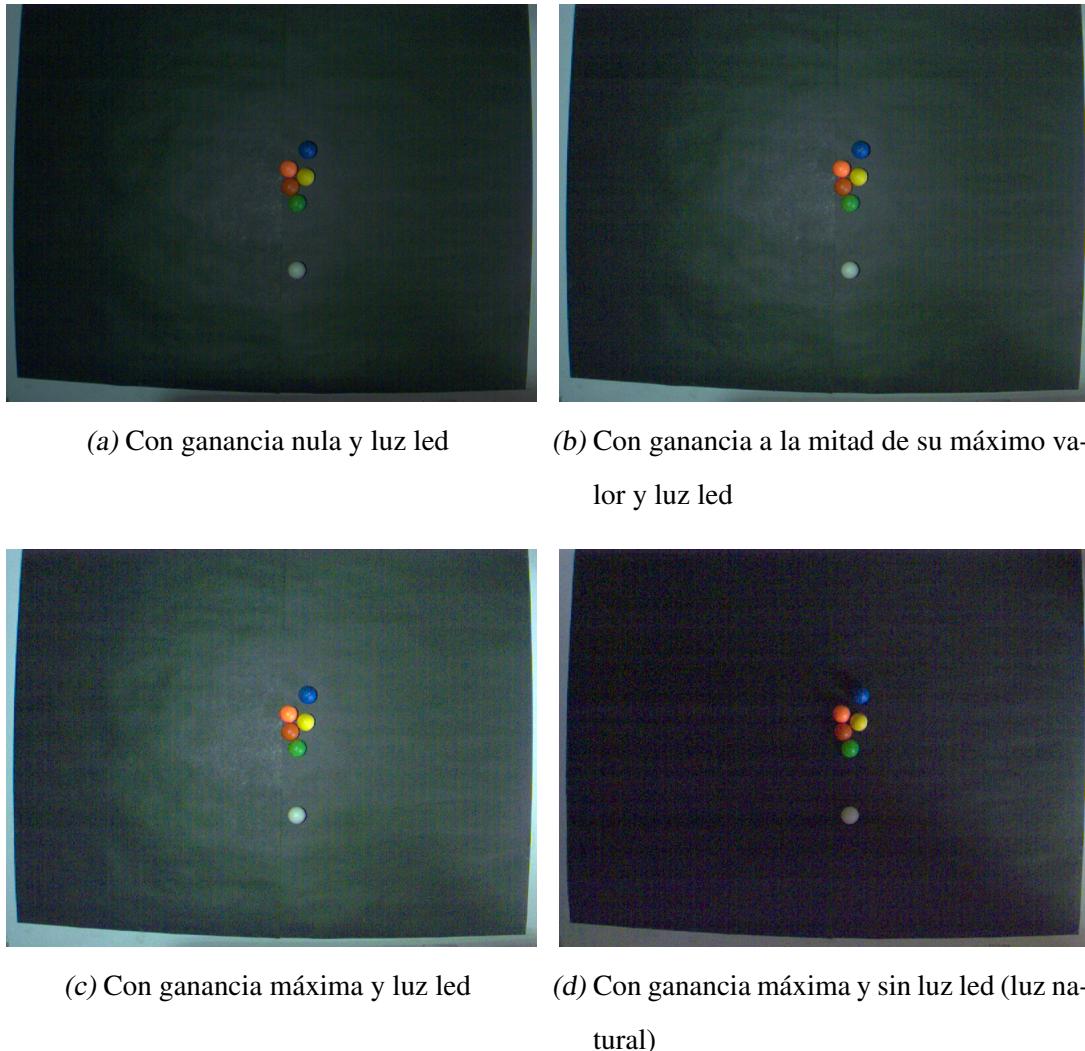
*Fig. 4.4:* Comportamiento de la intensidad de los píxeles al variar la velocidad de adquisición con resolución de 600 x 400 píxeles

#### 4.1.4. Ganancia ISO e Iluminación

Del resultado anterior, observamos cómo la intensidad en los píxeles se ve afectada ante el incremento o decremento de la velocidad de adquisición. Para mejorar la imagen usando altas tasas de velocidad, existen dos caminos posibles que pueden ser usados conjuntamente, el primero es la utilización de iluminación artificial, mientras que el segundo corresponde al aumento en la ganancia de la imagen.

Como se ha mencionado previamente, un aumento en la ganancia o ISO del sensor no sólo se incrementará la intensidad de los píxeles, sino que también aumentará la magnitud del ruido existente en la imagen. Esto lo observamos en las siguientes fotografías (figuras

4.5a, 4.5b, 4.5c, 4.5d), las cuales fueron tomadas a la máxima velocidad posible (400 fps) con una resolución de 800x600 píxeles, y en algunos casos iluminación led mediante el uso de un foco led de 10 [W] de potencia.



*Fig. 4.5:* Comportamiento del ruido al variar la ganancia del sensor a una tasa de 400 fps con resolución de 800 x 600 píxeles

Se observa de las figuras 4.5, que la luz led mejora la adquisición de colores observado en las esferas, ya que produce tonalidades uniformes reduciendo las sombras. Por otra parte, observamos que la luz led inyecta una tonalidad verde al fondo negro, lo cual resultará da-

ñino para la detección de colores en la etapa siguiente. Y ademas, se tiene que la ganancia a medida que aumenta, incrementa las intensidades de los colores, pero esto a su vez provoca que se incremente la tonalidad verde agregada por la luz led.

Se continua con la medición del ruido en los colores de la imagen, para ello se utiliza exclusivamente el canal que contiene la información del matiz (HUE) y se mide el promedio y la varianza según las ecuaciones 3.1 para lograr establecer una relación de que método disminuye el ruido de adquisición. Se utiliza un cuadrado de dimensiones 100x100 para las cuatro imágenes anteriores (figuras 4.5) situado en la misma posición para todas ellas (figura 4.6). Se observa de las figuras 4.7 y la tabla 4.1, que el ruido disminuye cuando se tiene iluminación led al comparar las varianzas, pero no se puede establecer una relación del ruido con respecto a la ganancia.

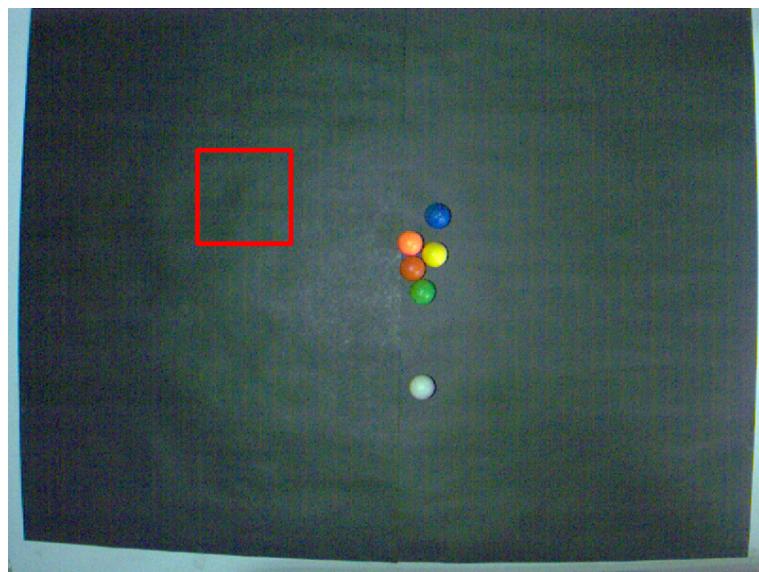
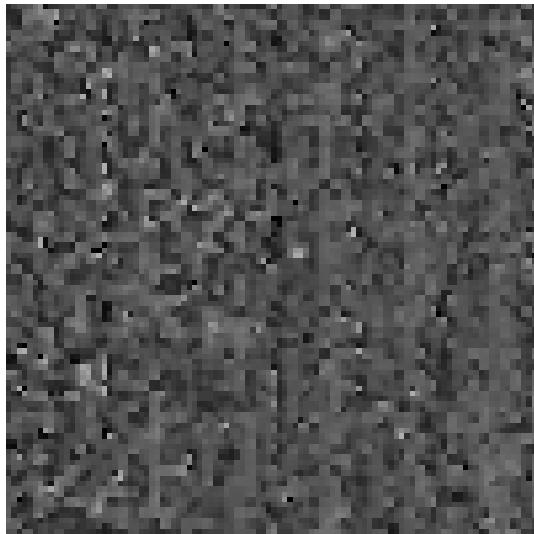
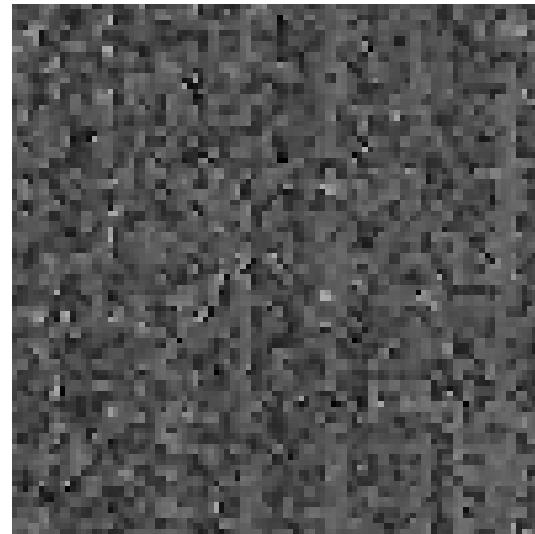


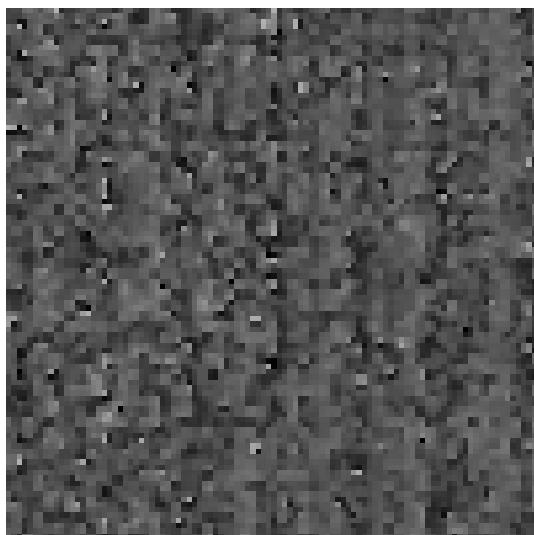
Fig. 4.6: Cuadrado utilizado para medir ruido en las imágenes



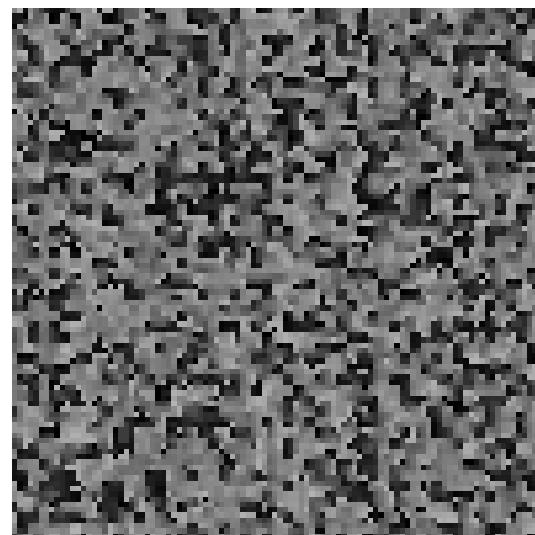
(a) Ruido con ganancia nula y luz led



(b) Ruido con ganancia a la mitad de su máximo valor y luz led



(c) Ruido con ganancia máxima y luz led



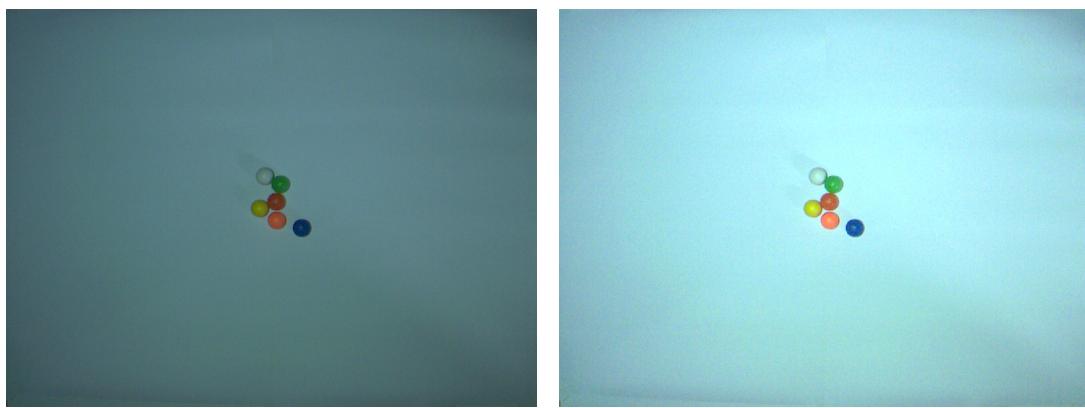
(d) Ruido con ganancia máxima y sin luz led  
(luz natural)

*Fig. 4.7: Ruido presente en el canal de Matiz de acuerdo a cada imagen*

*Tab. 4.1:* Resultados de la medición de ruido en fondo negro

Imagen	Media	Varianza
Ruido con ganancia nula y luz led	78.14	75.32
Ruido con ganancia a la mitad y luz led	79.03	77.8
Ruido con ganancia máxima y luz led	80.25	73.8
Ruido con ganancia máxima y sin luz led	97.57	90.09

Para eliminar la tonalidad verde, se utilizó un fondo blanco, el cual a pesar de generar una tonalidad azulada al ser expuesto por la luz led (figura 4.8) mantiene una tonalidad uniformidad. Además se elimina el efecto de las sombras externas que puedan afectar al campo de juego al introducir la iluminación artificial, lo cual no era apreciado con un fondo negro. En la tabla 4.2 se muestran los valores promedio y de varianza que al compararlos con el fondo negro, se observa una diferencia considerable para los casos que cuentan con iluminación, dejando tonalidades más uniformes y disminuyendo el ruido en la imagen. Aunque las mediciones muestran que usar una ganancia nula o máxima no afectan en gran forma al ruido generado, si se observa que los colores poseen una mayor intensidad (o valor) al utilizar una ganancia, lo cual, no modifica el desempeño del sistema de manera significativa.



(a) Con ganancia nula y luz led

(b) Con ganancia y luz led



(c) Con ganancia máxima y sin luz led

*Fig. 4.8:* Comportamiento del ruido al variar la ganancia del sensor a una tasa de 400 fps con resolución de 800 x 600 píxeles, con fondo blanco

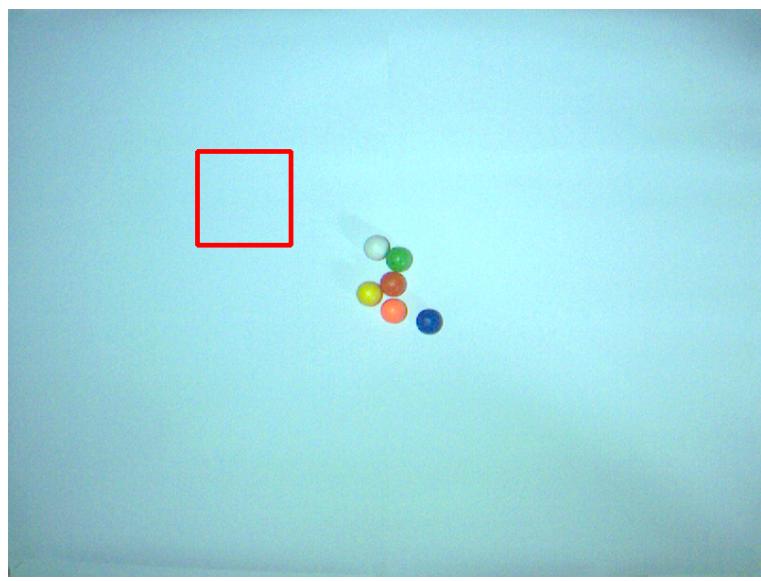
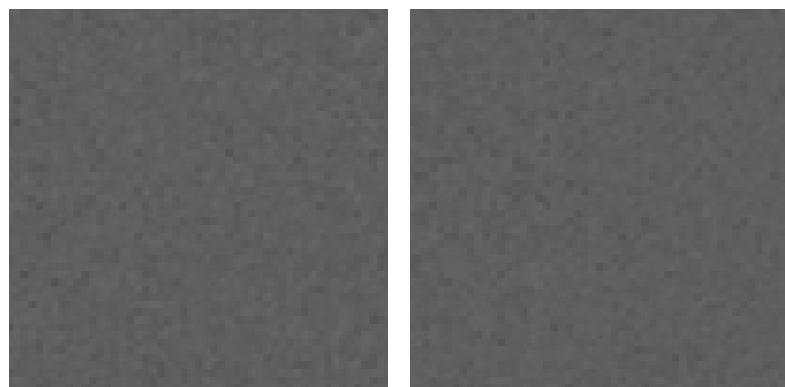
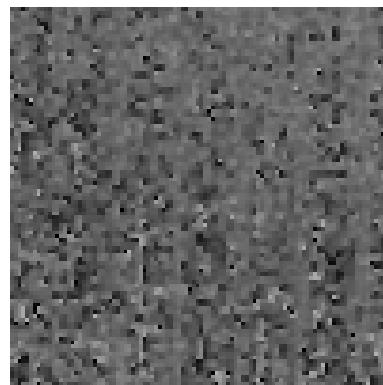


Fig. 4.9: Cuadrado utilizado para medir ruido en las imágenes



(a) Ruido con ganancia nula y luz led      (b) Ruido con ganancia máxima y luz led



(c) Ruido con ganancia máxima y sin luz led

Fig. 4.10: Ruido presente en el canal de Matiz de acuerdo a cada imagen con fondo blanco

Tab. 4.2: Resultados de la medición de ruido en fondo blanco

Imagen	Media	Varianza
Ruido con ganancia nula y luz led	94.73	5.50
Ruido con ganancia máxima y luz led	94.51	5.52
Ruido con ganancia máxima y sin luz led	94.71	69.47

#### 4.1.5. Iluminación y sombras

Se evalúa el comportamiento de la iluminación según la posición de las luces, donde se observa que se produce una saturación al posicionar los focos cerca del campo de juego (un foco led de 10 [W] y una ampolleta a gas de 15[W]), resultando en zonas completamente blancas dificultando medir los objetos que se encuentren sobre ellas (figura 4.11). Otro aspecto a mencionar, es la generación de sombras cuando la fuente lumínica no se encuentra posicionada sobre el objeto, por lo que para los elementos que están con cierto ángulo en referencia a la luz, provocarán sombras que afectarán al proceso de detección.



