



UNIVERSIDAD NACIONAL AUTÓNOMA DE
MÉXICO

FACULTAD DE ESTUDIOS SUPERIORES
ARAGÓN

“SISTEMA DE DETECCIÓN Y
SEGUIMIENTO PARA DERRIBO AÉREO DE
DRONES”

T E S I S

Que para obtener el título de:

INGENIERO ELÉCTRICO ELECTRÓNICO

(ÁREA CONTROL)

P R E S E N T A:

CLAUDIO HERNÁNDEZ POSADAS

Asesor de tesis:

M. en C. Arturo Ocampo Álvarez



MÉXICO 2015

Ciudad Nezahualcóyotl, Estado de México



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Agradecimientos

En primer lugar, quisiera hacer constar mi agradecimiento a la UNAM, los valores de la universidad quedaron muy impregnados en mí, de ella he obtenido una formación muy integral y me ha convertido en la persona que soy, siempre estaré en deuda con la Universidad.

También quisiera dar gracias a mi asesor de tesis, M. en C. Arturo Ocampo Álvarez, por sus acertados consejos y comentarios, por su confianza en un proyecto que en primera era difícil de concebir viable, por compartir su experiencia científica y por su tenaz orientación hacia el estudio de posgrado e investigación.

De igual manera quisiera expresar mi agradecimiento al M. en C. Ernesto Alonso Villegas Jiménez, por despertar esa inquietud de realizar proyectos diferentes a los comunes, por persuadirme y orientarme a la investigación, de ver más allá de lo que impone el plan de estudios, ha sido piedra angular en mi formación como ingeniero.

Del mismo modo, he de expresar mi gratitud al Ing. José Ramírez Mora por su asesoría y consejos en los proyectos que realizamos, por hacer del servicio social una actividad grata y por permitirnos trabajar en el laboratorio.

Así mismo expreso mi agradecimiento al Ing. Procoro Pablo Luna Escorza, por sus valiosas observaciones y correcciones que hizo a la tesis

Quisiera también expresar mi agradecimiento de forma muy especial a Emmanuel y Omar, por ser excelentes compañeros tanto dentro como fuera de la escuela, por las sugerencias en la realización de esta tesis y aunque hemos tomado diferentes caminos algún día volveremos a las viejas andadas.

Y por último y más significativo, quiero expresar mi agradecimiento a mi familia y en especial a mi Madre por darme su apoyo incondicional; comprensión, paciencia y confianza en mí, sin ello este trabajo nunca hubiera sido posible.

“La imaginación es más importante que el conocimiento.”

Albert Einstein (1879-1955)

*“Es fácil saber cuándo uno va por buen camino,
éste siempre es de subida”*

Ralph W. Emerson (1803 - 1882)

PRÓLOGO

Los vehículos aéreos no tripulados (VANT), en inglés, Unmanned Aerial Vehicle (UVA), mejor conocidos como “Drones”, han sido principalmente utilizados por militares y organizaciones gubernamentales, sin embargo el crecimiento en la práctica civil ha aumentado recientemente de manera exponencial, sobre todo por el desarrollo tecnológico, lo que ha originado el desarrollo de aplicaciones por hobbistas, investigadores, académicos e industriales [1].

Sin embargo, a consecuencia del aumento en el uso civil de los Drones [2], en algunos lugares (ej. EUA), ha sido necesario crear marcos regulatorios para hacer frente a su operación ilegal o indebida [2], algunas de estas operaciones son: invasión a la privacidad, entrega de paquetería indeseada o peligrosa, espionaje, etc.

En México y en otros países, también ha crecido el uso civil de los Drones aunque a menor escala, y en consecuencia, el reglamento no ha sido debidamente actualizado como en otros lugares.

Lo que estimula la posibilidad de realizar operaciones ilegales o indebidas con ellos, y en algunas de estas circunstancias es preferible corregir la irrupción de estos aparatos de manera inmediata [3].

En la actualidad existen sistemas similares para el derribo de Drones tales como “Sky-Guard” [4] los cuales son de costo muy elevado, de uso exclusivo del ejército y en consecuencia de difícil adquisición, estos proyectos militares surgieron con el fin de derribar UVA's espías, pero con el actual desarrollo tecnológico los Drones se han conseguido fabricar de manera casera ocasionando una alarmante preocupación, es por ello que surge la necesidad de crear normativas para su uso y sistemas que los inhabiliten cuando infrinjan en un delito.

Para lograr tal fin se desarrolla un sistema utilizando la técnica denominada “Visual Servoing”, en la que se fusionan diferentes áreas, como lo son: Procesamiento de imágenes, Cinemática, Dinámica, Teoría de Control y Cómputo en Tiempo Real [5].

Se tiene la necesidad de utilizar técnicas de visión artificial debido a que por sus características, los Drones no emiten calor y son de plástico, lo que dificulta su detección y seguimiento con los sistemas comúnmente empleados para tal fin.

La visión Artificial como sistema de instrumentación debe afrontar uno de sus problemas fundamentales: la estimación de un objeto en el espacio a partir de la percepción del entorno, para hacerle frente se ajustarán funciones ya definidas en Matlab®, únicamente para el sistema de seguimiento.

Una vez conseguido un óptimo seguimiento se lograra inhabilitar el objetivo eficientemente mediante la concentración de energía, haciendo uso de la tecnología láser la cual producirá ablación térmica en su material.

ÍNDICE

PRÓLOGO.....	4
OBJETIVO	10
OBJETIVOS PARTICULARES	10
JUSTIFICACIÓN.....	10
HIPÓTESIS	10
INTRODUCCIÓN.....	11
ANTECEDENTES Y GENERALIDADES.....	12
Drones	12
Visión Artificial.....	13
Sistema de Visión humano	13
Problemas de la Visión Artificial	15
Representación de imágenes digitales	17
Control Visual	18
Visual Servoing	19

CAPÍTULO 1 20

DISEÑO DEL HARDWARE PARA EL SISTEMA DE SEGUIMIENTO 20

1.0 INTRODUCCIÓN 21

1.1 CÁMARA 24

 1.1.1. Consideraciones para elegir una cámara 25

1.2 LÁSER 27

 1.2.1. EL haz láser 27

 1.2.1.1 Propiedades del haz láser 27

 1.2.1.2 Tipos de Láser 28

 1.2.2. Diodo Láser 28

 1.2.3. Triangulación laser 29

 1.2.3.1. Medición por Ultrasonido 29

 1.2.3.2. Medición por triangulación Láser 30

 1.2.3.3 Algoritmo para la triangulación láser 32

1.3 MICROCONTROLADOR 34

 1.3.1. Modulo USB del PIC18F4550 35

 1.3.2. Módulo TIMER 36

1.4 SERVOMOTORES 36

CAPÍTULO 2 38

ALGORITMOS DE VISIÓN ARTIFICIAL PARA DETECCIÓN Y SEGUIMIENTO 38

2.0 introducción 39

 2.0.1 Métodos de seguimiento de objetos 40

2.1 DETECTOR CASCADA 41

 2.1.1. Clasificación de Imágenes con patrones 41

 2.1.2. Extracción de características 42

 2.1.2.1. Características HOG 42

 2.1.2.2. Características Haar 43

 2.1.3. Integral de la Imagen o tablas de sumatoria (SAT) 44

 2.1.4. Arquitectura Cascada 46

 2.1.5. Algoritmo del detector cascada 47

 2.1.5.1. Funcionamiento del archivo .XML 47

2.2 DETECTOR DE MOVIMIENTO 48

 2.2.1. Funcionamiento del detector de movimiento 48

 2.2.2. Algoritmo para detección del primer plano 51

2.3 CAMSHIFT	52
2.3.1. Imagen de probabilidad.....	52
2.3.2. Centro de masas (Centroide).....	53
2.3.3. Convergencia	54
2.3.4. Algoritmo para el seguimiento de histogramas	54
2.4 SEGUIMIENTO DE CARACTERÍSTICAS (KLT Tracker).....	55
2.4.1. Selección de puntos característicos.....	55
2.4.2. Kanade Lucas.	56
2.4.3. Algoritmo para seguimiento de esquinas	57
CAPÍTULO 3	58
CONTROL DE ACTUACIÓN Y PREDICCIÓN DE LA TRAYECTORIA	58
3.1 ESQUEMA MECÁNICO	59
3.2 CONTROL	61
3.2.1. Perspectiva del movimiento	61
3.2.2. Objetivo de control.....	62
3.2.3. Control Adaptativo	63
3.2.3.1 Control Adaptativo por programación de ganancias.....	64
3.3 PREDICCIÓN	65
3.3.1. Horizonte de Predicción	65
3.3.2. Filtro de Kalman.....	66
3.3.2.1. Principio computacional del Filtro de Kalman	68
3.3.2.2. Algoritmo del Filtro de Kalman Discreto.....	69
3.3.3. Filtro de Kalman en Imágenes	71
3.3.3.1. Parámetros y sintonización	73
3.4 ANGULO DE ADELANTO Y UBICACIÓN ESPACIAL	74
3.4.1. Ubicación espacial (Bosquejo).....	76
3.4.2. CALCULO ÁNGULO DE ADELANTO	77
Capítulo 4.....	78
PROPUESTA DEL SISTEMA DE DERRIBO	78
4.0 Introducción	79
4.1 ESTADO DEL ARTE.....	80
4.1.1 Tiro Parabólico.....	80
4.1.2 proyectil autopropulsado	80