(a) Luces en las esquinas y sobre la superficie (b) Luces en las esquinas y levantadas, con de juego cierta inclinación



(c) Luces sobre el campo de juego

Fig. 4.11: Efecto del posicionamiento de las luces

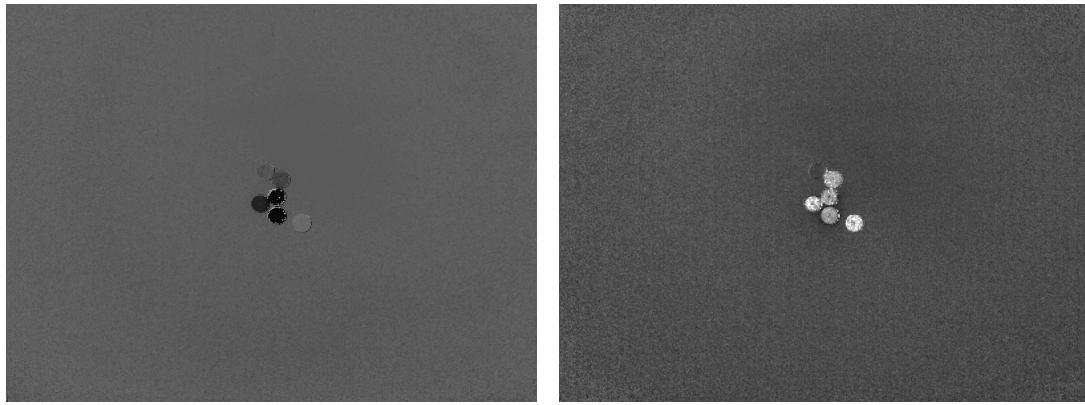
## 4.2. Identificación de la posición de la pelota

Utilizando una imagen de referencia adquirida por la cámara Basler (figura 4.12), considerando las condiciones de menor ruido descritas en la sección anterior, se procede a realizar el proceso de identificación por color para adquirir la posición de la pelota dentro de la imagen.



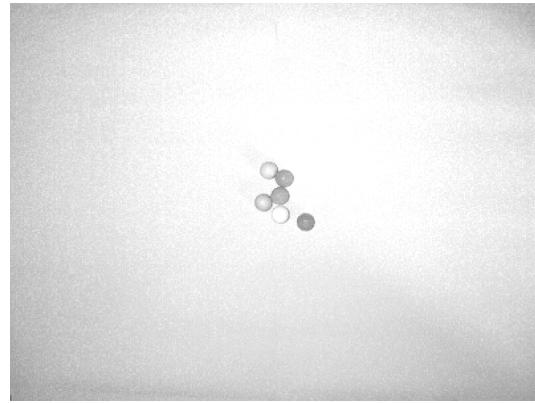
*Fig. 4.12:* Imagen de referencia

Luego de adquirir la imagen, se pasa desde el espacio de colores RGB proveniente desde la cámara Basler a uno HSV. Este proceso nos entregará un canal (Hue) que contiene la información de los colores de cada píxel (figura 4.13a) compactando la información de tres canales a sólo uno. Se muestra en las figuras 4.13 cada componente de forma independiente con valores desde 0 a 255.



(a) Hue: Matiz

(b) Saturation: Saturación



(c) Value: Valor

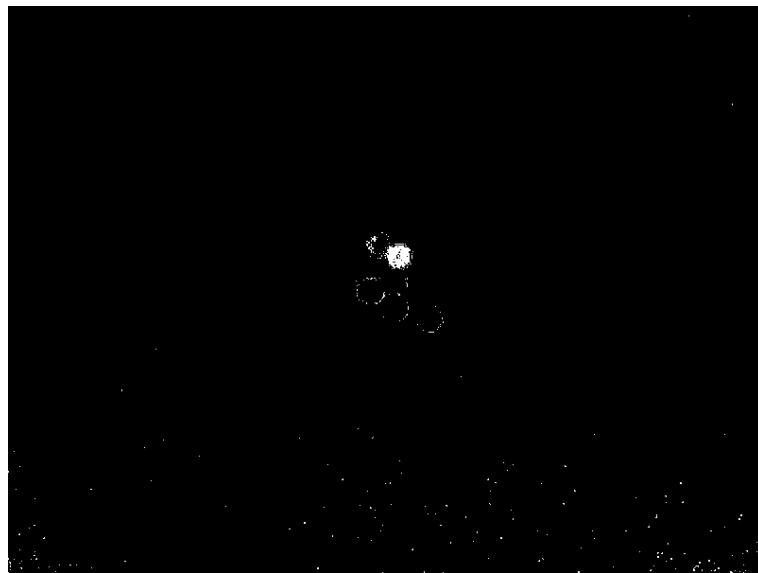
*Fig. 4.13: Canales HSV de la imagen*

Posteriormente, se procede a eliminar (llevar a un valor 0) a todo elemento que se encuentre fuera de los rangos deseados. Si bien el valor de matiz va de valores desde 0 a  $360^\circ$  (por la rotación sobre la circunferencia de colores (figura 2.22b)), los píxeles sólo poseen 256 valores posibles, siendo necesario escalar la información al rango permitido. Cada color poseerá un determinado valor, pero dado que estos no cambian de forma abrupta a medida que se recorre la circunferencia, se establece un rango donde encontrar los colores de interés, señalados en la figura 4.3.

*Tab. 4.3: Rango de valores para identificación de objetos*

Color	Rango de valores
Rojo	0-20
Amarillo	30-52
Verde	80-110
Azul	145-160

Para los próximos resultados se ha propuesto la identificación del color verde, aunque el procedimiento es independiente del matiz a identificar. Como resultado de esta selección, los píxeles tomaran un valor 1 cuando se encuentren dentro de los rangos del color verde y 0 en caso contrario. Sólo con el propósito de poder visualizar el resultado se han utilizado los valores 255 y 0 (figura 4.14). Se observa que en la imagen aparece una zona blanca donde se encuentra la pelota verde y además surgen puntos blancos donde no se encuentra la pelota a identificar, esto es producto del ruido provocado por los tres factores antes mencionados (ganancia de la cámara o sensibilidad ISO, iluminación y velocidad con que se adquiere la imagen), además del propio circuito electrónico del dispositivo.



*Fig. 4.14: Imagen sin filtrar*

Para reducir el ruido existente en la imagen, se procede a filtrarla con el kernel 3x3 pro-

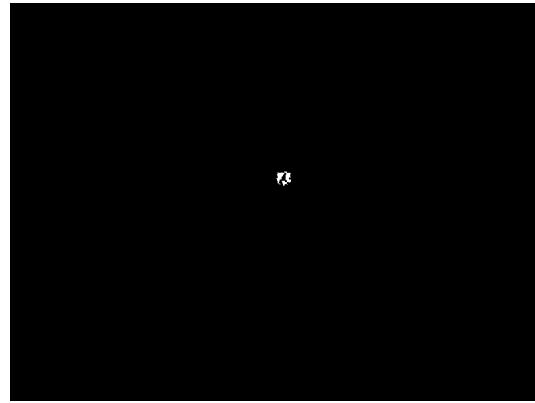
puesto en la etapa de metodología.

Se destaca que el filtro puede ser permisivo, o agresivo dependiendo del umbral que sea establecido. El valor máximo que puede llegar a tener es 9, donde todos los píxeles dentro del kernel 3x3 deben tener valor 1 para que el resultado de la operación sea 1. Por el contrario, el mínimo valores 0, donde independiente de la suma de los componentes dentro del kernel, el resultado siempre será mayor o igual al umbral entregando un valor 1 en todos los casos, es decir una imagen en blanco. En la figura 4.15 se muestra el resultado de aplicar un kernel con tres umbrales distintos, 3, 5 y 8, con el fin de compararlos. Se establece por inspección que un umbral de valor 5, resulta no ser tan agresivo eliminando información del objeto, pero si reduce en gran forma el ruido de las imágenes (figura 4.15b).



(a) Umbral = 3

(b) Umbral = 5



(c) Umbral = 8

*Fig. 4.15:* Comparación de filtro con umbrales 3, y 8.

Luego de obtener una imagen filtrada, se procede a encontrar las coordenadas de posición del objeto. Para ello se utiliza el método del centro de masa, de esta forma sólo los píxeles de valor 1 aportarán información al resultado. De esta forma, se obtiene la posición donde se acumula la mayor cantidad de puntos del color buscado. Si bien, hay píxeles que corresponden a ruido de la imagen, no logran incidir en el resultado en gran forma, siempre y cuando el objeto a identificar posea un color distintivo al resto. Esto entrega como resultado la posición de la pelota, el cual se puede apreciar en la imagen 4.16.

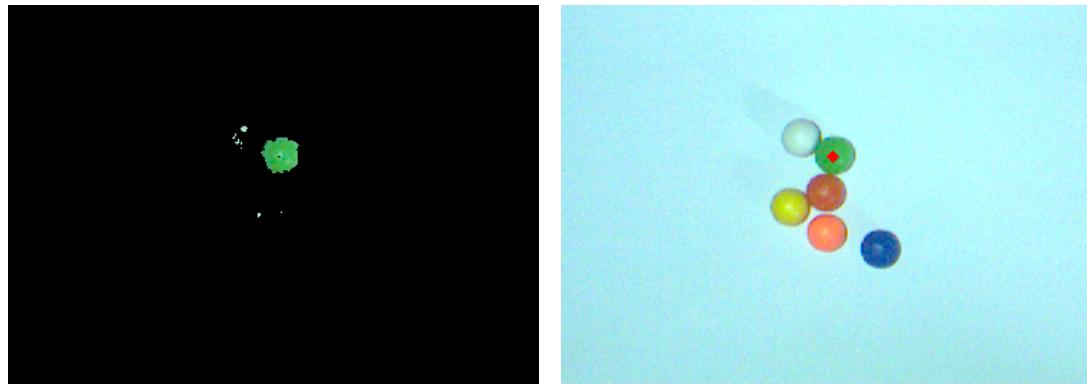


Fig. 4.16: Posición del objeto (punto rojo en la imagen sobre la pelota verde)

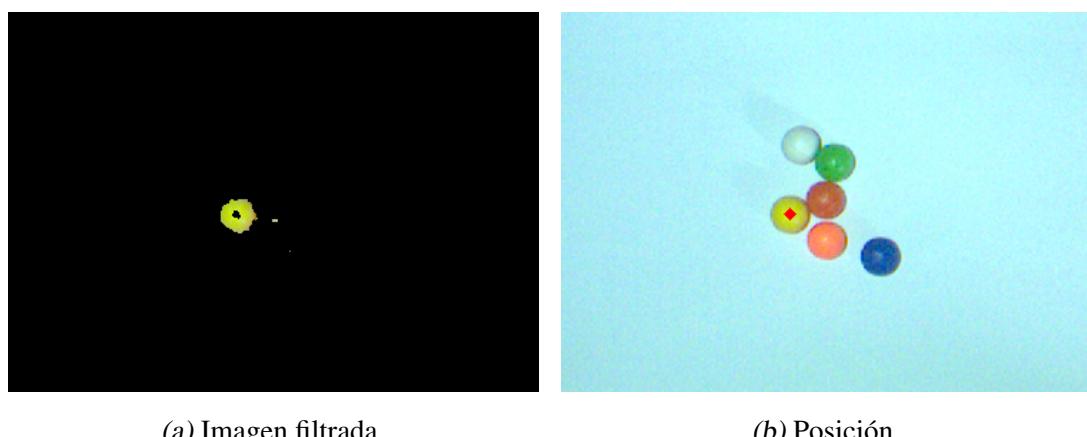
De igual forma podemos repetir el proceso para el color que sea necesario (figuras 4.17 a la 4.20). Como ha señalado, el color de los objetos debe ser distintivo para lograr separar un objeto del otro. En el caso de color rojo, tanto la pelota de naranja como la roja son detectadas por el algoritmo (figura 4.20) producto de que sus matices son similares, resultando en la detección de ambos objetos, y posicionando el centro de masa en medio de ambos.

Además, se observa en las figuras 4.17a, 4.18a, 4.19a, 4.19a, que el centro en los objetos aparece en negro. Esto es producto del reflejo que se presenta en ellos cambiando los colores en ese lugar y provocando que el algoritmo no logre identificar esas zonas. Este brillo no llega a generar problemas ya que el foco está justo encima y genera los reflejos en el centro de los balones. Pero en caso de posicionar 2 o 3 focos lumínicos en distintas posiciones, esto puede generar perturbaciones a la hora de obtener el centro de masa, ya que aumentando la

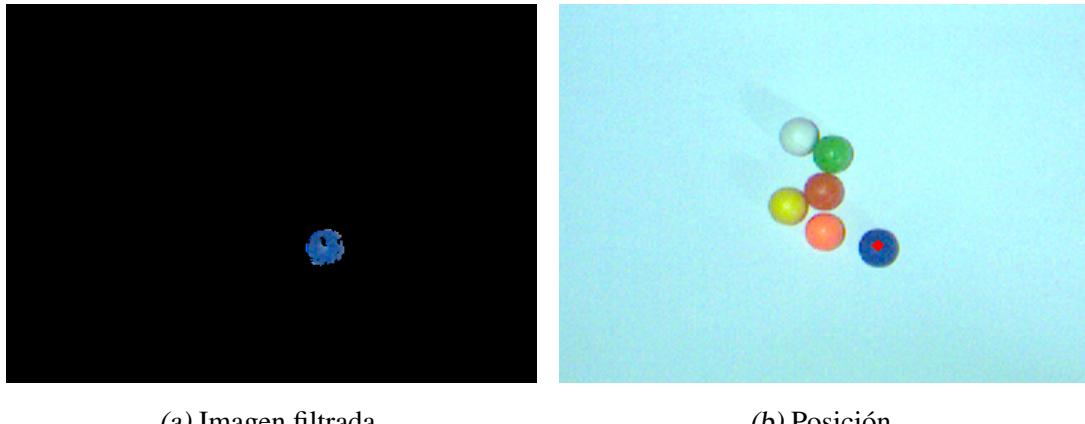
cantidad de iluminación sobre el campo de juego también aumentará el brillo reflejado. En casos extremos se podría utilizar filtros polarizados para disminuir el reflejo, pero estos también disminuirían la cantidad de luz que llega al sensor generando el problema que se tenía en un comienzo (falta de iluminación sobre la escena).



*Fig. 4.17: Detección color verde*



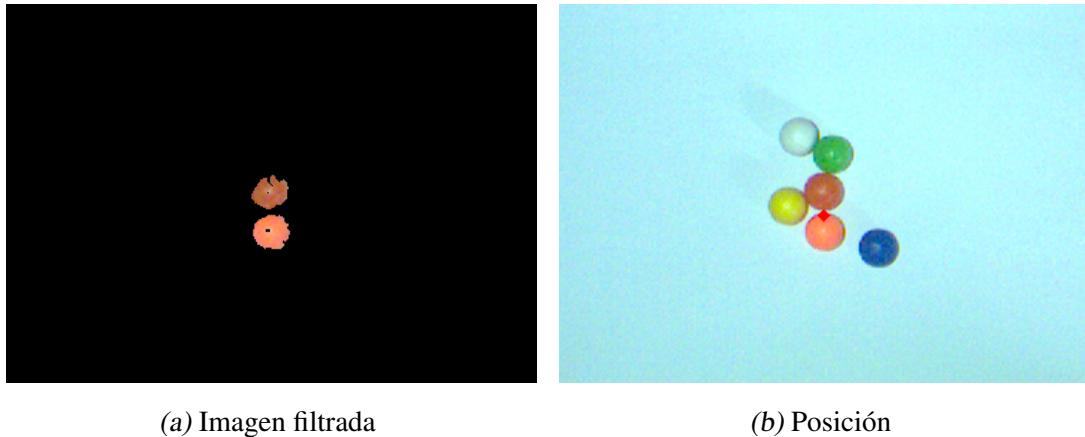
*Fig. 4.18: Detección color amarillo*



(a) Imagen filtrada

(b) Posición

Fig. 4.19: Detección color azul



(a) Imagen filtrada

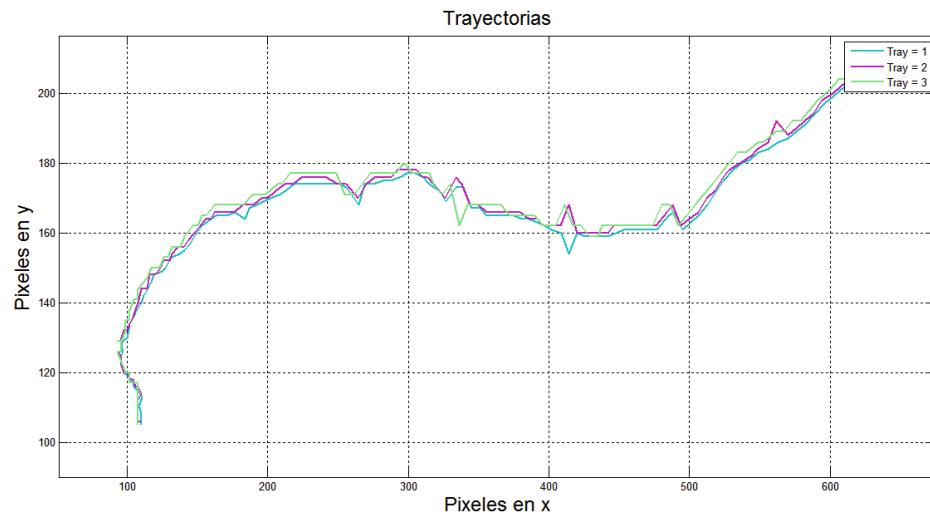
(b) Posición

Fig. 4.20: Detección color rojo

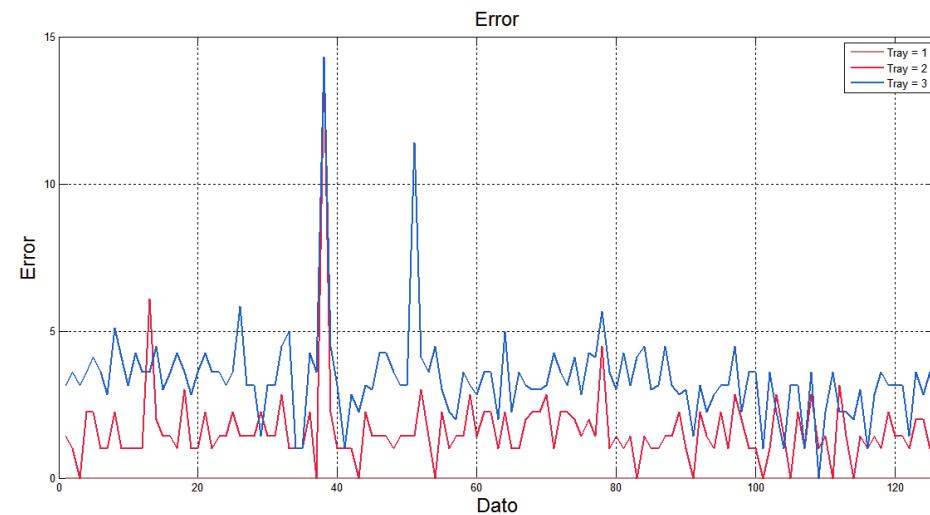
#### 4.2.1. Comportamiento de la detección según la resolución de la imagen

Para analizar la detección según la resolución de la imagen, se procesa un vídeo con el movimiento de una pelota, adquiriendo las posiciones a una resolución base de 800x600 píxeles, el cual se toma como referencia para realizar un análisis comparativo con el resto. Se procede a escalar las dimensiones de los cuadros del vídeo por factores de 2 (400x300), 3 (266x200), 10 (80x60), 15 (53x40), 20 (40x30) y 30 (26x20) y se trata de adquirir la posición de la pelota con el algoritmo desarrollado (figuras 4.21a, 4.22a, 4.23a), posteriormente se obtiene el error comparativo del seguimiento con cada resolución (figuras 4.21b, 4.22b, 4.23b),

con respecto al vídeo base de 800x600.

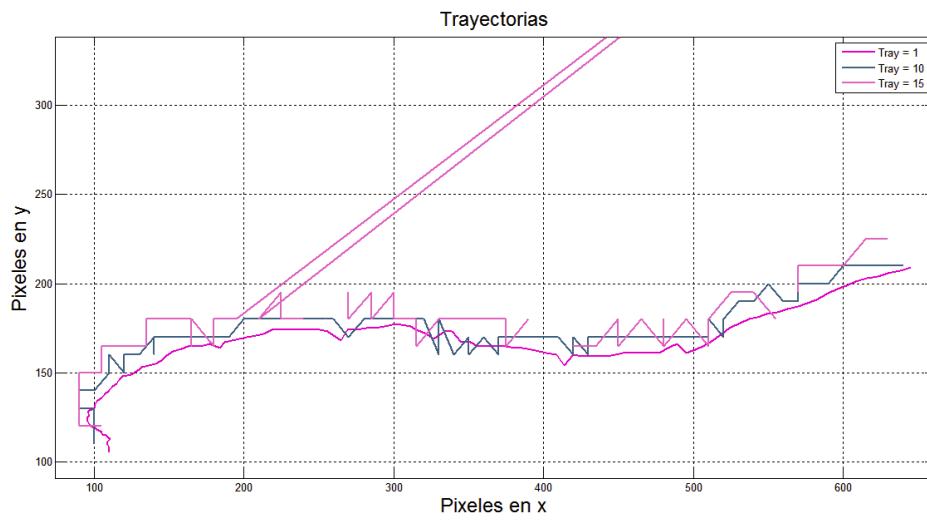


(a) Detección de la posición de la pelota. Escalados 1, 2 y 3

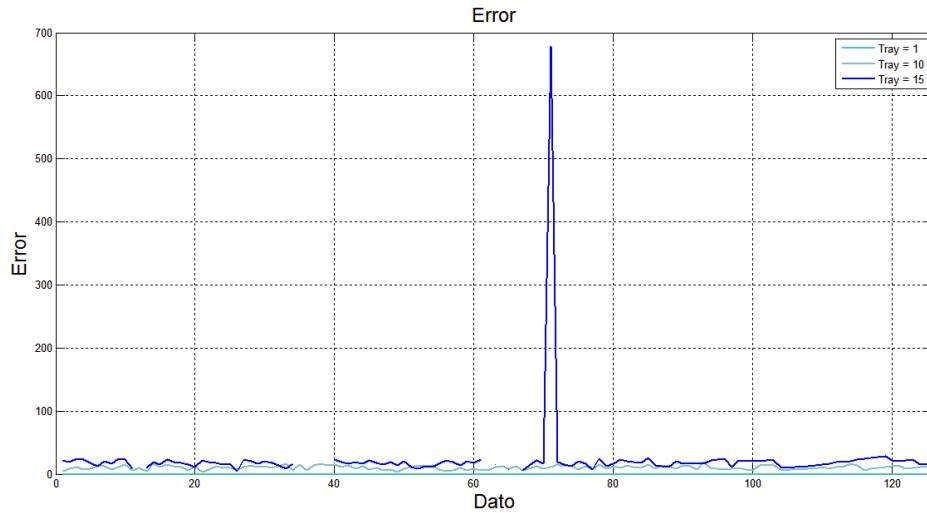


(b) Error comparativo a imagen original. Escalados 1, 2 y 3

*Fig. 4.21: Comparación factores de escalados 1, 2 y 3*

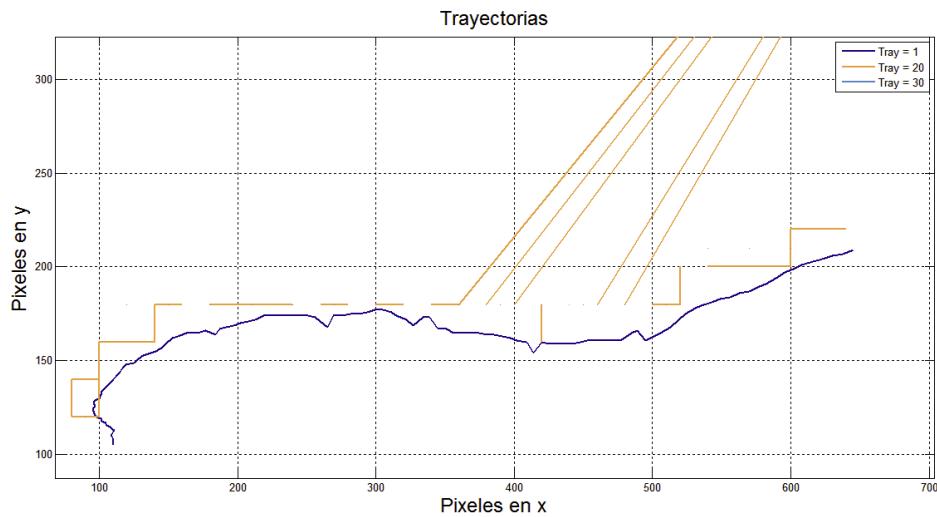


(a) Detección de la posición de la pelota. Escalados 1, 10 y 15

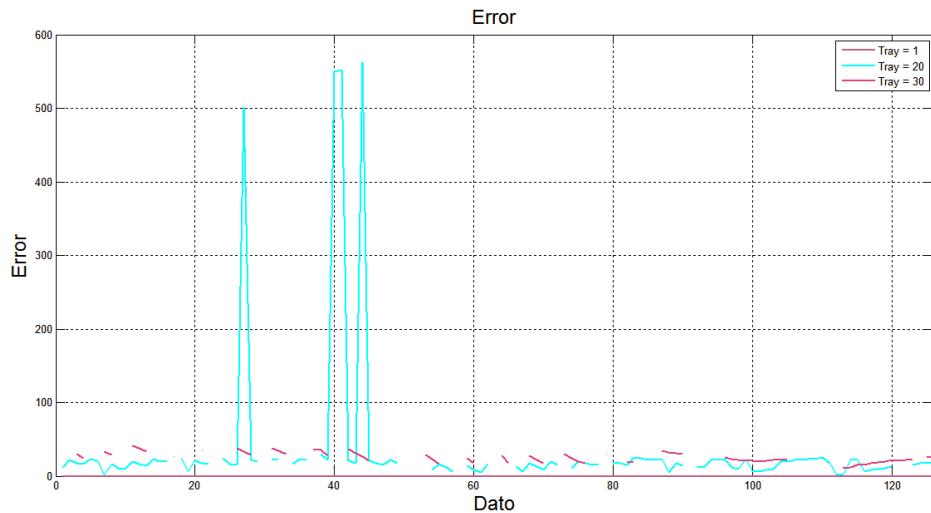


(b) Error comparativo a imagen original. Escalados 1, 2 y 3

*Fig. 4.22:* Comparación factores de escalados 1, 10 y 15



(a) Detección de la posición de la pelota. Escalados 1, 20 y 30



(b) Error comparativo a imagen original. Escalados 1, 20 y 30

*Fig. 4.23:* Comparación factores de escalados 1, 20 y 30

Observamos en las imágenes 4.22 y 4.23 que existe pérdida en el seguimiento del balón donde la señal no es continua, y además las señales poseen saltos abruptos que corresponden a mediciones incorrectas en la detección, donde el ruido de la imagen es capturado como la ubicación donde está la pelota.

La tabla 4.4 muestra los valores de promedio y varianza de las señales del error en cada una de las resoluciones planteadas. Se observa que desde un factor 10 en adelante el error es mayor a 10 píxeles con respecto a la imagen original, por lo que no resulta confiable usar resoluciones tan bajas. Además se observa que los factores 15, 20 y 30 pierden la posición de la pelota resultando en no aptos para la detección. Se destaca que bajar la resolución en un factor de 2 no afecta de gran forma al seguimiento, y constituye una baja en la carga computacional considerable al disminuir a 1/4 la cantidad de píxeles de la imagen.

*Tab. 4.4: Comparación de seguimiento, variando la resolución de imagen*

Factor de Escalamiento	Resolución (Píxeles)	Seguimiento Continuo	Error Promedio	Varianza
1	800 x 600	Si	-	-
2	400 x 300	Si	1.623	1.41
3	266 x 200	Si	3.370	1.57
10	80 x 60	Si	10.359	3.21
15	53 x 40	No	64.83	162.58
20	40 x 30	No	108.53	208.65
30	30 x 20	No	301.64	297.95

#### *4.3. Predicción de las condiciones de juego*

Para predecir la posición donde debe situarse el arquero para evitar las anotaciones en portería, se diseña una red ANFIS con el método “SubClustering” que crea una cantidad de reglas igual al número de grupos creados. Se realizan pruebas para medir el nivel de error luego de entrenar el algoritmo, variando la cantidad de puntos previos de la pelota, cantidad de reglas, cantidad de datos de entrenamiento, efecto del rango de las funciones de membresía, el comportamiento antes y después de entrenar la red y el desempeño ante la existencia de ruido en las mediciones.

#### *4.3.1. Variación del número de entradas*

La primera prueba, consta de utilizar distintas cantidades de posiciones previas de la pelota como entrada al sistema, midiendo el error con y sin ruido en cada uno de los casos para una cantidad fija de datos de entrenamiento. Cabe decir que el ruido utilizado para las mediciones fue de valores de -5 a +5 píxeles con una distribución normal para los puntos previos de las trayectorias. Los resultados se expresan en la tabla 4.5, en la que se observa que el error cuadrático medio disminuye al acercarse a 7 pares de datos en la entrada del sistema ANFIS, para el caso con ruido en las mediciones, y cercano al mejor comportamiento sin ruido. Por lo tanto se mantendrá un diseño de 7 posiciones previas x,y en la entrada del sistema para las siguientes etapas.

Tab. 4.5: Comportamiento del error variando el número de entradas

<b>Entradas (Pares de Coordenadas)</b>	<b>EMC Sin ruido</b>	<b>EMC Con ruido</b>	<b>Número de reglas</b>
2	44.93	102.43	23
3	43.39	78.05	23
5	43.50	65.12	23
7	43.63	52.40	23
9	44.15	58.10	23
11	45.62	60.08	23
13	53.42	64.40	23

#### *4.3.2. Variación del número de reglas*

En el siguiente paso, se mide el comportamiento del sistema al cambiar la cantidad de reglas asociada a la red, lo cual es expuesto en la tabla 4.6, donde se utilizaron 2, 5, 10, 20, 30 y 40 reglas. Obtenemos que el mejor comportamiento se logra al utilizar 10 reglas en el sistema difuso para el caso de las mediciones con ruido y 20 para las mediciones sin ruido.

Dado que un sistema con una menor cantidad de reglas y funciones de membresía permiten una mayor interpretación de las operaciones internas de la red, se escoge el modelo de 10 reglas al tener el menor EMC con ruido, el cual internamente también contendrá una menor cantidad de operaciones matemáticas a ejecutar reduciendo su tiempo de procesamiento.

*Tab. 4.6: Comparación del sistema según el número de reglas*

Número de Reglas	ECM sin ruido	ECM con ruido
2	59.07	61.50
5	47.49	52.54
10	44.70	48.89
20	43.01	51.78
30	43.20	58.43
40	49.42	57.50

#### 4.3.3. Variación del número de datos de entrenamiento

La tercera prueba identifica el comportamiento del sistema difuso cuando se entrena con una mayor cantidad de datos, evaluando el error cuadrático medio se obtiene la cantidad que más reduce este error. En esta medición se utilizaron hasta 5000 trayectorias distintas de entrenamiento con 500 de evaluación cada una, con y sin error en los datos de entrenamiento, utilizando 7 pares de datos de entrada y 10 reglas. Los resultados (tabla 4.7) muestran que un sistema con 4500 trayectorias de entrenamiento posee el mejor resultado con ruido en los datos y muy cercano al mejor resultado (5000 trayectorias) cuando se evalúa las mediciones sin ruido. Se escoge el modelo de 4500 datos.