4.1.3 Armas de energía dirigida (DEW).....	81
4.1.4 Sistemas Análogos	81
4.2 ABLACIÓN TÉRMICA LÁSER.....	82
4.2.1. Ventajas del láser.....	82
4.2.2. Mecanismo del proceso.....	83
4.2.3. Efectos producidos por la absorción de radiación láser	84
Capítulo 5.....	85
IMPLEMENTACIÓN Y RESULTADOS	85
5.0 Introducción	86
5.1 DISEÑO DE SOFTWARE	88
5.1.1. Escaneo del Entorno	89
5.1.2. Detección.....	91
5.1.2.1. Detección de movimiento	91
5.1.2.2. Entrenamiento de Patrones	93
5.1.2.3. Detección de Patrones	95
5.1.3. Seguimiento.....	96
5.1.3.1. Seguimiento de color.....	97
5.1.3.2. Seguimiento de texturas.....	98
5.1.4. Predicción de movimiento.....	101
5.1.5. Diseño del Software para la triangulación láser	104
5.1.6. Diseño de la tabla de ganancias del control adaptativo	111
5.1.7. Diseño del Software para el control de servomotores	114
5.1.7.1. Creación de frecuencia	116
5.1.7.2. Creación de PWM	118
5.2 DISEÑO DE HARDWARE	120
5.2.1. Tarjeta de interfaz	120
5.2.2. Circuito acoplador para el láser	123
5.2.3. Actuadores.....	124
5.2.4. Cámara.....	125
5.2.5. Láser	126
5.3 Experimento	126
5.4 Resultados	127
5.4.1. Escaneo.....	127
5.4.2. Detección de movimiento	127

5.4.2.1. Tiempo de evolución	127
5.4.2.2. Falsas detecciones	128
5.4.3. Entrenamiento.....	129
5.4.3.1. Elección de características.....	129
5.4.3.2. Tiempo de entrenamiento.....	130
5.4.4. Detección de patrón	130
5.4.4.1. Veracidad en la detección (Tipo Haar).....	130
5.4.4.2. Detección en diferentes zonas de mapeo	130
5.4.4.3. Rotación del objeto	131
5.4.4.4. Veracidad en la detección (Tipo Hog).....	132
5.4.4.5. Detección en diferentes zonas de mapeo	132
5.4.4.6. Rotación del objeto	133
5.4.5. Seguimiento.....	134
5.4.5.1. Seguimiento de esquinas.....	134
5.4.5.2. Cambios bruscos de iluminación	135
5.4.5.3. Velocidad de seguimiento	136
5.4.5.4. Seguimiento de color.....	137
5.4.5.5. Cambios de iluminación.....	138
5.4.5.6. Velocidad de seguimiento	138
5.4.6. Predicción	139
5.4.6.1. Estimación por perdida.....	141
5.4.7. Velocidad adaptable	142
5.4.8. Triangulación láser.....	144
CONCLUSIONES	146
Mejoras Futuras.....	147
BIBLIOGRAFÍA	148
ANEXO	150

OBJETIVO

Diseñar y comprobar el funcionamiento del sistema de detección y seguimiento para cualquier objeto móvil en el espacio.

OBJETIVOS PARTICULARES

- Comparar diversos métodos para la detección y seguimiento de objetos en el espacio aéreo y/o terrestre.
- Predecir el comportamiento futuro del objeto y su posición espacial final después de una trayectoria recorrida.
- Coordinar los movimientos mecánicos de manera eficiente para apuntar a las coordenadas espaciales del objeto.
- Estudiar la forma de utilizar un sistema haz láser para inhabilitar el objeto.

JUSTIFICACIÓN

Se pretende en esta tesis dar una solución al problema de obtener la ubicación espacial de objetos en movimiento, con el propósito de resolver de manera económica el control del espacio aéreo-terrestre en zonas privadas o de restringido acceso.

HIPÓTESIS

Problema a resolver

Detectar objetos conocidos (patrones) o desconocidos (para futuro entrenamiento) con cámara en movimiento con el fin de conocer su posición espacial referenciada y realizar una acción de predicción y control utilizando Visual Servoing.

Suposición que se quiere demostrar

Se quiere demostrar que los sistemas que utilizan Visión artificial pueden realizar acciones de control con mucha exactitud reduciendo el número de componentes de los que se componen actualmente sus sistemas equivalentes.

INTRODUCCIÓN

El desarrollo de nuevas tecnologías que permitan reemplazar las actuales u operar en circunstancias nuevas con el propósito de solucionar los problemas, ha sido y será la causa de nuevas investigaciones en las que se conjunen no solo uno sino diversos campos del conocimiento.

En este sentido es que ha surgido en los últimos años el Control Visual-“Visual Servoing”, que por sus características puede aplicarse en muchos campos del conocimiento, tales como: La medicina para realizar con brazos microcirugía; en la industria manufacturera para realizar acomodo y ensamblaje de piezas; en la robótica móvil etc.

Lo que se pretende conseguir con estos sistemas es tratar de imitar las destrezas que poseen los seres humanos con el fin de suplirlos en las labores cotidianas, repetitivas y peligrosas o que no son posibles de realizar por un simple humano.

Como ha ocurrido desde el inicio de la civilización los humanos han intentado delegar las tareas duras o tediosas a otras personas en lugar de hacerlos ellos mismos, esta tendencia ha incrementado al mismo tiempo que la tecnología.

El campo de la inteligencia artificial facilita la interacción hombre-máquina, pero sin duda uno de los problemas de los robots autónomos es la interacción máquina-entorno. La visión artificial es la línea de investigación que trata de resolver el problema de: la percepción visual del entorno.

La detección y reconocimiento de objetos es una actividad habitual de nuestras vidas. Estamos constantemente buscando, detectando y reconociendo objetos, sin embargo aún no es claro la forma en que percibimos los objetos de manera tan precisa y aparentemente sin esfuerzo. Con el objetivo de reconocer un objeto, se debe saber al menos una característica de dicho objeto. El problema principal es ¿Cómo distinguir un patrón en función de la apariencia en el mundo real en la que se encuentra?, es decir, como se puede obtener una representación universal de un objeto que sea capaz de hacer frente a cambios en la variación del objeto y la diversidad escenas en las que se puede encontrar en el mundo. Es una de las problemáticas que se abordaran en el presente trabajo.

El profundo conocimiento de los elementos que interactúan en los sistemas, que los hace funcionar correctamente, es a lo que aspiran todos los diseñadores, pues conociendo esto las mejoras a los sistemas son limitadas solo por la imaginación.

Estudiando este sistema se encontró una singularidad: en este océano de fotones se intuyó que los fotones que se utilizan para la detección y seguimiento son los mismos que se utilizarán para el derribo, solo que en diferente dirección y potencia.

ANTECEDENTES Y GENERALIDADES

Drones

La denominación (vehículo aéreo no tripulado), de siglas (VANT), proviene del inglés (Unmanned Aerial Vehicle), de siglas (UAV), pero el término más extendido es “Drone” que literalmente se traduce como zángano, este nombramiento resulta por la característica de “zumar” que tiene algunos de ellos.

En la actualidad existe una amplia variedad de formas, tamaños, configuraciones y características en el diseño de los Drones. Históricamente los Drones eran simplemente aviones pilotados remotamente pero cada vez se está empleando más el control autónomo para el recorrido de los Drones.

Los actuales usos de los Drones son:

1. Entrega de paquetería

Surgen por iniciativas de empresas, para hacer más atractiva, veloz y eficiente la entrega de productos, estos productos van protegidos dentro de un compartimento inferior al Drone, el cual se abre al llegar al punto destino desplegando una cuerda con el producto anclado hasta llegar al suelo. Algunos son tele-operados y otros comienzan a estar integrados con cámaras y sensores que le permitan hacer el mapeo de la zona donde maniobrarán evadiendo obstáculos. Algunos son:

Empresa	Proyecto
Google x	Project Wing
Amazon	Amazon Prime Air
Domino's	DomiCopter
Google	Proyecto Tango
DHL	Paketkopter

2. Espionaje

Los emplean con este fin generalmente el ejército y los centros de Inteligencia

3. Medios de comunicación

La toma de video aéreas, así como las fotografías en lugares de difícil acceso ahora son posibles para los medios de comunicación, con cámaras instaladas en estos dispositivos, lo que les permite tener mayores atractivos visuales para sus contenidos informativos

4. Perturbación de la paz pública

Son hechos fortuitos que han ocurrido en diversos países

5. Invasión de la privacidad

Visión Artificial

La visión artificial se puede definir como un proceso que produce, a partir de imágenes del mundo exterior, una descripción que es útil para el observador y que no tiene información irrelevante [6]. Se dice proceso porque es un tratamiento digital de la imagen, es una descripción porque extrae características que ayudan al sistema a comprender la escena y no contiene información irrelevante ya que es enfocada.

Se divide en dos campos: el Procesamiento Digital de Imágenes y la Visión Computacional, la primera tiene como objetivo la descripción y reconocimiento de una imagen digital para mejorar la calidad de interpretación humana, la segunda pretende dotar a los ordenadores de la capacidad de poder simular la visión humana y que sea en tiempo real.

Desde una perspectiva más general, la visión artificial por computador es la capacidad de la máquina para ver el mundo que lo rodea, más precisamente para deducir la estructura y las propiedades del mundo tridimensional a partir de una o más imágenes bidimensionales [7].

Sistema de Visión humano

Se considera el sistema sensorial que más depende el ser humano, con el podemos percibir forma, volumen, color, luminosidad, tamaño y demás cualidades que rodean los objetos que en conjunto nos ofrecen la posibilidad de ubicarnos espacialmente y realizar actividades que de otra forma serían imposibles llevarlas a cabo.

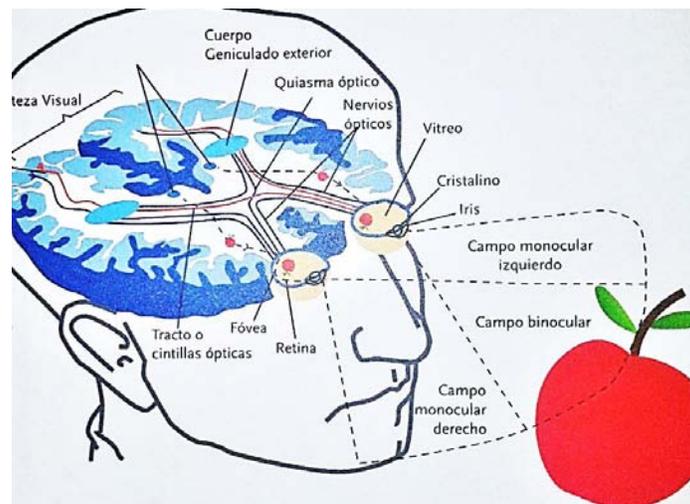


Figura 0.1: Sistema de visión humano.

En este sentido los sistemas de Visión artificial están inspirados en la manera en que la biología procesa la información obtenida de fotones que inciden en la retina, en esta analogía ambos tienen un sistema de adquisición para los humanos se trata de los ojos y en la visión artificial son las cámaras, después de la adquisición se procesa la información que para los humanos se trata del cerebro y para los sistemas de visión artificial son los procesadores y algoritmos, al final se pretende alcanzar el mismo resultado.



Figura 0.2. Analogía entre los sistemas de visión biológico y computacional.

Los humanos y todos los seres vivos que tiene sistemas similares al igual que los sensores de las cámaras detectan solo fotones, es decir, luz irradiada por alguna fuente de luz blanca (como el sol), lo que podemos “ver” son solo colores que se reflejan de los objetos (la luz que no absorben los pigmentos) es lo que detecta nuestro sensor (ojo), lo que observamos es la interpretación de esos colores, pues los relacionamos con figuras conocidas y así es como describimos entornos y entendemos escenas.

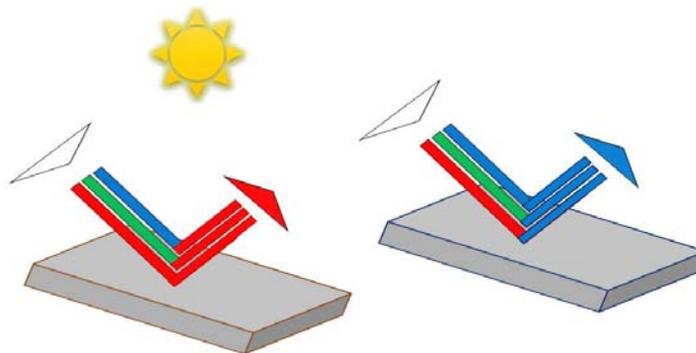


Figura 0.3. Reflejo de fotones en texturas con diferentes pigmentos.

Problemas de la Visión Artificial

La implementación de los sistemas de visión presentan algunas dificultades, en los ambientes controlados deben ser solucionados con los mismos algoritmos, cuando se trata de visión activa, es decir, de entornos cambiantes estas dificultades se agravan aún más y requieren soluciones más creativas, estos problemas son:

- Cambios de iluminación. Es el problema más prominente debido a que los sensores de la cámara detectan solo fotones que inciden en ellos, los cuales son dependientes del ángulo de incidencia que se tenga en los objetos, de esta forma se podrá visualizar diferentes perspectivas de un mismo objeto, que se puede interpretar como objetos diferentes en cada una, es por eso que se debe elegir las más conveniente para una aplicación en particular.

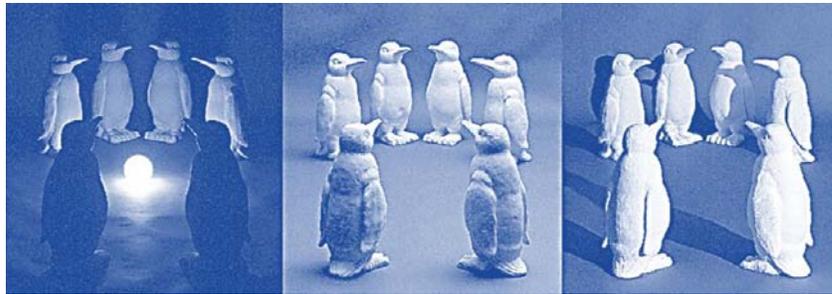


Figura 0.4: Diferentes perspectivas de iluminación.

- Cambios de escala. El escalamiento de los objetos en movimiento es posible solucionarlo desde software, pero tiene sus límites, el objeto no puede escalar en ascenso demasiado porque abarcará todo el campo de visión de la cámara y no podrá distinguir objeto alguno, tampoco debe escalar en descenso demasiado porque dependiendo de la resolución de la cámara puede perderlo en la escena.

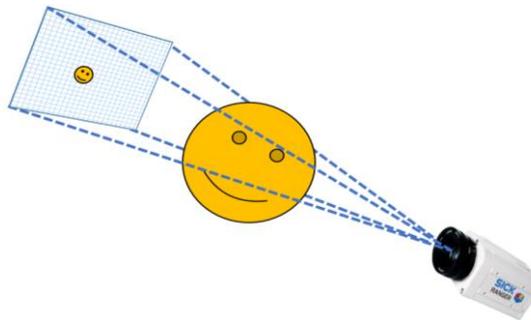


Figura 0.5: Efecto en la traslación mínima y máxima del objeto.

- Deformación (rotación). La rotación de objetos en movimiento puede solucionarse por software, es un problema particular del reconocimiento y seguimiento de patrones, ya que

puede cambiar de forma, textura y color sin realizar un movimiento traslacional aparente mostrando otro lado del objeto.

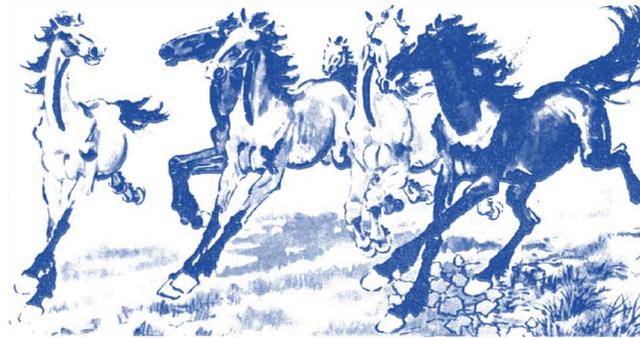


Figura 0.6: Diferentes rotaciones de un mismo objeto.