Tab. 4.7: Error cuadrático medio para distintas cantidades de datos con y sin ruido, en sistema con 7 pares de entradas

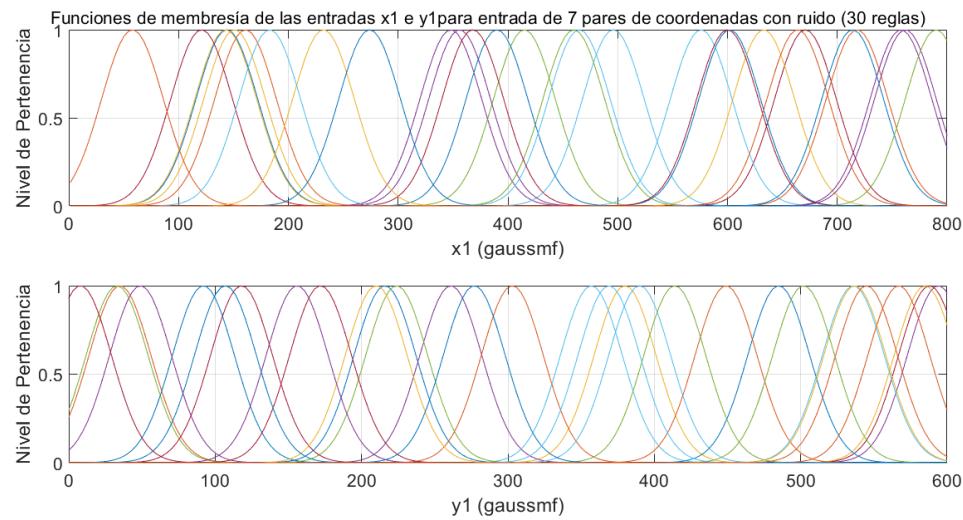
Número de Trayectorias de entrenamiento	ECM sin ruido	ECM con ruido
100	566.33	815.32
500	73.60	64.83
1000	51.85	57.26
1500	49.14	56.38
2000	47.01	52.30
2500	46.00	51.73
3000	45.24	49.94
3500	44.77	48.76
4000	44.35	50.68
4500	44.00	46.73
5000	43.94	49.46

#### 4.3.4. Variación del rango en las funciones de membresía

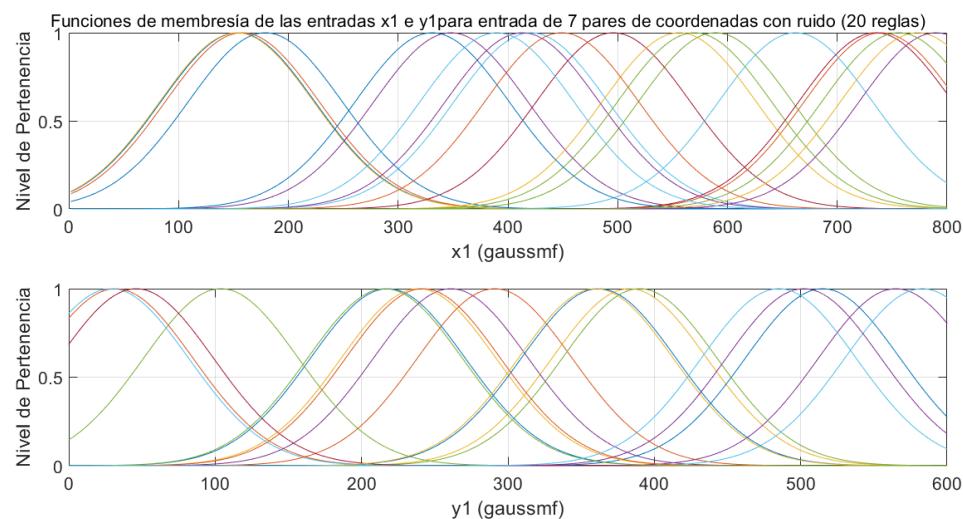
La cuarta prueba que se realiza, es la evaluación al diminuir el ancho de las funciones de membresía, disminuyendo el rango de influencia del cluster (“ClusterInfluenceRange”) el cual crea grupos más pequeños y en mayor cantidad produciendo más reglas al sistema. Se modifica el método de generación de reglas para que el número de estas sea proporcional al tamaño de los grupos. Se utilizan tres mediciones diferentes para grupos que ocupen un 10 %, 25 % y 50 % del rango total (rango de [0 800] en “x”, [0 600] en “y”) calculando el error cuadrático medio con respecto a la salida deseada. Para 7 pares de entradas con 4500 trayectorias de entrenamiento y 500 de validación se obtiene la tabla 4.8 y se observa cómo cambia la forma de los grupos en las figuras 4.23. Se aprecia además que el sistema con menor error se logra al utilizar un 50 % del rango en las funciones de membresía para el caso con ruido de medición, lo que será utilizado para la estructura de la red.

Tab. 4.8: Error según la función de membresía

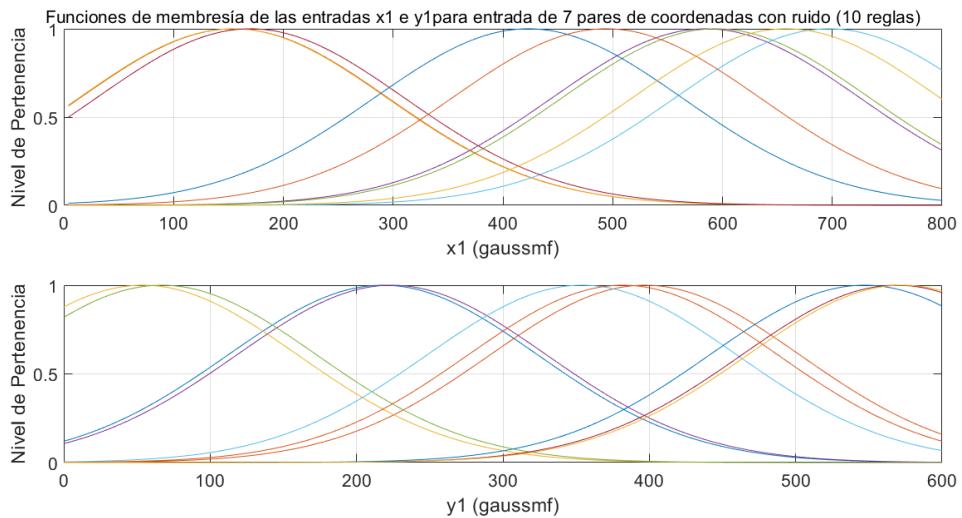
Tamaño de grupos	ECM sin ruido	ECM con ruido	Número de reglas
10 % del Rango	46.15	50.88	30
25 % del Rango	43.96	48.51	20
50 % del Rango	44.00	46.73	10



(a) Caso 1: 10 % del rango



(b) Caso 2: 25 % del rango

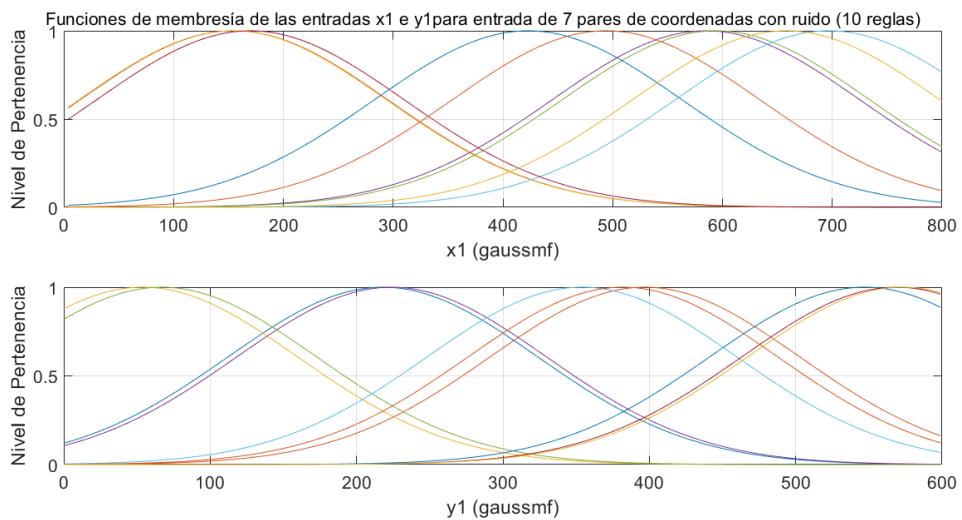


(c) Caso 3: 50 % del rango

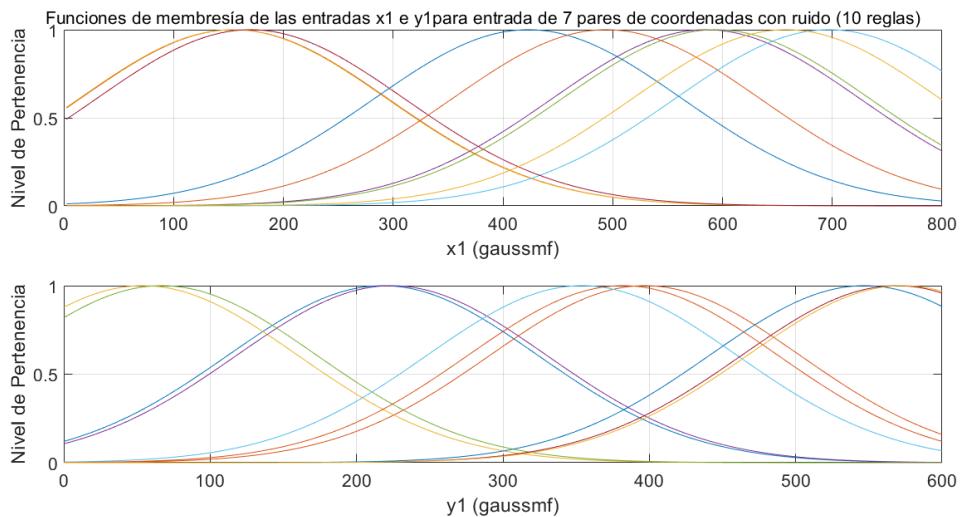
*Fig. 4.23: Modificación en las funciones de membresía*

#### 4.3.5. Evaluación del error antes y después del entrenamiento

Como última instancia, se comprueba el comportamiento del sistema difuso antes y después de ser entrenado con los mejores resultados de cada prueba (proporción del 50 % de las funciones de membresía, 7 pares de coordenadas en la entrada, 4500 datos de entrenamiento y 10 reglas). De las figuras 4.24, observamos que las funciones de membresía se mantienen igual antes y después de entrenarse, producto del método “SubClustering”, ya posee las características suficientes como para dar una buena asignación de las funciones. Pero lo que sí se ve modificado, es el valor de los parámetros consecuentes de la red, siendo estas las funciones de salida que darán el valor final al resultado del sistema. Esta modificación a los valores de los parámetros por medio del entrenamiento, trae como consecuencia un mejor comportamiento en la evaluación de nuevos datos, lo cual se puede apreciar en la tabla 4.9, donde se disminuye del error cuadrático medio al evaluar nuevas entradas.



(d) Funciones de membresía antes de entrenarse



(e) Funciones de membresía después de entrenarse

Fig. 4.24: Comparación de las funciones de membresía antes y después de entrenarse (con ruido)

Tab. 4.9: Comparación antes y después de entrenarse

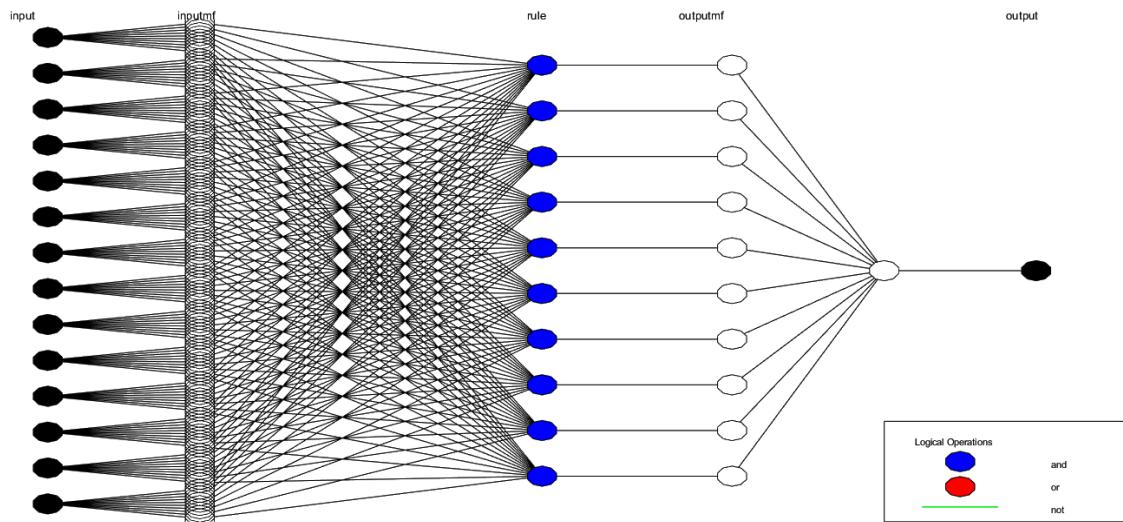
	<b>ECM sin ruido</b>	<b>ECM con ruido</b>
Antes de entrenarse	43.94	92.53
Después de entrenarse	44.00	47.35

Los resultados de la tabla 4.9 indican que si bien el sistema puede funcionar mejor sin entrenamiento al existir un ambiente sin perturbaciones, al incorporar la incertidumbre y el ruido a los datos, una red ANFIS entrenada responde de mejor manera que la anterior.

Otro aspecto importante a señalar, es que se observan funciones de pertenencia que se superponen, lo cual indica que la cantidad de estas puede ser disminuida, no obstante los resultados indicaron que una menor cantidad de reglas (y por ende funciones de membresía) aumentaban el error del sistema. Una de las causas, es que la cantidad de funciones es igual para las coordenadas “x” e “y” que poseen distinto rango de valores, lo que genera que las funciones cubran distintas proporciones, necesitando una mayor o menor cantidad de grupos dependiendo de la coordenada.

#### *4.3.6. Estructura de la red ANFIS*

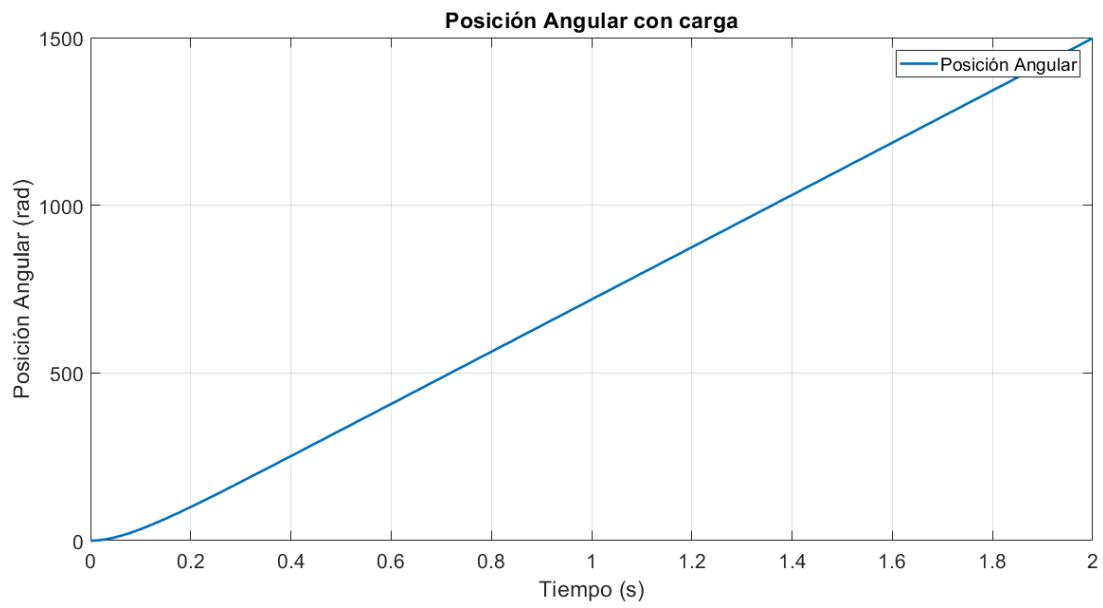
Finalmente, la estructura que responde mejor según las pruebas realizadas, corresponde a un sistema difuso con 14 entradas compuesta por 7 pares de datos x,y correspondientes a las posiciones previas de la pelota. Luego cada una de estas entradas es evaluada en 10 funciones de membresía del tipo Gaussiano en la capa 1, para luego identificarse según las 10 reglas existentes en el sistema (capa 2) y establecer el comportamiento que dará como resultado la posición en “y” donde el arquero deberá situarse (figura 4.25).



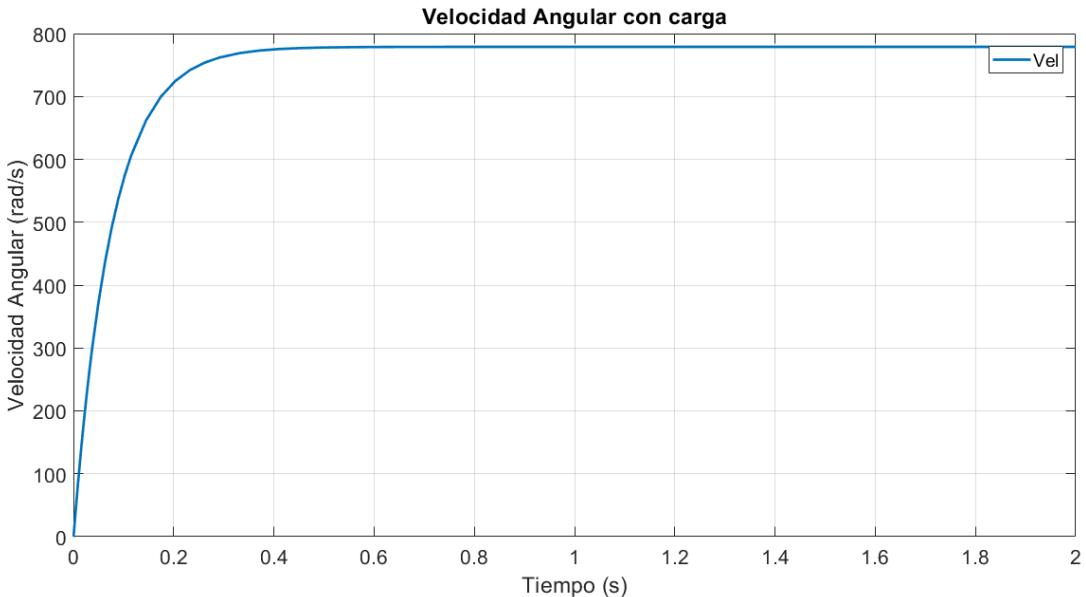
*Fig. 4.25: Estructura del sistema difuso*

#### 4.4. Motor DC y Control de posición

Para la simulación del motor DC se tomara en cuenta las características del motor comercial Maxon descrito en el capítulo anterior. Mediante el modelo dinámico, es posible evaluar el motor al conectarlo con un voltaje de 24 [V], y obtener las características de posición en radianes (figura 4.26a) y velocidad en radianes por segundo (figura 4.26b), la cual alcanza un valor de 778 rad/s con carga, equivalente a 7429 RPM, demorando 0.5 segundos en alcanzar la máxima velocidad.



(a) Curva de posición del motor con carga en Radianes



(b) Curva de velocidad del motor con carga en Radianes/segundo

Fig. 4.26: Mediciones de velocidad y posición del motor

Con el modelo del motor, se procede a calcular el controlador PD para lograr seguimiento a una señal de referencia. Para ello utilizamos la función de transferencia del motor de lazo abierto, considerando el torque de carga como una perturbación al sistema y mediante la he-

rramienta Sisotool que provee Matlab, encontramos el controlador PD mediante asignación de polos.

La función de transferencia del motor puede ser expresada como:

$$G(s) = \frac{1,2107 * 10^6}{s(s + 2847)(s + 12,46)} \quad (4.1a)$$

$$G(s) = \frac{K_G}{s(s + P1)(s + P2)} \quad (4.1b)$$

$$(4.1c)$$

Donde  $K_G$  corresponde a la amplitud de la planta  $G(s)$ ,  $P1$  corresponde al primer polo (-2847) y  $P2$  al segundo polo (-12.46) del modelo del motor  $G(s)$ .

En cuanto al modelo de control PD, lo podemos expresar de la forma:

$$C_{PD}(s) = K(s + b) \quad (4.2a)$$

Al obtener la función de transferencia de lazo cerrado obtenemos:

$$H(s) = \frac{C_{PD}(s) * G(s)}{1 + C_{PD}(s) * G(s)} \quad (4.3a)$$

$$H(s) = \frac{K(s + b) * K_G}{s(s + P1)(s + P2) + K(s + b)K_G} \quad (4.3b)$$

Al utilizar el valor de  $b$  en función del polo más lento del sistema ( $P2 = -12.46$ ), podemos obtener una mayor velocidad de respuesta al eliminar el polo dominante, es decir la dinámica

más lenta, lo cual entrega una función de transferencia de lazo cerrado:

$$H(s) = \frac{K_G K}{s^2 + sP1 + K_G K} \quad (4.4a)$$

$$H(s) = \frac{K * 1,2107 * 10^6}{s^2 + s2847 + K * 1,2107 * 10^6} \quad (4.4b)$$

Finalmente, para observar el comportamiento del controlador en lazo cerrado a medida que varía K, se utiliza la herramienta SISOTOOL de Matlab, la cual permite modificar el valor de la ganancia para alcanzar un comportamiento críticamente amortiguado como se muestra en la figura 4.27, llegando a tener ambas raíces iguales, quedando de la forma:

$$C_{PD}(s) = 1,6737(s + 12,46) \quad (4.5a)$$

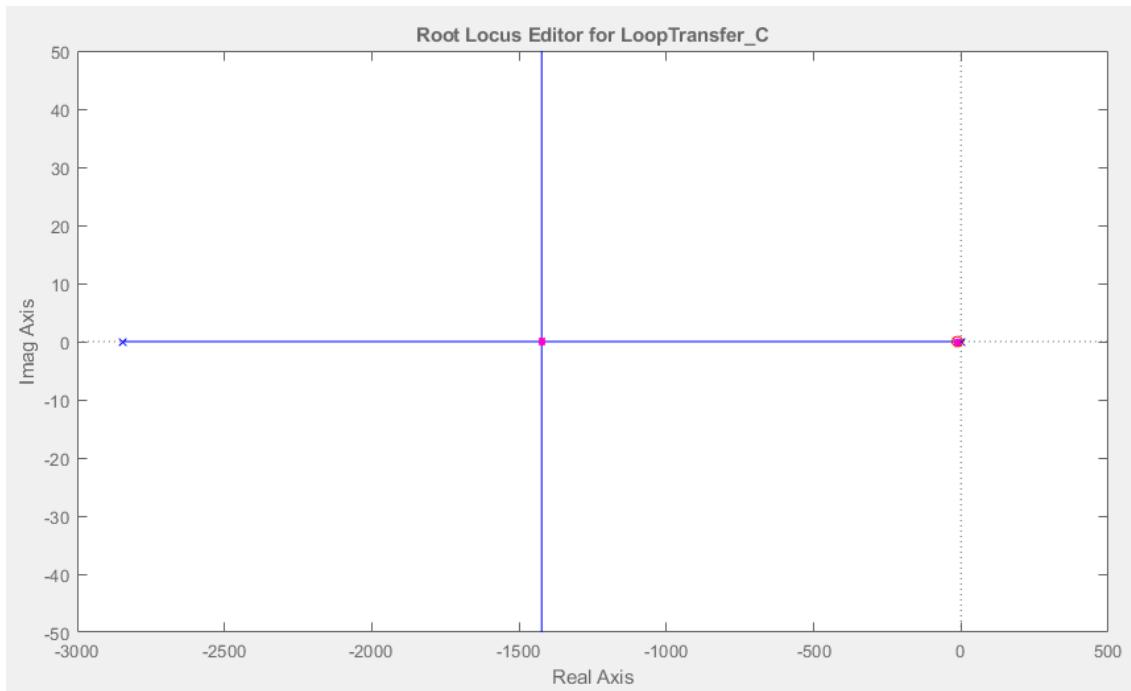


Fig. 4.27: Root Locus

Como resultado se obtiene la función de transferencia del sistema de la forma:

$$H(s) = \frac{2,0264e06}{(s + 1424)(s + 1423)} \quad (4.6a)$$

Quedando las raíces de lazo cerrado casi iguales, justo antes de volverse imaginarias lo cual provocaría oscilaciones dentro del sistema. Además, se previene la existencia de “overshoot” que puede resultar dañino para el movimiento del motor, si se le asigna una posición a los extremos de la cancha, ya que tratará de sobrepasar los límites físicos del campo de juego. Como resultado de este diseño, el sistema posee una respuesta a escalón unitario con error en estado estacionario igual a cero, ganancia unitaria a frecuencia cero, tiempo de levantamiento de 2.36 [ms] (90 % alcanzado del valor final) y un tiempo de asentamiento de 4.1 [ms] (bajo el 2 % de error sobre el valor final) (figura 4.28).

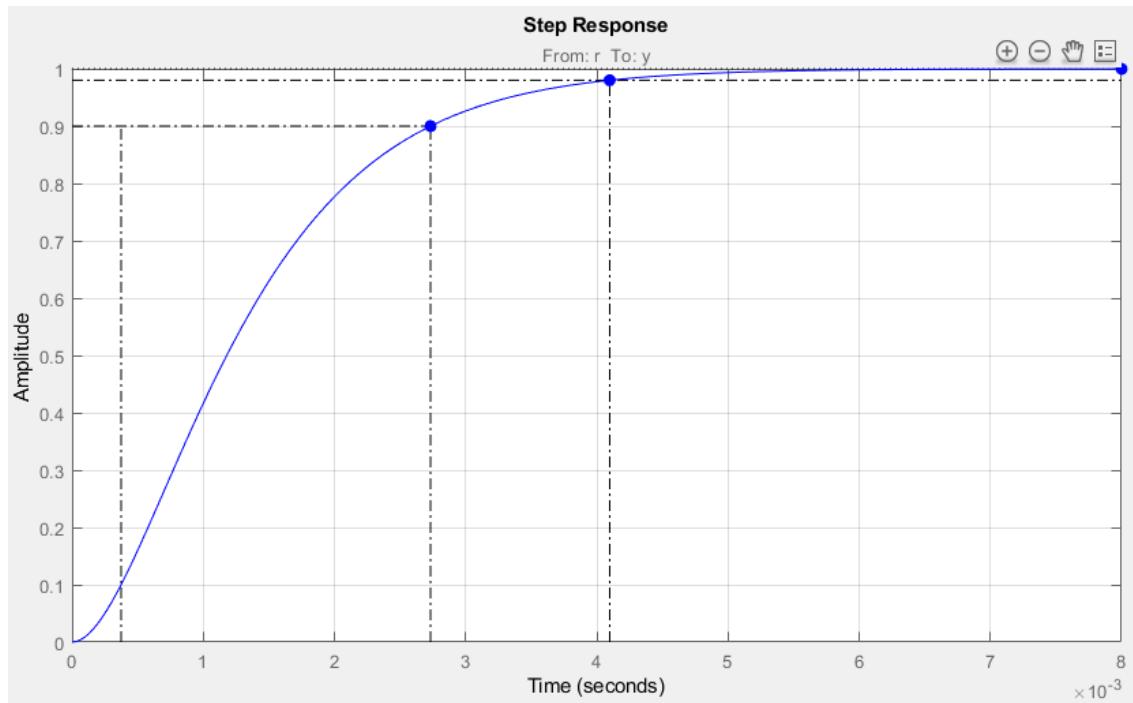


Fig. 4.28: Respuesta a escalón

Aunque la respuesta a escalón unitario muestra un tiempo de respuesta del orden de los milisegundos, este modelo no toma en cuenta las restricciones físicas existentes del motor

DC. Una restricción importante del motor, es que no puede tener una entrada de voltaje mayor a las especificadas por el fabricante, por lo tanto existe un restricción en la señal de salida del controlador de  $+ - 24$  [V].

Finalmente, se agrega el controlador junto con las restricciones al modelo del motor, pero se debe modificar las unidades con que se está trabajando, ya que las coordenadas enviadas por el simulador corresponden a valores en píxeles (desde 0 a 600) mientras que el controlador está funcionando con referencias en radianes. Recordando que el sistema para mover al arquero, corresponde a un mecanismo de arrastre, se toman en cuenta las medidas de  $0.96 \times 0.72$  [ $m^2$ ] equivalentes a  $800 \times 600$  píxeles, y a  $R$  como el radio de giro del motor equivalente a  $0.02$  [m].

Distancia por pixel recorrido:

$$\frac{0.72}{600} \frac{m}{pixel} = 1,2 * 10^{-3} \frac{m}{pixel} \quad (4.7)$$

Distancia recorrida por cada vuelta:

$$2 * \pi * R = 2 * \pi * 0,02 \frac{m}{vuelta} \quad (4.8)$$

Distancia recorrida por cada radian:

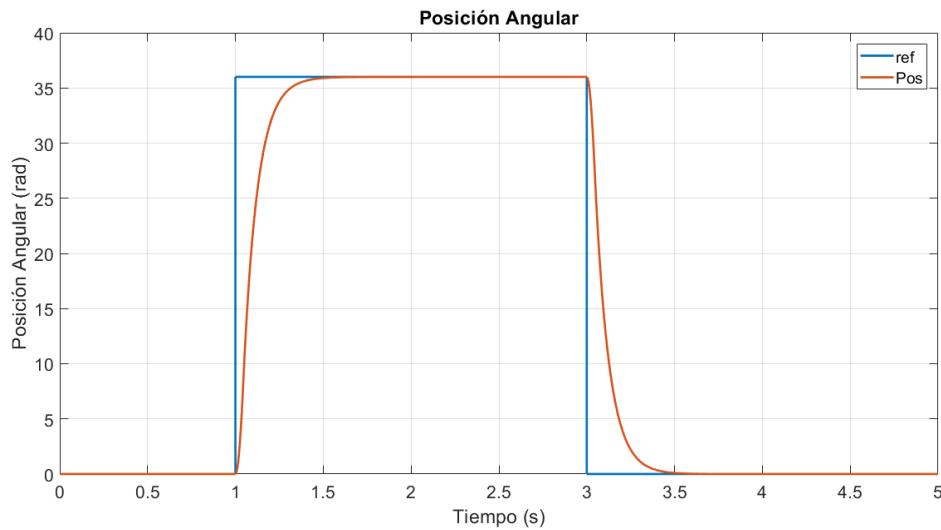
$$2 * \pi * 0,02 \frac{m}{vuelta} \frac{1}{2\pi} \frac{vuelta}{rad} = 0,02 \frac{m}{rad} \quad (4.9)$$

Radianes recorridos por cada pixel:

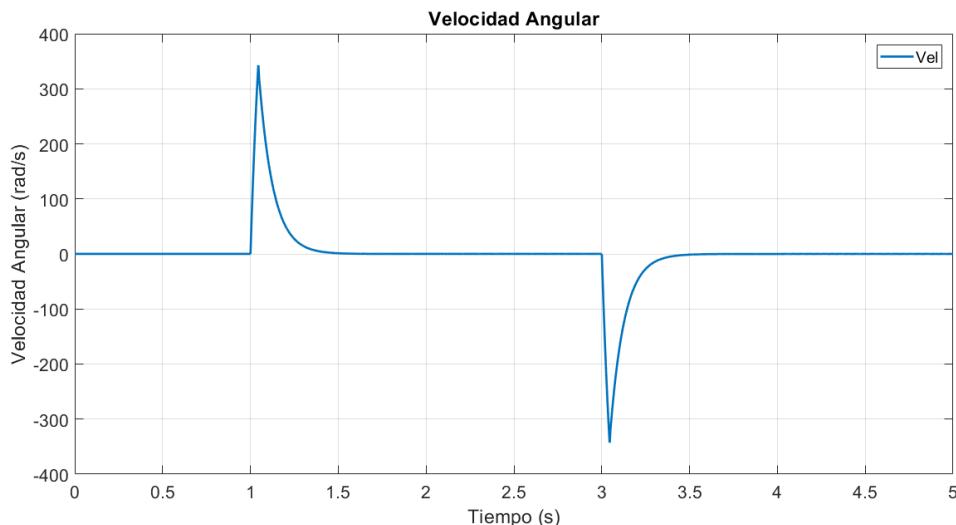
$$1,2 * 10^{-3} \frac{m}{pixel} \frac{1}{0,02} \frac{rad}{m} = 0,06 \frac{rad}{pixel} \quad (4.10)$$

Con este resultado tenemos una conversión directa desde los píxeles de referencia que nos entrega el procesamiento de imágenes, a los radianes que se necesita mover el motor.

Al añadir el controlador al modelo con las limitaciones incluidas, se observa el comportamiento ante una referencia que cambia de 0 a 36 [rad] (equivalente a 600 píxeles) y de vuelta a 0, es decir desde un extremo a otro. En las figuras 4.29a y 4.29b se aprecia el correcto seguimiento ante la referencia de entrada, donde el tiempo que tarda en llegar hasta la referencia es de 0.5 segundos y 0.2 en el tiempo de levantamiento, recordando que en el caso anterior era la velocidad que se estaba evaluando, y en este caso es la posición.



(a) Posición angular del motor. Referencia en azul. Posición en rojo



(b) Velocidad angular del motor

*Fig. 4.29:* Respuestas del motor con control PD

#### 4.5. Simulación

Para lograr simular el comportamiento que permite posicionar al arquero en el lugar adecuado, se implementa en Simulink un sistema de tres etapas, la primera crea nuevas posiciones utilizando como parámetros una posición, velocidad y des-aceleración inicial. La segunda

etapa utiliza el algoritmo de ANFIS para predecir la posición donde deberá ubicarse el arquero, donde se deben ingresar las 7 posiciones previas de la trayectoria del balón. Y finalmente, se utiliza el motor con control PD para moverse a la posición de referencia. El modelo utiliza los bloques “sFunction” de Simulink [24], los cuales permiten programar código dentro de la simulación, lo cual facilita añadir las etapas previamente descritas, como se puede observar en la figura 4.30, existe un bloque adicional llamado “sField”, que permite visualizar los cambios que se van desarrollando dentro del juego, a medida que transcurre el tiempo.