- Oclusión. Cuando ocurre en la detección, si es una oclusión parcial se puede solucionar por software, cuando se trata de seguimiento es posible predecir su movimiento cuando es una oclusión parcial o total.

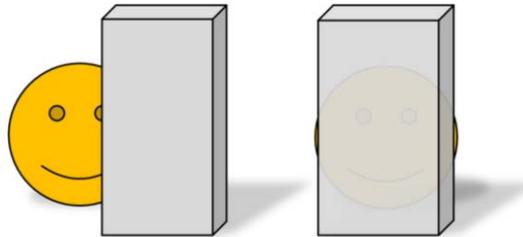


Figura 0.7: Oclusión parcial y total de un objeto.

- Movimiento. El rápido movimiento de los objetos en una secuencia de frames depende de la velocidad de toma y del procesamiento de los frames, si el objeto se mueve más rápido que la toma de frames se deforma en la imagen y no será posible describirlo, se soluciona por hardware.



Figura 0.8: Alta velocidad de movimiento en un objeto.

-
- Pérdida de información. Este problema ocurre cuando se utiliza solo una cámara, debido a que solo detecta una perspectiva del entorno, en este proceso convierte una escena de tres dimensiones en una imagen de dos dimensiones con lo que se pierde información. La solución se hace con hardware, con sistemas de visión estereó (con dos cámaras) y visualizar el entorno en tres dimensiones.

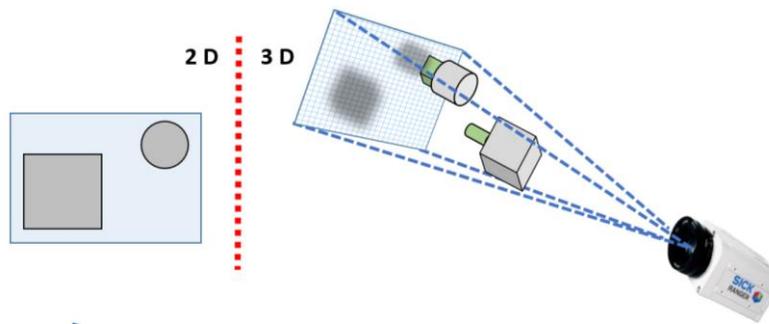


Figura 0.9: Pérdida de información en la transformación de 3D a 2D.

Representación de imágenes digitales

El término imagen se refiere a una función de intensidad bidimensional, representada como $I(x, y)$, donde (x, y) son las coordenadas espaciales y el valor I en cualquier punto es proporcional a la intensidad de la imagen en ese punto.

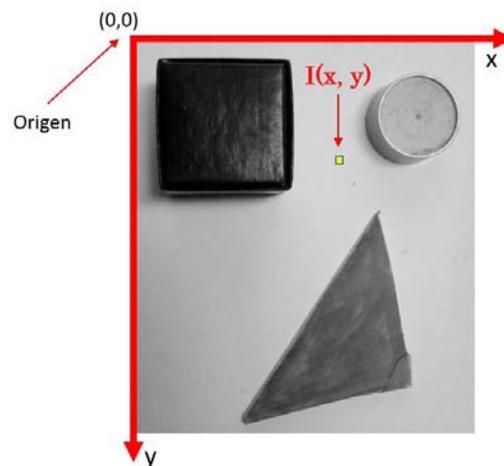


Figura 0.10: Imagen digital.

Control Visual

La visión por computador es una fuente rica en información, que no sólo puede ser usada para la detección y seguimiento de objetos, sino que adicionalmente puede generar información que puede ser empleada para realizar el posicionamiento de un robot relativo a un objetivo, lo que suele denominarse como Control Visual, (Hutchinson. 1996).

Al inicio el control visual se encontraba desacoplado, es decir, por un lado se tenía la cámara (extracción de características visuales) y por otro el sistema de actuación (ley de control), a esto se le denominaba retroalimentación visual -“Visual Fedback” y se podía configurar como: “Ver y mover” estático y dinámico.

La configuración estática se denomina así ya que la cámara se encontraba estática en entornos controlados, el control era sencillo pues se encontraba en lazo semi-abierto, era de esta manera ya que en aquellos tiempos el procesamiento era lento y no convenía depender solo de la imagen debido a que no se habían desarrollado algoritmos suficientemente efectivos. En su ejecución primeramente se extraían y procesaban las características de la imagen que en general era una tarea lenta, después se ejecutaba la acción de control que era relativamente más rápida, este proceso continuaba asumiendo que el entorno permanecía estático.

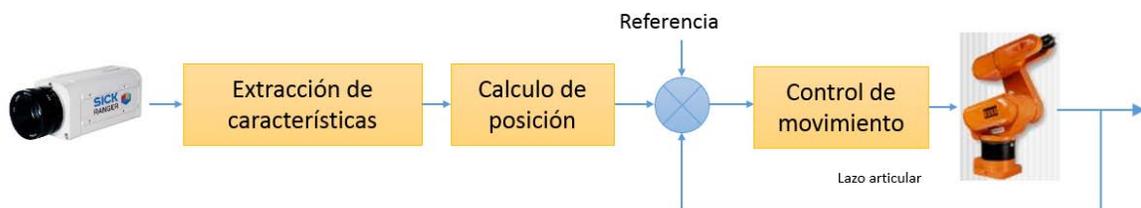


Figura 0.11: Esquema de control “Ver y mover” Estático.

Ver y mover dinámico, también llamado “control visual indirecto dinámico”, en esta mejora los comandos aplicados a los actuadores son actualizados en función al sensado visual durante el movimiento de los mismos, aquí ya se implementan sistemas de control en lazo cerrado, sin embargo aún se encontraban los lazos desacoplados y trabajaban en paralelo, en consecuencia el control de movimiento debía ser más rápido que la extracción de características.

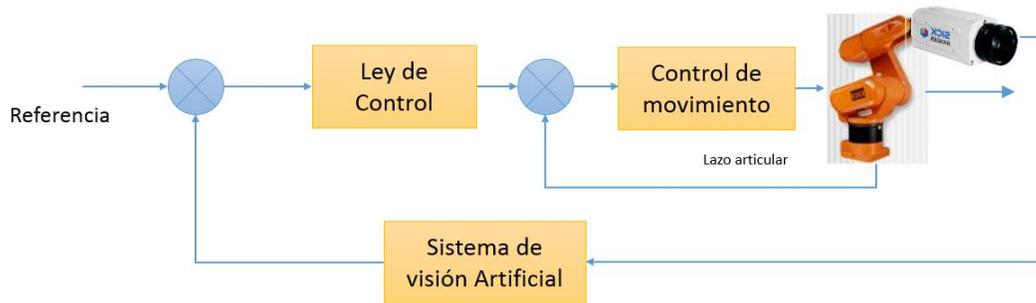


Figura 0.12: Esquema de control “Ver y mover” Dinámico.

Visual Servoing

Al acoplamiento de estos lazos se le denomina control servo visual o “control visual directo”, en este único lazo los comandos de actuación se obtienen directamente a partir del procesamiento de los errores de las características de la imagen, este esquema de control los ciclos deben ser muy rápidos.

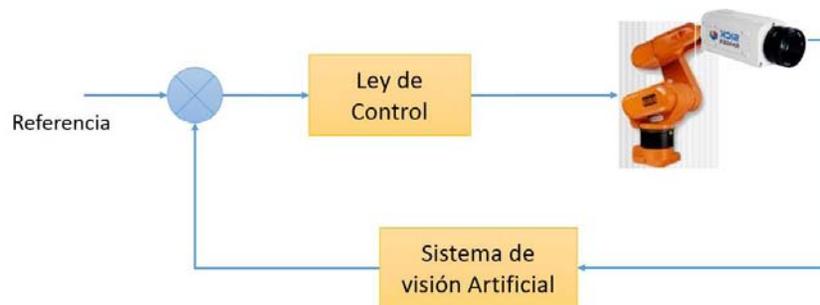


Figura 0.13: Esquema de Control Directo.

Visual Servoing es el resultado de fusionar varias áreas entre las cuales podemos mencionar:

- Visión Computacional
- Cinemática-Dinámica
- Teoría de control
- Cómputo en tiempo real.

Gracias a sus características es posible emplearlo en situaciones de entornos cambiantes, utilizando cámara dinámica y apoyándose de algoritmos de control avanzados.

La tarea de Visual Servoing es el control de robots usando información visual haciéndolos capaces de manipular su ambiente de trabajo en lugar de únicamente observar. (Loreto; 2002), por tal razón es una unión muy interesante y desafiante para cualquiera interesado en alguna de sus áreas.

CAPÍTULO 1

DISEÑO DEL HARDWARE PARA EL SISTEMA DE SEGUIMIENTO

1.0 INTRODUCCIÓN

Existen diversos métodos para realizar el seguimiento de un objeto en movimiento, todos ellos se basan en configuraciones de movimientos independientes que puede realizar las articulaciones de los eslabones de un sistema mecánico, los movimientos van desde la traslación, rotación o unión de ambas, teniendo a su vez diversos grados de libertad, en todo caso describiendo trayectorias basadas en coordenadas cartesianas, cilíndricas o esféricas. Cada uno presenta ventajas y desventajas.

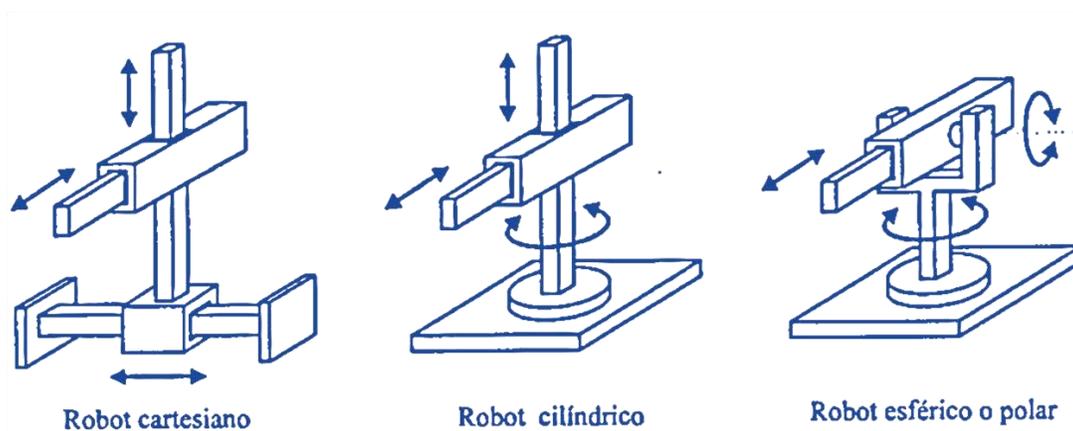


Figura 1.1: Configuraciones con distintos sistemas coordenados.

La configuración cartesiana, es denominada así, a causa de sus articulaciones que se mueven en forma traslacional en tres grados de libertad que genera un espacio cubico y describiendo coordenadas Cartesianas en cada movimiento. Esta configuración se piensa la ideal, pues detectaría el objeto en cualquiera de sus lados y sería difícil que lo perdiera, pero su implementación física es compleja ya que debe coordinar todos sus actuadores para posicionarse siempre frente a una de las caras del objeto.

El cilíndrico, se designa de esta manera porque sus dos grados de libertad (rotacional y traslacional) describen en todo momento un cilindro con sus inherentes coordenadas cilíndricas. Este resuelve el problema anterior ya que no necesariamente se debe posicionar frente al objeto, gracias a rotación de uno de sus ejes puede seguir el objeto cuando pase frente a él, rotando hasta perderlo, su implementación es más sencilla que la anterior, el único inconveniente ocurre cuando el objeto a seguir pasa por encima de él, pues no es capaz de girar hacia arriba o abajo.

La configuración esférica soluciona todos los problemas anteriores ya que con dos grados de libertad rotacionales es capaz de situarse en cualquier coordenada espacial, utilizando las coordenadas esféricas para ello. Su principal problema es que no cuenta con ningún grado en traslación lo que significa que permanecerá en un solo sitio esperando a que pase un objeto frente a él. Otro

inconveniente que presenta es que debido al espacio esférico que proyecta crea distorsiones geométricas en la proyección plana del campo de visión cuando realiza el movimiento rotacional.

Puesto que las características requeridas por el sistema es que pueda seguir objetos moviéndose en el espacio aéreo sin dificultades, la configuración más conveniente es la esférica ya que es más sencilla su implementación y es capaz de posicionarse para seguir objetos que pasen por encima del sistema, el problema de la distorsión geométrica puede llegar a ser despreciable debido a que sistema mantendrá la mayoría de las veces el objeto en el centro donde no ocurre tal distorsión.

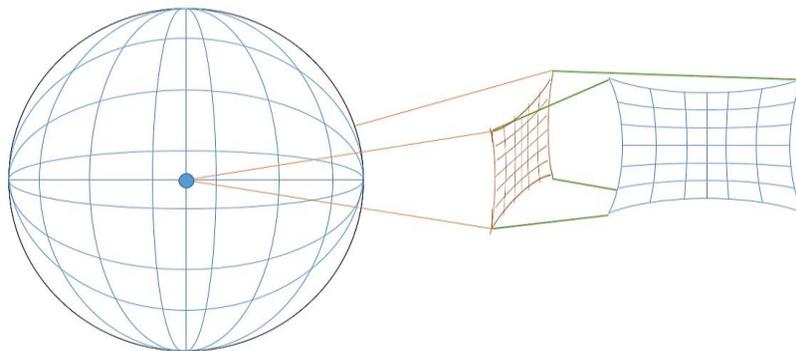


Figura 1.2: Distorsión geométrica en una representación esférica.

Para poder realizar estos movimientos rotacionales la mejor solución es utilizar motores, estos son dispositivos electromecánicos encargados de transformar la energía eléctrica en energía mecánica, en específico crear movimiento rotacional, los motores deben ser capaces de posicionar o mover a velocidades uniformes las articulaciones y de una manera precisa, para tal situación existen tres maneras de llegar a esto con diferentes mecanismos de motores, los cuales son: motores de CD, Motores a Pasos y Servomotores.

Los Motores de CD pueden ser factibles de utilizar sí son auxiliados de una retroalimentación, ya que por principio operan en lazo abierto, la retroalimentación puede ser digital por medio de encoders llegado a tener resoluciones muy altas por revolución, teóricamente son la solución ideal, pero su implementación y acoplamiento mecánico final resulta complejo.

Los Motores a Pasos también cuentan con resoluciones altas por revolución pero no son viables ya que por sus características operan en lazo abierto, y además también es compleja su implementación, pues necesita drivers para crear el control, la interfaz para generar pasos y también el acoplamiento mecánico.

Los Servomotores de modelado tienen un lazo cerrado interno que lo ayuda a corregir su posición, su resolución depende del ancho de pulso (PWM) que recibe y este depende de las características

del Microcontrolador que lo gobierna, llegando a resoluciones aceptables, su implementación y acoplamiento mecánico es relativamente sencillo.

Por lo expuesto anteriormente y para fines académicos se utilizarán dos Servomotores para formar el espacio Esférico con el campo de visión y serán gobernados por interrupciones en tiempos controlados con el Microcontrolador.

El sistema estará conformado por cámara, procesador de imágenes, Microcontrolador y motores organizados como lo muestra la imagen.

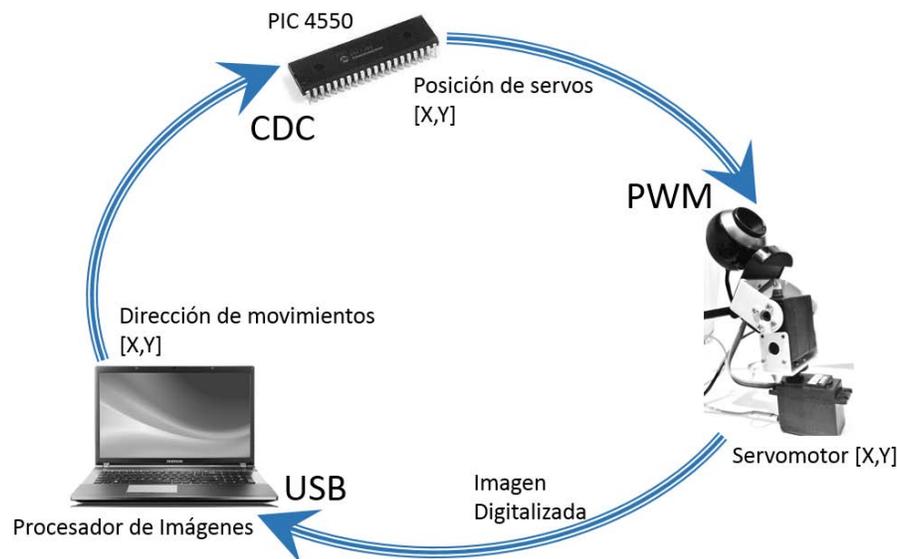


Figura 1.2: Esquema general de seguimiento.