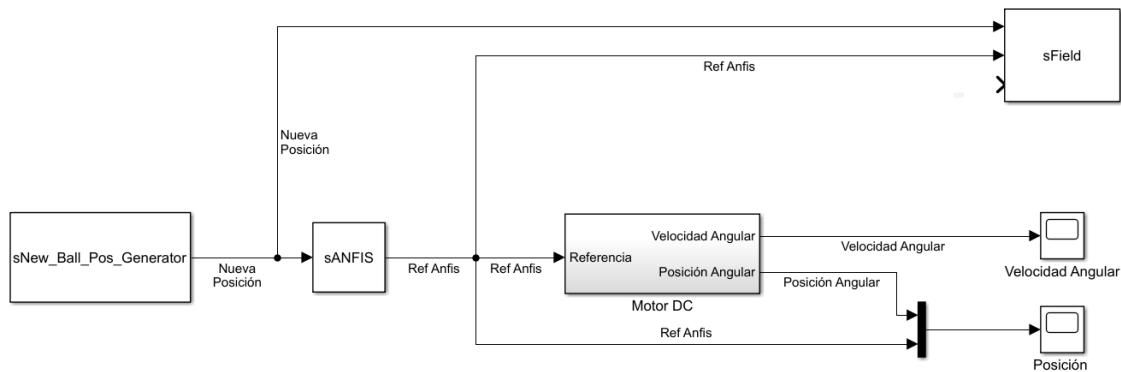
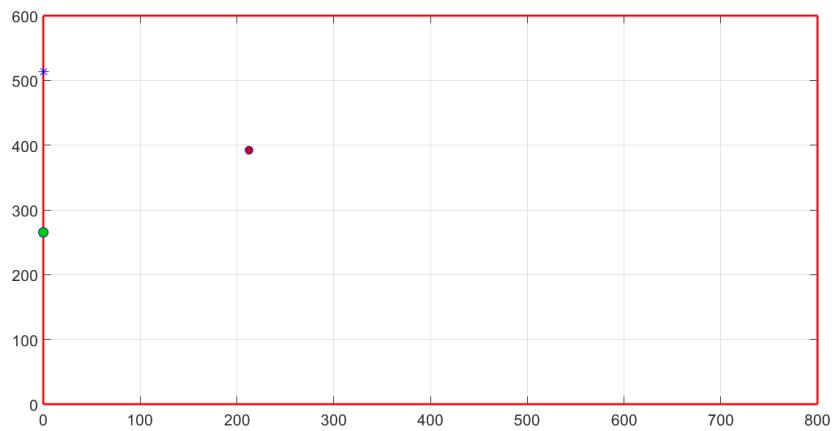
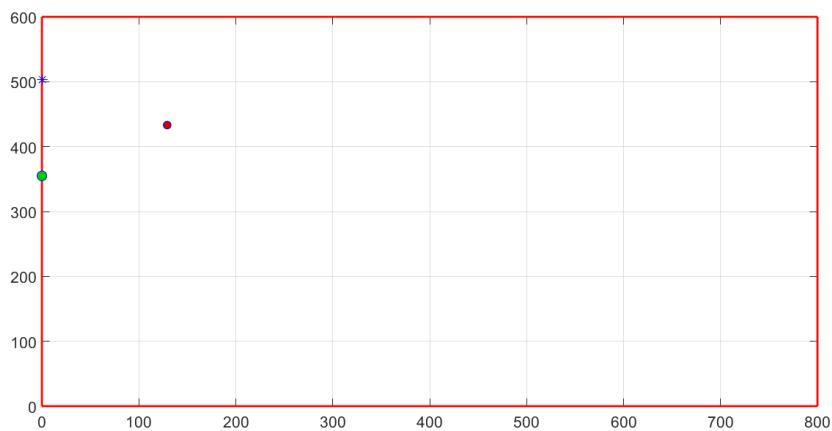


Fig. 4.30: Respuesta a escalón

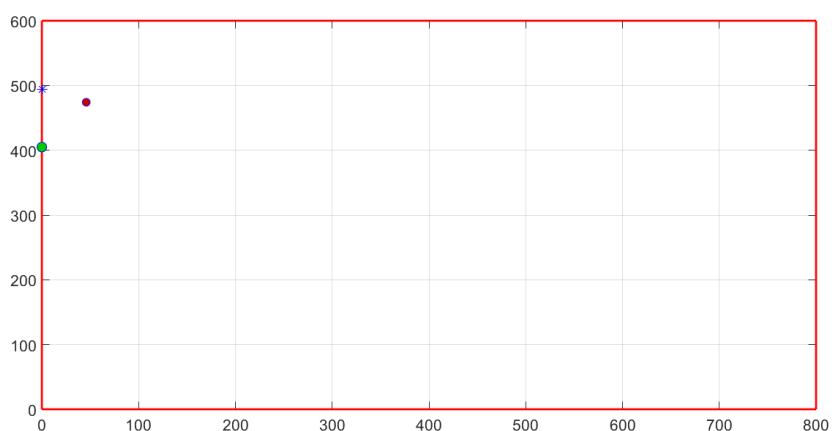
En las figuras 4.31a, 4.31b y 4.31c observamos tres instantes distintos, donde se muestra la posición de la pelota en un círculo rojo, la referencia entregada por el algoritmo ANFIS como un asterisco azul, y la posición del arquero como un círculo en verde.



(a) Posición a



(b) Posición b



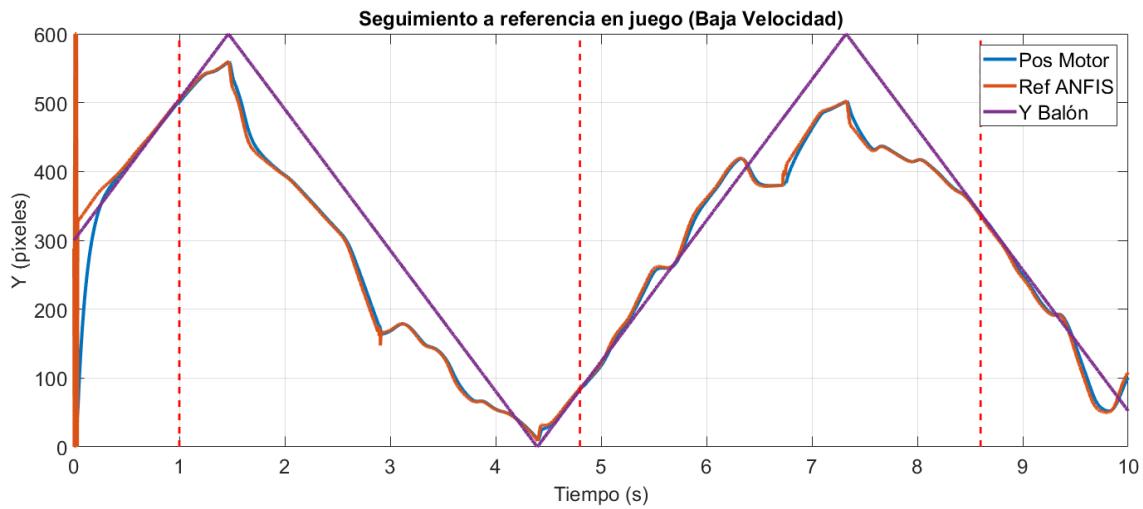
(c) Posición c

Fig. 4.31: Instantes durante la simulación. Circulo rojo: Posición de la pelota. Asterisco azul: Referencia entregada por ANFIS, Circulo verde posición de arquero

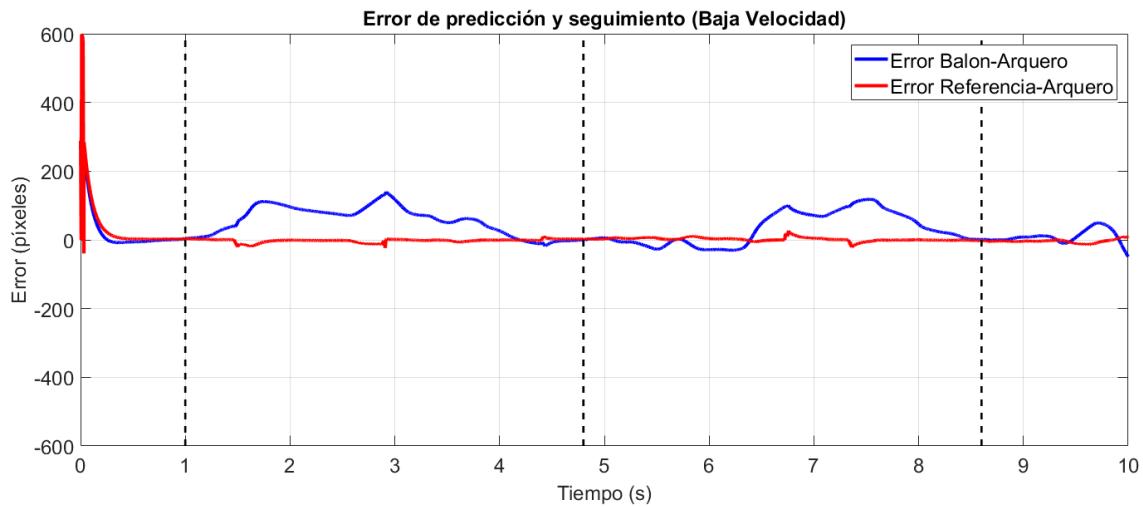
En las figuras 4.32, 4.34, 4.36 muestran los comportamientos de la posición del motor, la referencia ANFIS y la posición del balón, junto con las 4.33, 4.35, 4.37 que indican el error entre la posición del balón y el arquero, y el error entre la referencia y el arquero para una velocidad de 0.5, 1 y 5 m/s respectivamente, donde se logra apreciar un comportamiento más fluido en el movimiento de la referencia y con bajas perturbaciones, al tener velocidades medias y bajas. Para estos casos, no fue aplicado roce sobre el balón, ni des-aceleración al rebotar sobre los límites del campo de juego. Además, se posiciona una linea de color negro, en el instante cuando la pelota golpea el extremo del arquero, siendo este lugar donde la posición del motor y del balón deben ser iguales para poder bloquear el tiro a portería.

Con los resultados de la simulación, se puede señalar que la referencia entregada por ANFIS no siempre corresponde a una respuesta acertada, donde el mayor error se produce a máxima velocidad. En comparación, al tener velocidades medias y bajas, la red ANFIS va mejorando su referencia situándose sobre el lugar de impacto.

Por otro lado, se aprecia el correcto seguimiento del motor DC controlado, aunque en el caso de alta velocidad, este no alcanza a posicionarse en la referencia antes de que el balón impacte, lo cual si se consigue a media y bajas velocidades.



*Fig. 4.32:* Seguimiento a referencia con tiro de 0.5 m/s. Lineas segmentadas indican el instante en que la pelota llega al extremo del arquero.



*Fig. 4.33:* Error con tiro de 0.5 m/s. Lineas segmentadas indican el instante en que la pelota llega al extremo del arquero.

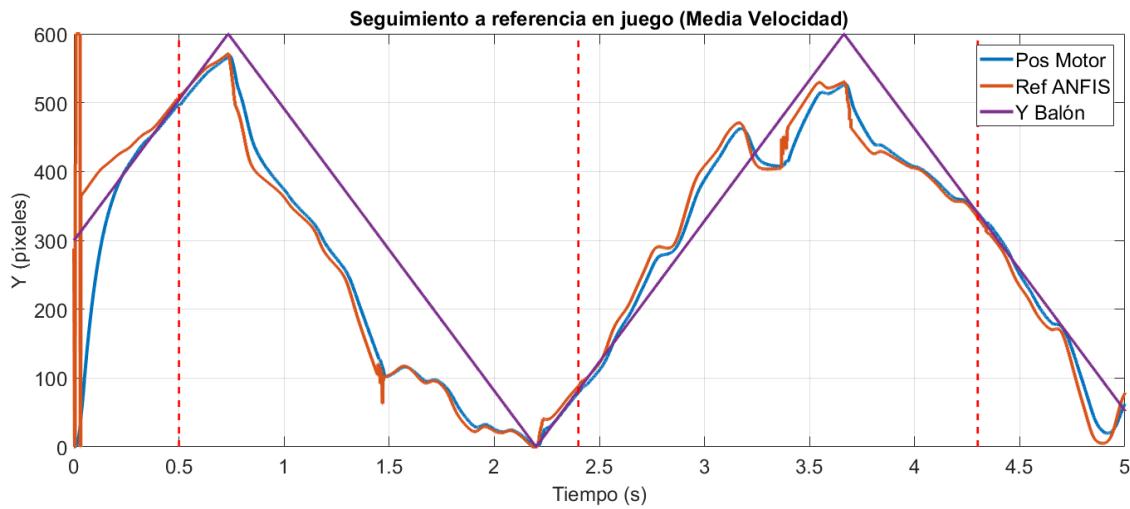


Fig. 4.34: Seguimiento a referencia con tiro de 1 m/s. Lineas segmentadas indican el instante en que la pelota llega al extremo del arquero.

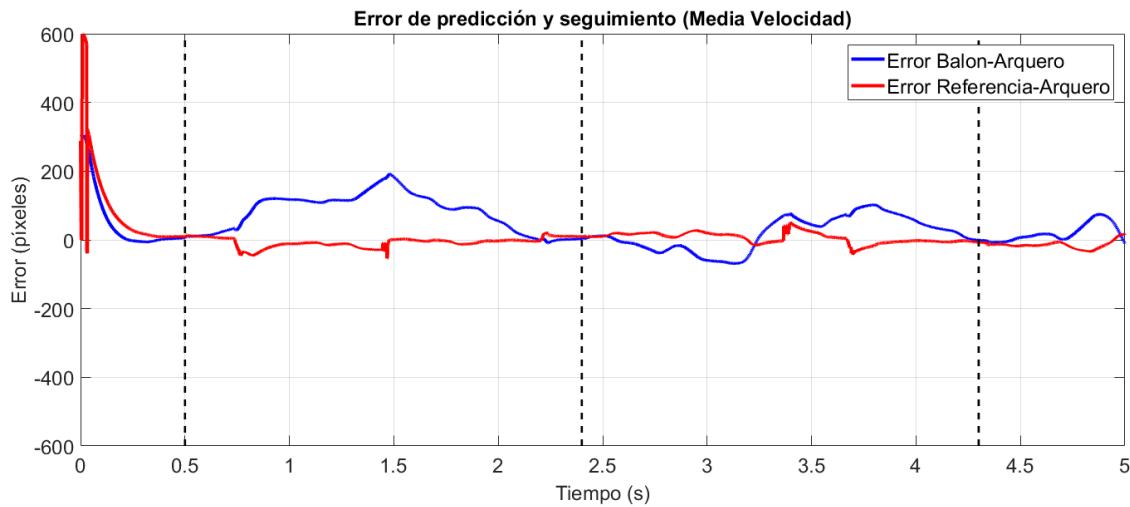


Fig. 4.35: Error con tiro de 1 m/s. Lineas segmentadas indican el instante en que la pelota llega al extremo del arquero.

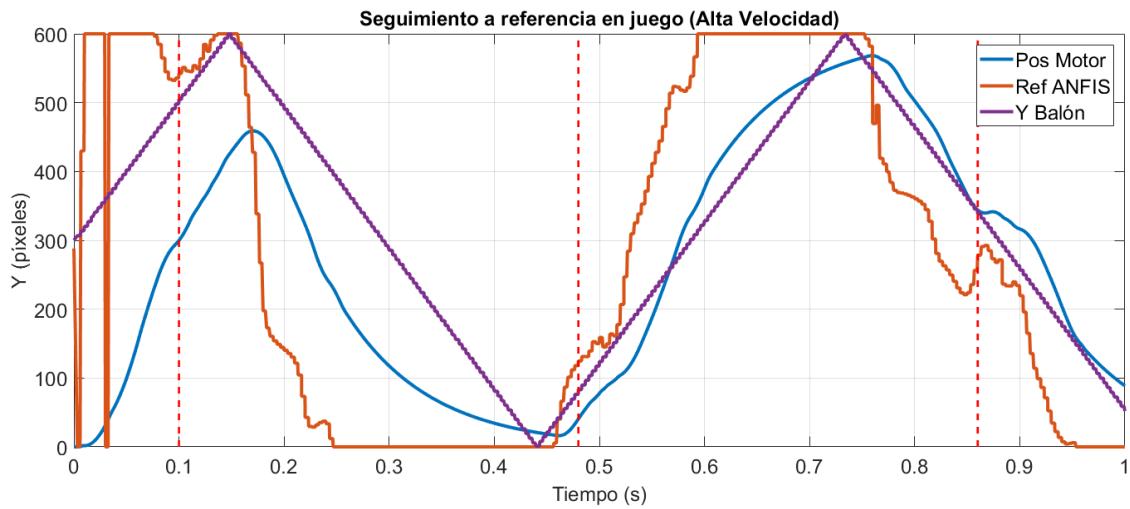


Fig. 4.36: Seguimiento a referencia con tiro de 5 m/s. Lineas segmentadas indican el instante en que la pelota llega al extremo del arquero.

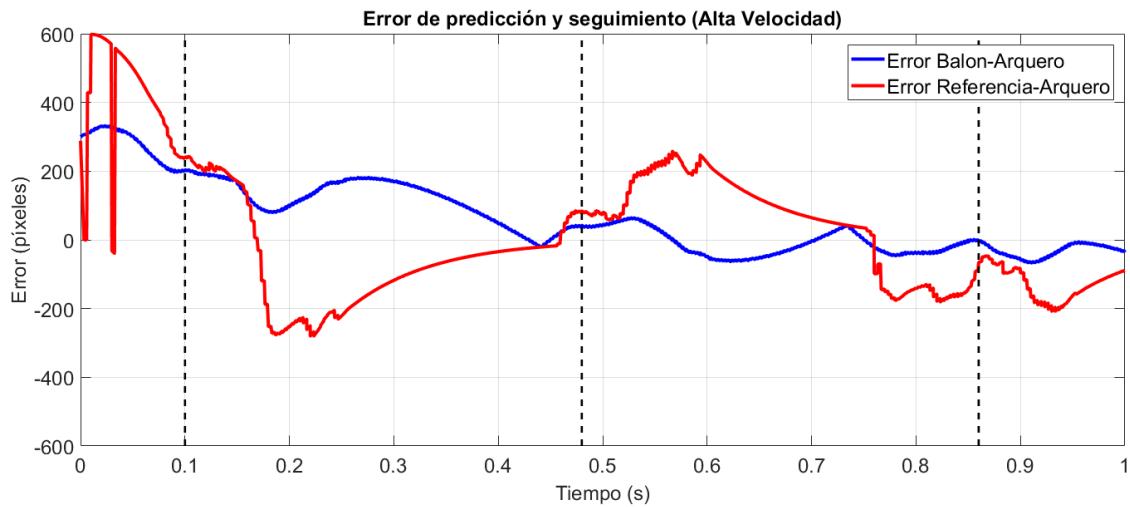


Fig. 4.37: Error con tiro de 5 m/s. Lineas segmentadas indican el instante en que la pelota llega al extremo del arquero.