La primera etapa para el sistema de seguimiento, es la etapa de adquisición de imágenes donde constantemente se encuentra capturándolas, el sistema de seguimiento inicia cuando detecta un objeto y da paso a que continúe las siguientes etapas, mientras tanto la imagen que captura es digitalizada y enviada al procesador de imágenes en protocolo USB, aquí es donde se procesa y se obtiene las coordenadas y el sentido de giro de los actuadores, estos datos son enviados al Microcontrolador en protocolo CDC, el cual gobierna la posición de los servomotores con el uso de PWM, que a su vez estos actuadores posicionaran la cámara para obtener una nueva imagen del objeto detectado y se repetirá todo el ciclo hasta perder el objeto. Cada vez que el sistema centre el objeto en la imagen medirá su lejanía por el método de triangulación láser.

A continuación se describirán con más detalle:

1.1 CÁMARA

Este dispositivo genera una imagen del entorno por la composición interna que posee, internamente cuenta con un arreglo de sensores CMOS (Semiconductor de Oxido Metálico Complementario) o CCD (Dispositivo de Carga Acoplada), ambos utilizan dispositivos semiconductores sensibles a la luz los primeros utilizan fotodiodos en serie con un MOSFET y los segundos utilizan capacitores CMOS.

Cualquiera que sea la constitución de la cámara, en ella los sensores se encuentran acomodados en forma matricial, la cantidad de estos sensores determinan la resolución de la imagen.

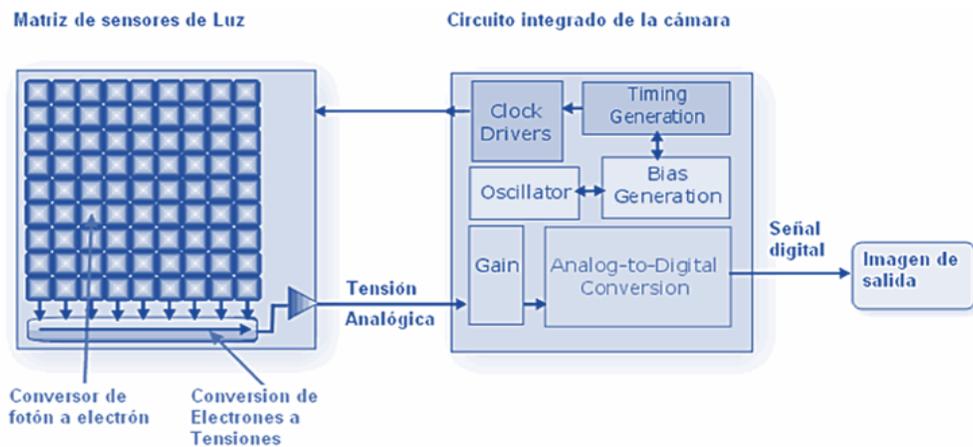


Figura 1.3: Esquema para adquisición de imágenes.

Estos dispositivos pueden absorber la luz que incide en los sensores, lo que da inicio a una etapa de conversión analógica-digital, este proceso convierte la exposición de luz en tensión eléctrica en un tiempo de muestreo determinado, generando una imagen digital conformada de elementos llamados pixeles (picture element) que en conjunto generan un tamaño ($m \times n$), que a su vez este conjunto de pixeles se denominan frame o cuadro, a los que uniéndolos en una secuencia continua en tiempos regulares, genera video en frames o cuadros por segundo.

Los pixeles tendrán un valor asignado, que corresponde al nivel de luminosidad del punto correspondiente en la escena captada.

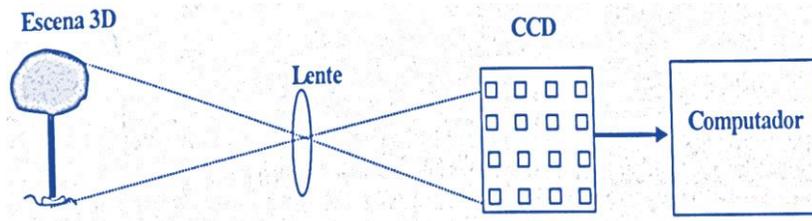


Figura 1.4: Forma de asignar los valores de intensidad a cada pixel.

La intensidad de luminosidad que obtiene en cada pixel es el producto de una componente de iluminación y una de reflectancia, la primera es consecuencia de las fuentes de iluminación existentes en la escena en el momento de captura, la segunda está asociada a los pigmentos del objeto [7].

El sistema de adquisición está restringido a la capacidad del procesador de imágenes, debido a que una imagen en alta resolución contiene una gran cantidad de pixeles lo que dificultaría la velocidad de procesamiento, al igual una alta velocidad de muestreo en la captura de imagen aumenta la carga al procesamiento y disminuye su velocidad.

1.1.1. Consideraciones para elegir una cámara

Campo de visión. Es el área completa que puede percibir la cámara, el área que puede visualizar depende de la distancia entre el objeto y cámara que varía con el lente que lleva puesto.

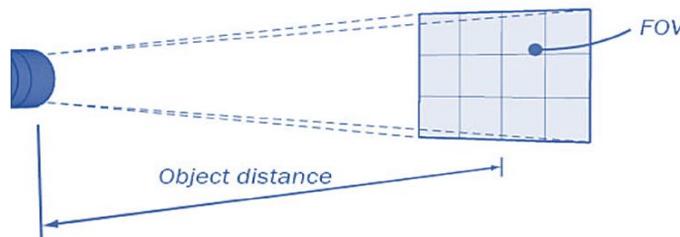


Figura 1.5: Campo de visión.

Distancia Focal. Es la distancia entre el centro de la lente y el punto focal, si el punto focal se encuentra en el sensor, se dice que la imagen es nítida. Si la longitud focal es larga abarca menos campo de visión y la resolución aumenta, por el contrario si la longitud focal es corta abarca más espacio de visión y la resolución baja.

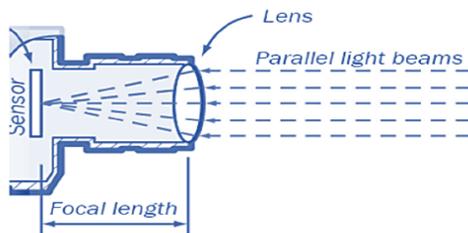


Figura 1.6: Distancia focal.

Distancia mínima. Es la mínima distancia (objeto-cámara) que la cámara puede enfocar un objeto, esto ocurre con cámaras que tienen lentes con una longitud focal larga.

Profundidad de Campo. El plano focal es la distancia a la que conseguimos la mayor nitidez posible y se encuentra dentro de la profundidad de campo.

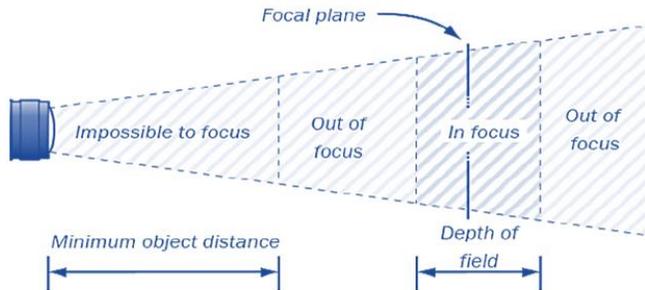


Figura 1.7: Profundidad de campo y distancia mínima.

Lente. El cambio de lente determina el ángulo del campo de visión, las longitudes focales comunes son 8, 12, 16, 25 y 35 mm.

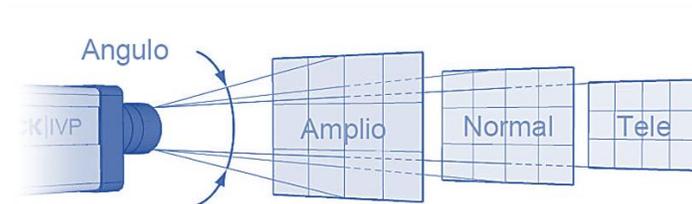


Figura 1.8: Tipos de lentes.

Más aún existen cámaras que pueden operar para detectar otros espectros no visibles para el ojo humano, como la luz ultravioleta, infrarroja, incluso para ver en la oscuridad. Para esta aplicación se utilizó una Webcam por las prestaciones que ofrece y la compatibilidad con el software de procesamiento.

Con la Webcam obtenemos imágenes en el espacio de color RGB y una velocidad de cuadros por segundo aceptable, esta es capaz de auto-enfocarse a la exposición de la luz.

Después de obtener la imagen, es procesada con diferentes algoritmos en un CPU, utilizando la paquetería Matlab® con su toolbox Visión, con lo que se obtendrá las coordenadas del objeto respecto a la cámara. Haciendo uso de una máscara en el centro de la imagen como referencia para indicar el sentido de giro los actuadores con el fin de tratar en todo momento de centrar el objeto.

1.2 LÁSER

1.2.1. EL haz láser

La definición más aceptada es: “Light Amplification by Stimulated Emission of Radiation” (Amplificación de luz estimulada por la emisión de luz). Es un dispositivo que utiliza la emisión estimulada de radiación en un medio apropiado, para generar un haz de luz cuyas características especiales de monocromaticidad, coherencia, y direccionalidad se encuentran perfectamente controladas [8].

1.2.1.1 Propiedades del haz láser

a. Coherencia:

Se dice que una onda electromagnética es coherente cuando su fase no varía aleatoriamente a través del espacio o a lo largo del tiempo. Las fuentes de luz de la naturaleza suelen emitir ondas cuya fase varía de forma desordenada, lo cual indica una emisión espontánea. La coherencia del láser es el resultado de la emisión estimulada, en la que cada nuevo fotón es emitido de forma idéntica a las anteriores y por lo tanto contribuyendo de igual forma a la fase del campo electromagnético.

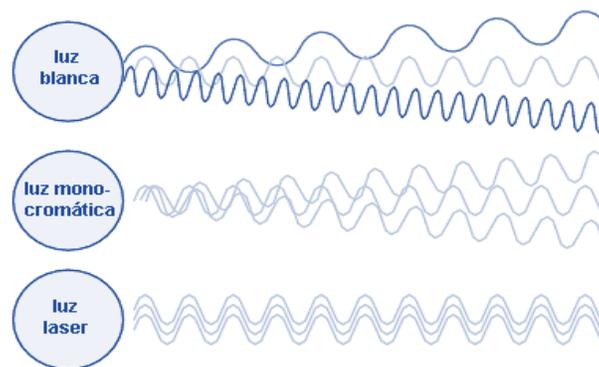


Figura 1.9: Fases para distintas fuentes de luz.

b. Direccionalidad:

Se trata de una consecuencia directa de la cavidad, ya que sólo los fotones que viajan en dirección perpendicular a sus espejos contribuyen al proceso de amplificación. Esto es fundamental para aplicaciones donde se requiere concentrar la energía en una zona determinada.

c. Monocromaticidad:

Ocurre cuando solo se tiene una sola longitud de onda (se compone de un solo color), aunque no es posible llegar a un campo monocromático ideal, los láseres emiten un intervalo de frecuencias lo suficiente estrecho para que pueda considerarse perfectamente monocromática, se trata de una consecuencia de la emisión estimulada, ya que al ser fotones iguales, tienen la misma longitud de onda.

1.2.1.2 Tipos de Láser

Se clasifican por tipo de medio activo que está asociado al mecanismo de excitación. Se puede dividir en 3 grandes familias: de gas, de estado sólido y de semiconductor.

1.2.2. Diodo Láser

Este tipo de láser se empleó en esta aplicación, el diodo está compuesto por una unión pn con dos capas de arseniuro de galio dopado; la unión pn guarda relación con la longitud de onda de la luz que va a ser emitida, existe una superficie altamente reflectante en un extremo de la unión y otra parcialmente reflectante, ambas creando una cavidad resonante para los fotones, esta cavidad es importante pues en ella se reflejan y en consecuencia amplifican solo los fotones alineados perpendicularmente proporcionando al láser una dirección muy definida.

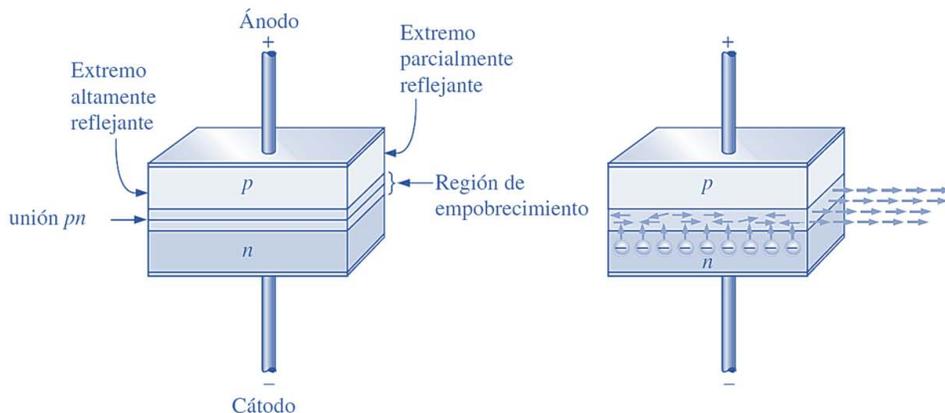


Figura 1.10: Esquema interno del diodo láser.

El dispositivo funciona en el momento en que es polarizado, cuando ocurre esto los electrones se desplazan a través de la unión recombinándose de igual forma que un diodo de propósito general, a medida que los electrones ocupan huecos para recombinarse se liberan fotones. Un fotón liberado puede chocar con un átomo y provocar que otro fotón sea liberado.

A medida que la corriente de polarización en directa se incrementa, más electrones entran a la región de empobrecimiento y hacen que se liberen más fotones. Con el tiempo alguno de los fotones que se mueven al azar dentro de la región de empobrecimiento chocan con las superficies reflejantes perpendicularmente. Este movimiento de vaivén de los fotones se incrementa a medida que la generación de fotones “crece como una bola de nieve” hasta que los fotones que atraviesan el extremo parcialmente reflejado formando un haz láser muy intenso.

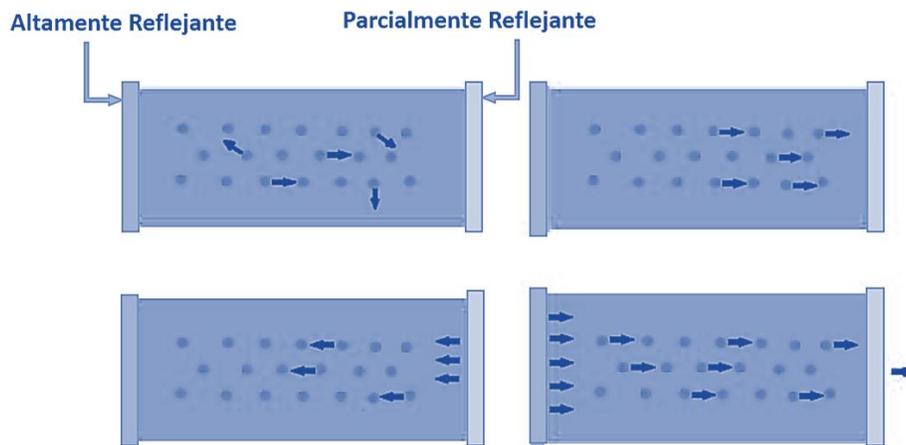


Fig.1.11. Cavidad interna del diodo láser

Cada fotón producido en este proceso es idéntico a los otros fotones en cuanto a nivel de energía, relación de fase y frecuencia. [9]

1.2.3. Triangulación láser

Existen diversos métodos para medir objetos de manera no invasiva, es decir, sin hacer contacto físico con el objeto y el patrón a comparar, el éxito de esto radica en las características del objeto, el entorno que lo rodea y sobre todo las condiciones del medio que existe entre el objeto y el sistema de medición. Los métodos más empleados son:

1.2.3.1. Medición por Ultrasonido

Los sistemas de este tipo son constituidos por componentes electrónicos (emisor-receptor) que interactúan sin problemas con microcontroladores. El ultrasonido es una onda de sonido de alta frecuencia, por arriba del campo audible (20-20 KHz), esta técnica se basa en la emisión de un tren

de pulsos ultrasónicos con una frecuencia en el orden (38-50 KHz), esta onda rebota en el objeto al que se quiere determinar la distancia, la onda es recibida midiendo el tiempo entre la emisión y recepción del rebote de la señal, de esta manera se determina la distancia entre el sensor y el objeto donde se produjo el rebote.