## 5. CONCLUSIONES

Mediante el desarrollo del presente trabajo de estudio, se puede concluir en primer lugar el cumplimiento del objetivo de crear un sistema computacional simulado que permita posicionar al arquero para bloquear los tiros rivales, siendo un factor relevante para la tasa de éxito la velocidad con que se desplaza la pelota.

En segundo lugar, se destaca el compromiso generado por el ambiente para lograr un buen resultado final, ya que sin la existencia de un entorno controlado, el efecto de sombras e iluminación dinámica, provocarían sobre las imágenes condiciones donde sería difícil adquirir la información del juego de una buena manera, repercutiendo en el resultado final.

Por otra parte, el algoritmo de detección por color de los objetos, para determinar la posición de estos en la imagen, resulta ser efectivo según las pruebas que se realizaron, mas no robusto, ya que depende de la exclusividad del color a ser detectado. Aun así, al ser un ambiente controlado, esto no presenta mayor problema, siendo un método que logra detectar correctamente la posición, bajo las condiciones establecidas durante el estudio.

En cuanto al sistema difuso ANFIS, resulta ser un método avanzado para modelar sistemas no lineales, donde se puede aplicar el conocimiento sobre el proceso a la estructura de la red. Pero una gran cantidad de entradas, funciones de membresía y reglas en el sistema, dificulta de gran manera el entendimiento interno de la red, transformándose en una especie de red neuronal o caja negra. Por otro lado, es probable que en algunos casos no resulte prescindible el entrenamiento de la red, ya que mediante una correcta inicialización, el sistema ya puede lograr un bajo error como se observó al comparar la red antes y después de entrenarse

con los datos sin ruido de medición.

En quinto lugar, se obtiene que el diseño del controlador para el motor DC fue correcto ya que logra realizar el seguimiento ante una referencia dada, logrando cumplir el objetivo deseado, si bien el método empleado no alcanza la mayor velocidad de actuación posible al usar un modelo críticamente amortiguado, si asegura que el mecanismo no tenga un “overshoot”, lo cual generaría problemas al tratar de mover al arquero sobre las esquinas del campo de juego, ya que este golpearía los extremos al tratar de sobrepasar esas posiciones.

Finalmente, se concluye que el desarrollo de la aplicación está sujeta en gran parte a las restricciones que plantean los sensores y actuadores, mientras que una correcta utilización de estos dispositivos está sujeto a la habilidad y estudio con que se escogen e implementan los algoritmos para manipularlos, los cuales sin un estudio detallado de los componentes no hubiera sido posible alcanzar los resultados de este estudio.

## 6. POSIBLES MEJORAS

Como posible mejora, se propone utilizar una detección dinámica cuando se está adquiriendo la posición de la pelota a través de procesamiento de imágenes, esto significa reducir el tamaño de búsqueda y revisar sólo un sector restringido ya que la pelota no se puede mover de un extremo a otro con tanta rapidez. Esto significaría utilizar un ROI (Región de Interés) dinámico para la localización de la pelota, resultando en una menor cantidad de píxeles por procesar y aumentando la velocidad de computación.

Durante las pruebas realizadas para obtener el mejor resultado posible de ANFIS, se señaló que los datos utilizados corresponden a trayectorias generadas aleatoriamente tanto en posición, como en velocidad. Una posible mejora puede ser el uso de un set de datos que contenga posiciones clave para el funcionamiento del sistema, como por ejemplo trayectorias cercanas a la ubicación del arquero, de esta forma se dará mayor prioridad cuando la pelota esté llegando al extremo de interés.

## 7. ANEXO A: IMPLEMENTACIÓN DEL TRABAJO DESARROLLADO

Para la implementación del sistema en un ambiente de trabajo “Online”, se utilizan los datos que fueron capturados durante la etapa de simulación, y se procede a medir los tiempos de ejecución de cada proceso, los cuales se registran en la tabla 7.1. En cuanto a la parte de procesamiento de imágenes se utilizó OpenCV para implementar los algoritmos planteados durante este estudio y además fue implementada una solución en GPU utilizando la herramienta de programación de hardware CUDA 8.0 de NVIDIA, sobre una tarjeta GeForce GT 540M con los mismos algoritmos del caso anterior para lograr procesamiento paralelo sobre la imagen. Por otra parte, ANFIS sólo fue ejecutado por CPU.

Tab. 7.1: Tiempos de procesamiento en CPU y GPU para obtención de posición de la pelota, por cuadro de vídeo.

<b>Proceso</b>	<b>Sub-Procesos</b>	<b>Tiempo de ejecución en CPU (ms)</b>	<b>Tiempo de ejecución en GPU (ms)</b>
Procesamiento de imagen			
	Traspaso de datos desde CPU a GPU	-	0.227
	RGB a HSV y rango de color	2.394	0.910
	Aplicación de filtro	0.427	0.658
	Obtener Centro de masa	2.510	1.612
	Traspaso de datos desde GPU a CPU	-	0.001
	<b>Total</b>	<b>5.331 ms</b>	<b>3.408 ms</b>
ANFIS	-	1.3	-
	<b>Total</b>	<b>1.3 ms</b>	-

Se observa de los tiempos de ejecución, que el tiempo de procesamiento en GPU es menor al de CPU, producto de la ejecución por bloques de píxeles en comparación al procedimiento pixel a pixel realizado por CPU. Por otra parte, también se aprecia que el proceso limitante en este caso será el de procesamiento de imágenes con el mayor tiempo de ejecución por cuadro de vídeo, ya que al utilizar una programación por hilos o “threads”, los procesos con menores tiempos de ejecución quedarán a la espera de la información entrante y cuando la reciban, terminarán antes que una nueva información les sea provista, lo cual puede ser observado en la figura 7.1, donde se ajusta el tiempo de adquisición de imágenes al proceso más lento, que en este caso es el procesamiento de imágenes con un tiempo de 3.4 [ms]. No se toma en cuenta el motor DC, ya que este corresponde a un proceso continuo. Además, se observa basado en los colores la secuencia de ejecución del sistema, lo cual da como resultado un

retardo de 4.7 [ms] desde que se adquiere el cuadro de vídeo y se entrega la referencia. Cabe destacar que el tiempo de adquisición no tarda 3.4 [ms] en ejecutarse, sino que este tiempo es el necesario en adquirir un nuevo frame.

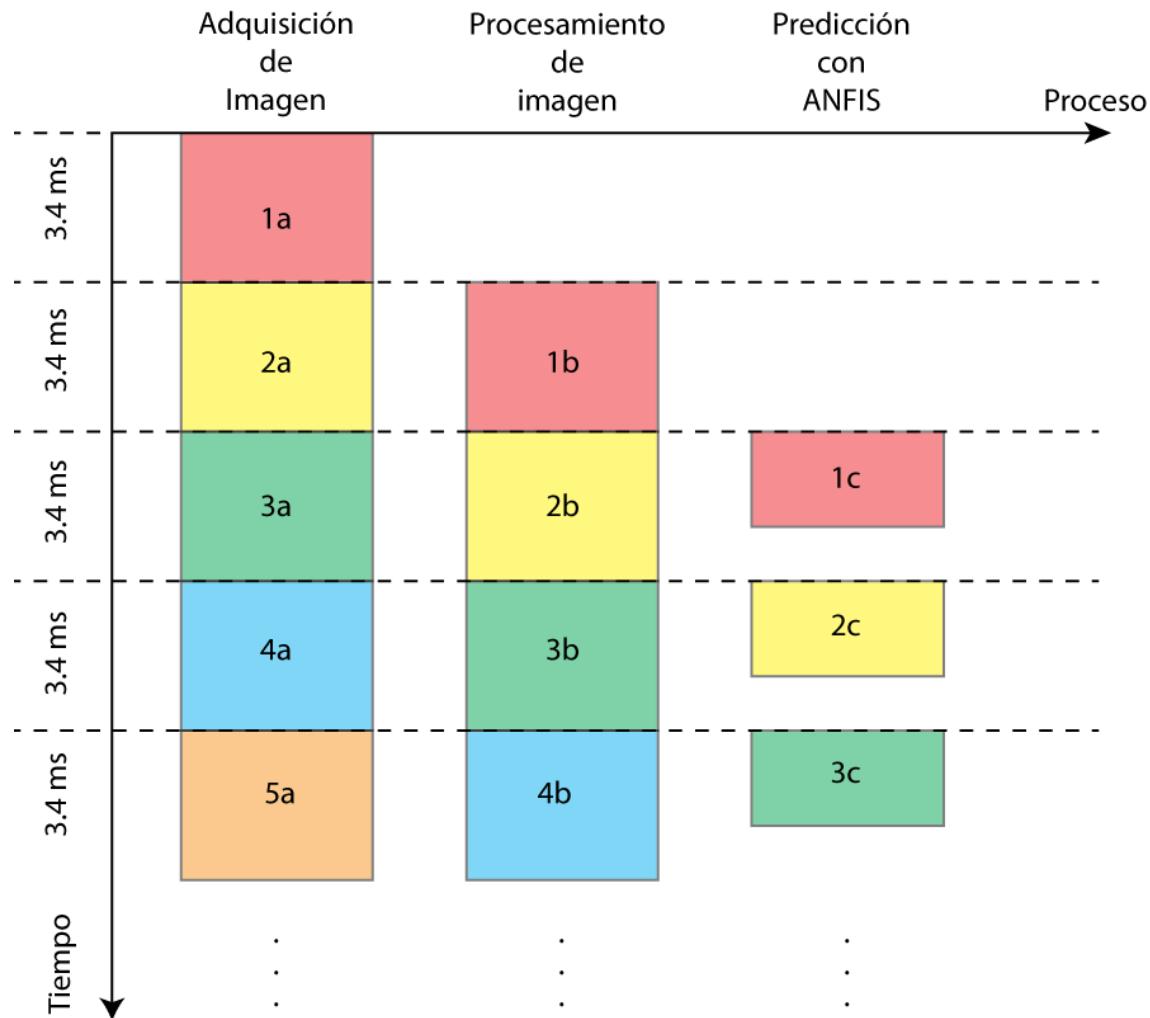


Fig. 7.1: Tiempos de ejecución de los procesos

Finalmente, se desarrolla una aplicación con interfaz de usuario con procesamiento “Offline” para evaluación de vídeos guardados y “Online”, de forma que cumpla con los objetivos de estudio mencionados (figuras 7.2, 7.3). La primera, cuenta con la opción de seleccionar la región de interés de la imagen, ya que los bordes de esta pueden contener información no deseada, por lo cual se permite la eliminación de los margenes por medio de la selección

manual. Además, se permite seleccionar el color del objeto a reconocer. En cuanto a la opción de procesamiento “Online”, la aplicación buscará por los dispositivos de vídeo Basler conectados al momento de ejecutarse, mientras que dará la opción de buscarlos si es que son conectados luego.

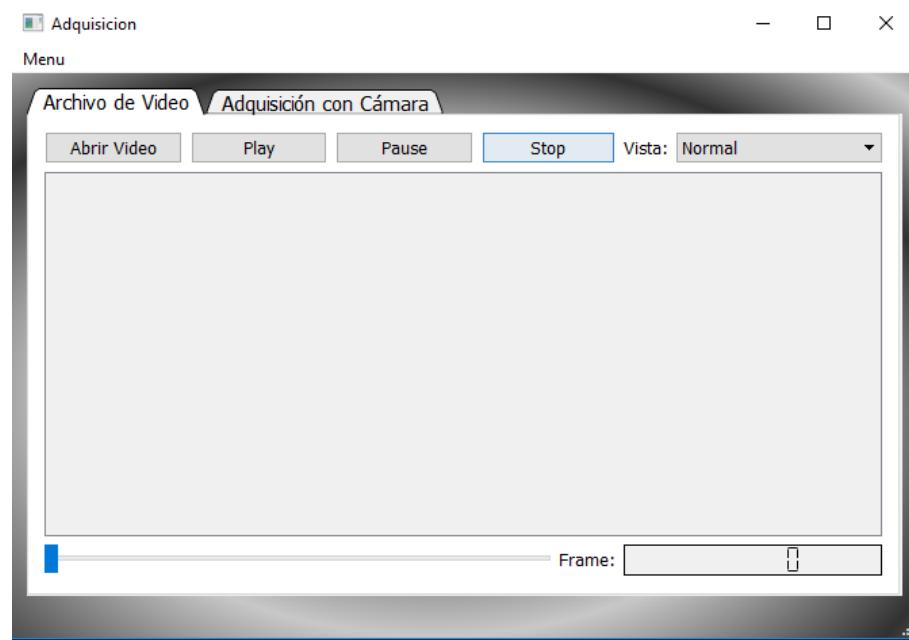


Fig. 7.2: Programa con ejecución Off-line

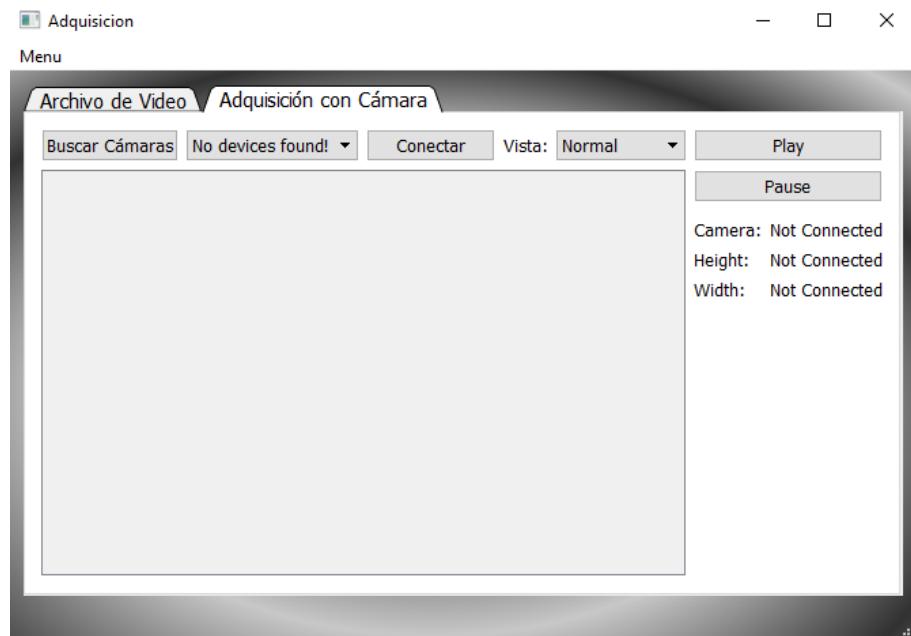
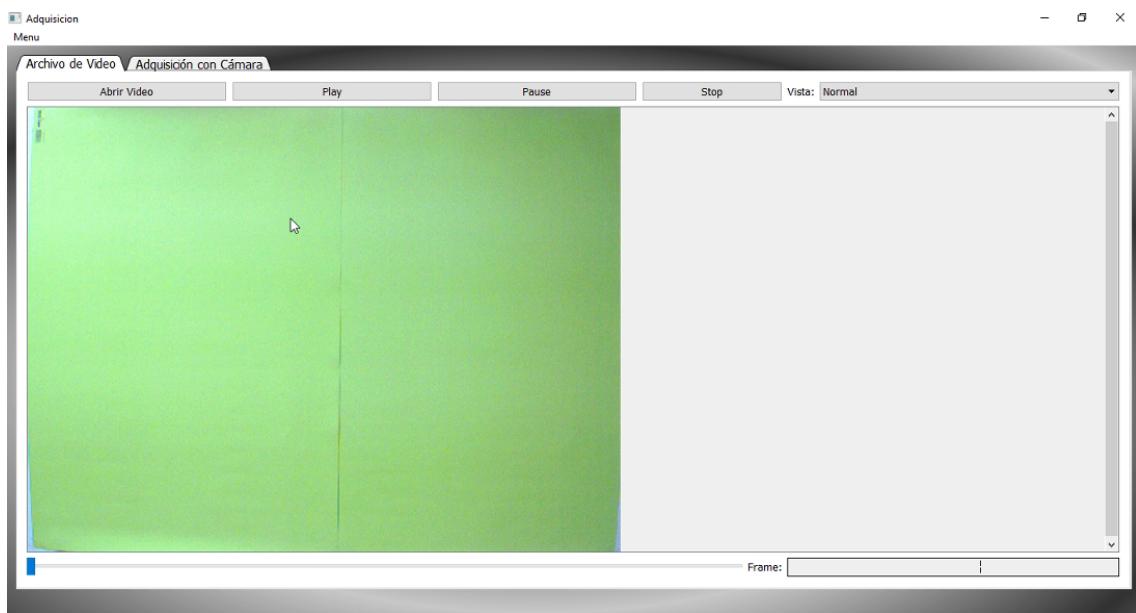
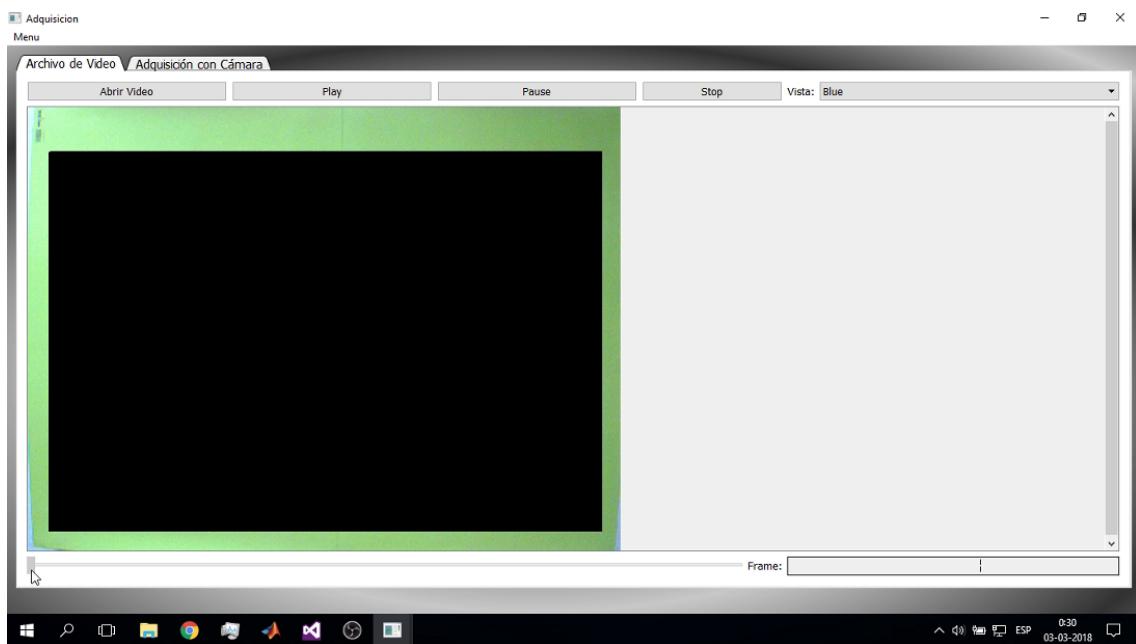


Fig. 7.3: Programa con ejecución On-line

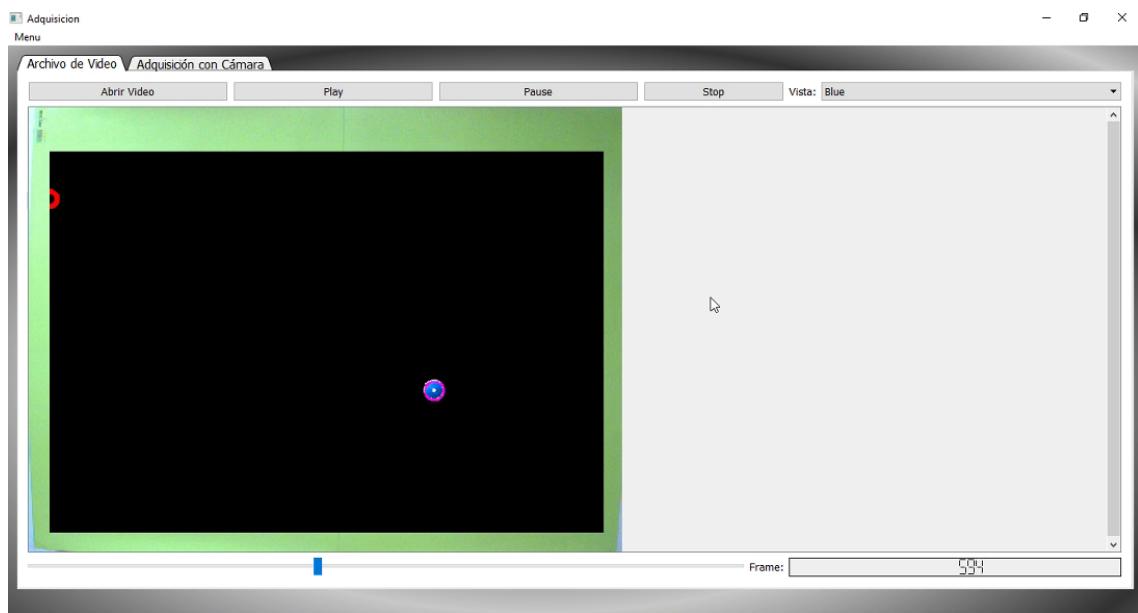
En las figuras 7.4a, 7.4b, 7.3d, 7.2e, 7.2f y 7.1g se muestra una secuencia correspondiente a la ejecución la aplicación con interfaz gráfica utilizando un vídeo grabado, donde se utiliza una región de interés que elimine los margenes que puedan interferir con la detección. Se hace el seguimiento de una pelota de color azul y se utiliza el algoritmo de ANFIS para predecir su posición en el extremo izquierdo de la imagen.



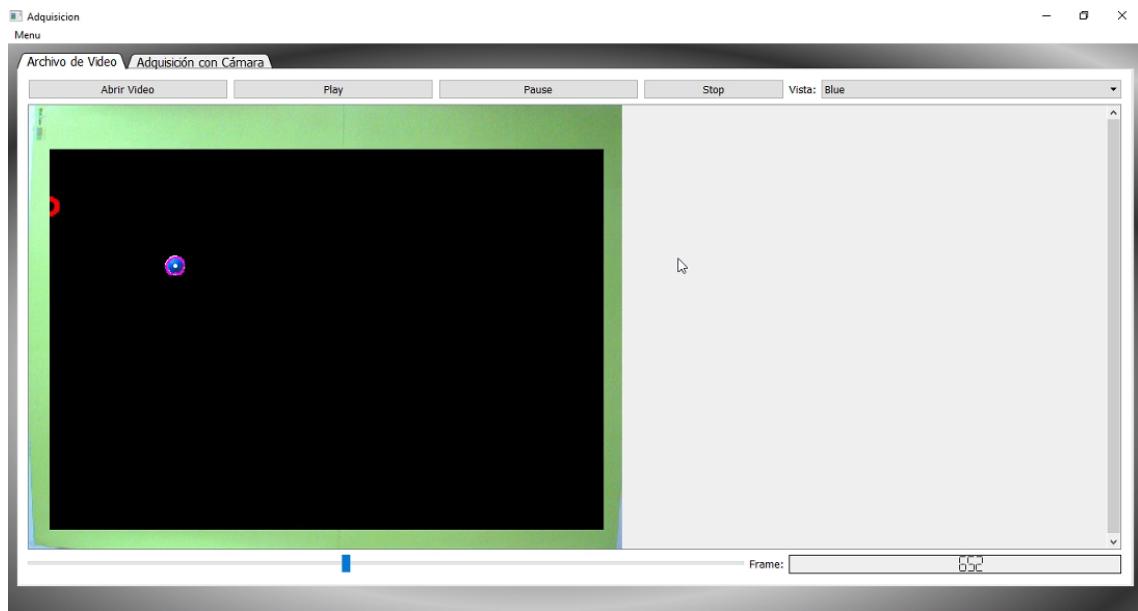
(a) Estado incial



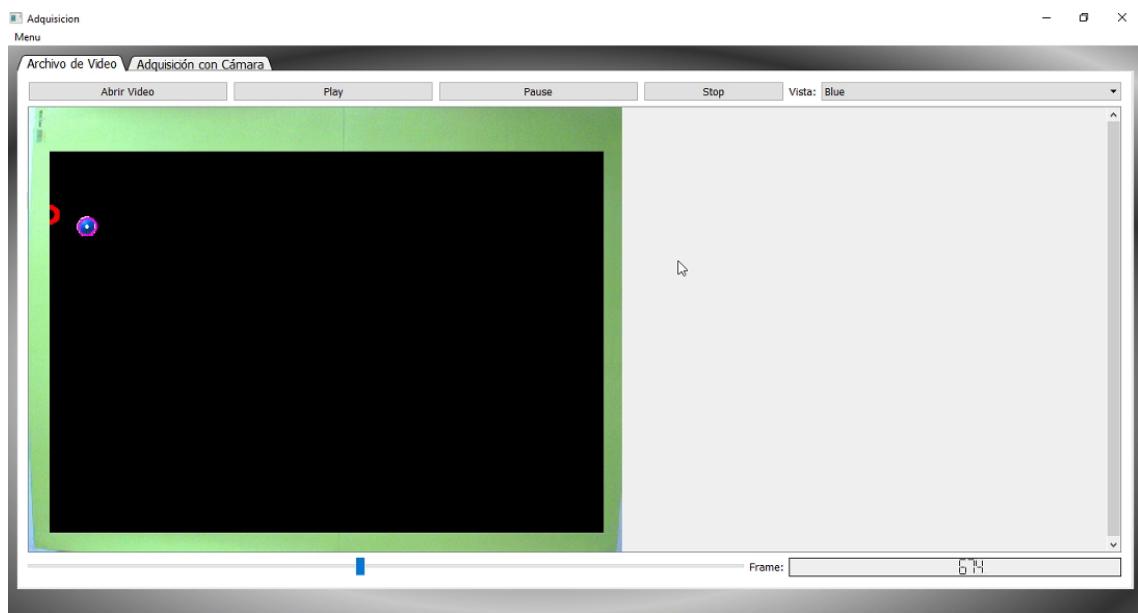
(b) Selección de la región de interes



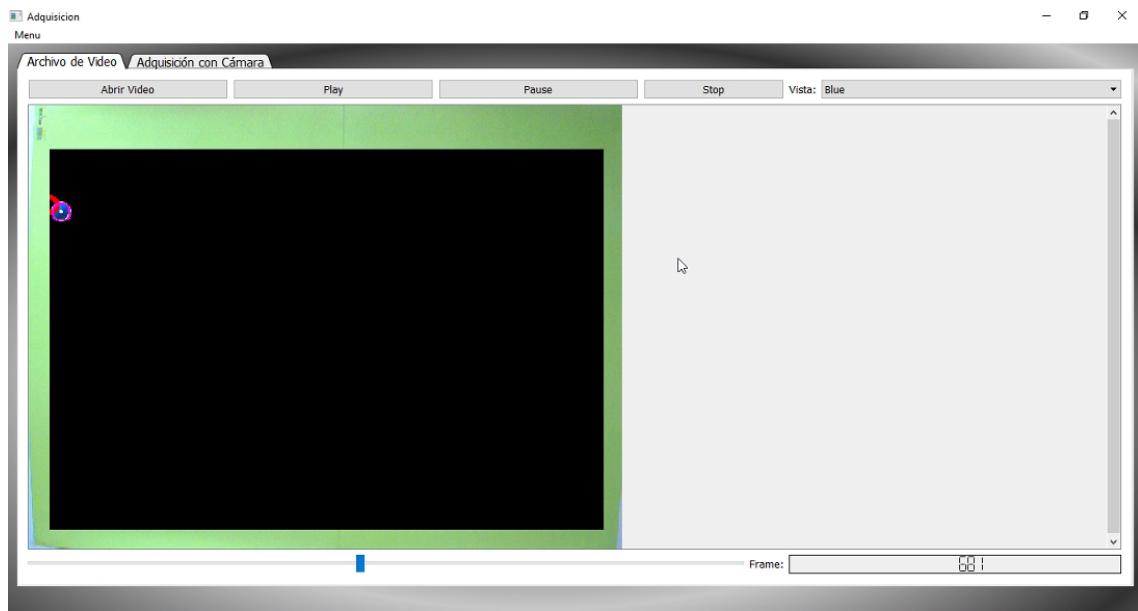
(c) Posición 1



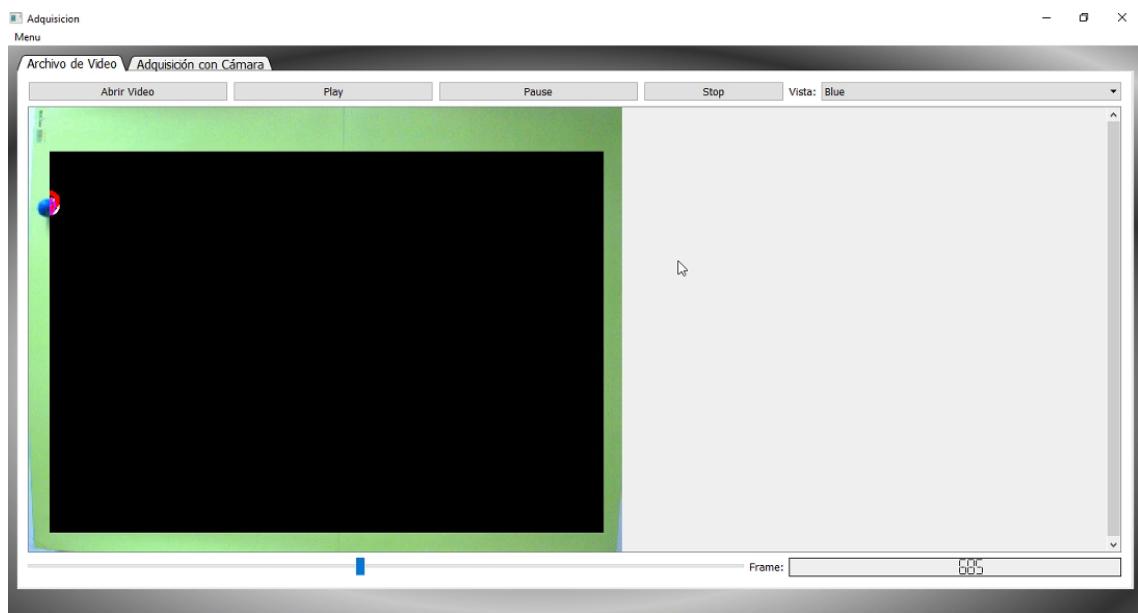
(d) Posición 2



(e) Posición 3



(f) Posición 4



(g) Posición 5

Fig. 7.1: Ejecución del programa. Detección de pelota azul y respuesta de ANFIS en extremo izquierdo (en rojo)

## Bibliografía

- [1] Federación Internacional de Fútbol de Mesa. <https://www.tablesoccer.org/>
- [2] Reglamentación Internacional de fútbol de mesa. 2018
- [3] T. Chau and J. Then and M. Turnbull and S. Wan and S. Cheng, Robotic Foosball Table, University of Adelaide, 2007.
- [4] Gauselmann GmbH, StarKick, 2005.
- [5] C. Salzmann, M. Savary, C. Picard. Robotic Foosball Table, Ecole Polytechnique Fédérale, 2013
- [6] R. Janssen, J. Best, R. Van Molengraft y M. Steinbuch. The Design of a Semi-Automated Football Table, 2010.
- [7] R. Janssen, J. Best, R. Van Molengraft y M. Steinbuch. The Design of a Semi-Automated Football Table Chap IV Implementations, 2010.
- [8] Reglamentación Internacional de fútbol de mesa. 2016.  
<https://www.tablesoccer.org/rules/documents/ITSFRulesSpanishNew2016.pdf>
- [9] Máxima Velocidad alcanzada en competiciones FEFM. 2018. <http://www.fefm.es/p/el-futbolin>
- [10] Donald A. McQuarrie and John D. Simon. "Physical Chemistry: A Molecular Approach. C.1.3: Photoelectric Effect Explained with Quantum Hypothesis". 2018.
- [11] Efecto Dolly, por Calvin Hoag. <https://www.youtube.com/watch?v=VNO3BtNT9bY>.  
2012

- [12] Basler. Especificaciones Cámara Basler. <https://www.baslerweb.com/en/products/cameras/area-scan-cameras/ace/aca800-510uc/>
- [13] OpenCV. Features Matching. <https://docs.opencv.org/>
- [14] Bay, H. and Tuytelaars, T. and Van Gool, L. “SURF: Speeded Up Robust Features”, 9th European Conference on Computer Vision, 2006.
- [15] J. Canny. A Computational Approach to Edge Detection, IEEE Trans. on Pattern Analysis and Machine Intelligence, 8(6), pp. 679-698. 1986.
- [16] Christiano Gava, Gabriele Bleser 2D projective transformations (homographies).
- [17] Fuzzy Logic Desinger. Matlab & Simulink. <https://www.mathworks.com/>
- [18] Artificial Neuronal Network. Matlab & Simulink. <https://www.mathworks.com/>
- [19] ANFIS. Matlab & Simulink. <https://www.mathworks.com/>
- [20] Seung-Ki Sul. Control of Electric Machine Drive System. Electric Machine Drive System. 1, pp. 4-6.
- [21] Graham C. Goodwin, Stefan F. Graebe, Mario E. Salgado. Control System Design. Classical PID Control.
- [22] Maxon Motor. DCX 35 L. <https://www.maxonmotor.es>
- [23] Seung-Ki Sul. Control of Electric Machine Drive System. Electric Machine Drive System. 1, pp. 9-10.
- [24] sFunction level 2. Matlab & Simulink. <https://www.mathworks.com/>