Esta medición se calcula teniendo en cuenta la velocidad de sonido en el aire, que varía con la densidad y temperatura del aire en el ambiente, los cuales deben ser considerados en el cálculo de la distancia.

Otra desventaja que tiene esta medición es que el sonido por su naturaleza es generado en un punto y se expande a través del medio viajando de forma cónica, ocasionando que la medición se realice sobre el objeto más próximo y de mayor tamaño dentro del campo de actuación.

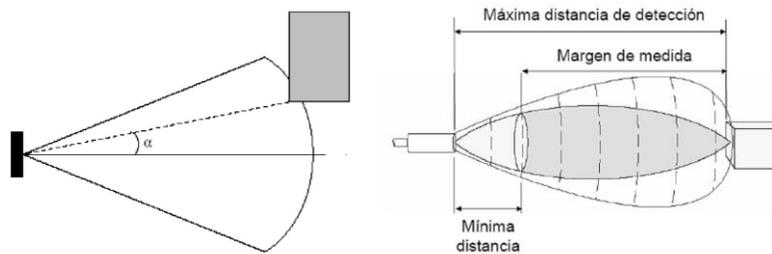


Figura 1.12: Propagación del sonido y efectos en la medición.

1.2.3.2. Medición por triangulación Láser

Este tipo de sistemas solo se encuentra en el mercado para uso industrial, por lo que es necesario fabricarlo para experimentos, el sistema se compone de un láser, un sensor óptico (cámara) y un procesador de imágenes; los cuales están distribuidos para formar un triángulo rectángulo frente al objeto a medir.

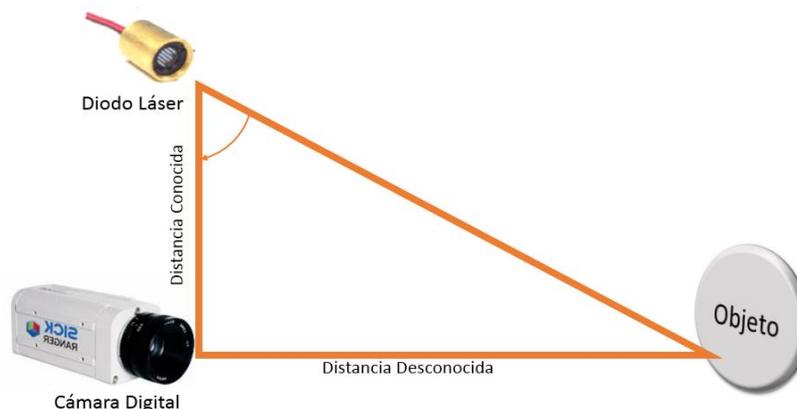


Figura 1.13: Esquema general para realizar triangulación láser.

El método consiste en determinar la distancia a la que se encuentra la superficie de un objeto respecto al sistema de medición, mediante la interpretación de la imagen capturada que tiene un punto que es más luminoso respecto al entorno, este a su vez es producto de la incidencia un haz láser que se refleja en la superficie analizada del objeto, esto se logra capturando desde una cámara, la posición del espectro del láser en la superficie del objeto en análisis. Al acercarse la superficie a la cámara, el espectro reflejado se desplaza a través de los píxeles de la imagen. Esto se produce debido al ángulo de incidencia del láser respecto a la cámara. El desplazamiento en los píxeles, se puede relacionar directamente con la distancia en la que se ubica la superficie respecto al sensor.

Existen diferentes configuraciones para realiza el efecto de triangulación láser, la que se analizará a continuación tiene la ventaja de tener mayor resolución, como se puede ver en la figura abarca todo el campo visual, sin embargo tiende a medir el objeto más cercano donde incide el haz láser y por lo normal detecta objetos debajo del centro focal de la imagen, ya que es la zona donde se proyecta en mayor medida el haz.

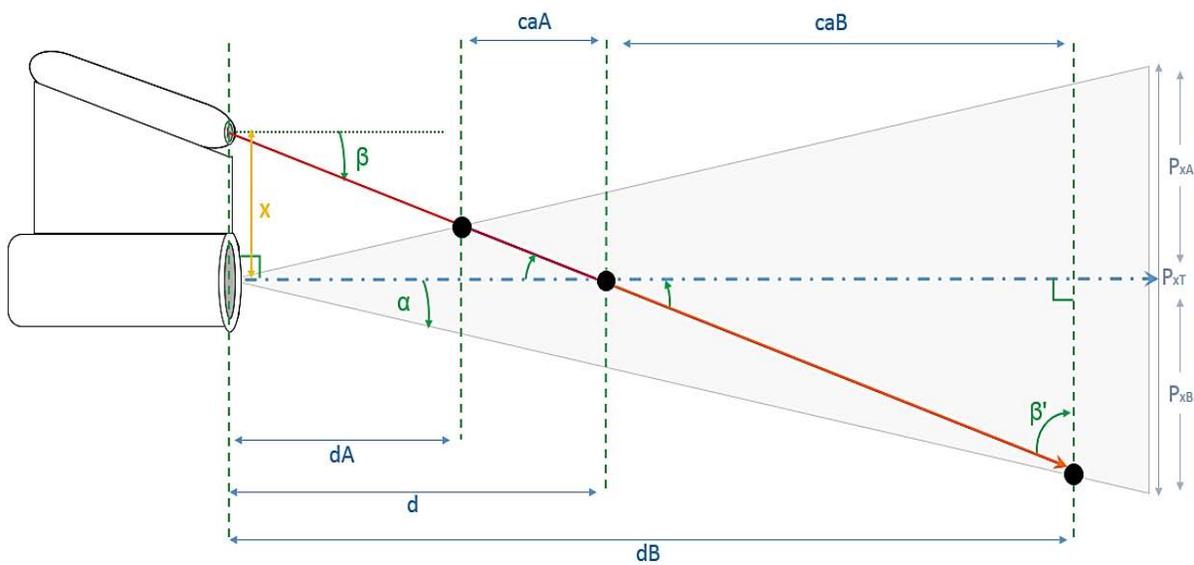


Figura 1.14: Esquema implementado para la triangulación láser.

En donde:

X : Distancia conocida entre el láser y la cámara.

α : Ángulo entre respecto al centro focal de la imagen.

β : Ángulo de inclinación del láser.

β' : Ángulo complementario de inclinación del láser.

P_{xT} : Número total de pixeles en X .

P_{xA} : Numero de pixeles zona A.

P_{xB} : Numero de pixeles zona B.

caA : cateto adyacente del triángulo A.

caB : cateto adyacente del triángulo B.

dA : determina la mínima distancia para que un objeto sea detectado por la cámara, la distancia se calcula como:

$$dA = \frac{x}{\tan(\alpha) + \tan(\beta)} \quad (1.1)$$

dB : determina la máxima distancia para que un objeto sea detectado por una cámara, la distancia se calcula como:

$$dB = \frac{x}{\tan(\beta) - \tan(\alpha)} \quad (1.2)$$

d : determina la distancia donde el láser intercepta el centro focal de la imagen, entre $dA < d < dB$.

$$d = x(\tan(\beta')) \quad (1.3)$$

1.2.3.3 Algoritmo para la triangulación láser

El primer paso es determinar las constantes físicas que regirán al sistema $(x, \beta$ y $Px)$, con ellas se definirá la distancia mínima-máxima así como la resolución en la medición, el ángulo α depende de las características internas de la cámara (se trata del ángulo del campo de visión) y se obtiene experimentalmente. En la imagen se puede observar que el método consiste en calcular los catetos de los triángulos en función de los parámetros físicos iniciales, estos triángulos surgen de la partición del campo de visión en función del centro focal de la cámara dando como resultado dos zonas A y B, con su respectivo triángulo.

Para calcular los catetos adyacentes de los triángulos A y B se debe suponer que ambos surgen del cálculo de las distancias mínima y máxima.

$$caA = d - dA \quad (1.4)$$

$$caB = dB - d \quad (1.5)$$

Estas suposiciones servirán para calcular la hipotenusa de ambos triángulos que cruzan el campo de visión de cada zona.

$$hipA = \frac{caA}{\cos(\beta)} \quad (1.6)$$

$$hipB = \frac{caB}{\cos(\beta)} \quad (1.7)$$

Asimismo se puede observar en la imagen que existe una relación entre el pixel seleccionado y la hipotenusa, expresándose como un factor de resolución.

$$FacA = \frac{hipA}{P_{xA}} \quad (1.8)$$

$$FacB = \frac{hipB}{P_{xB}} \quad (1.9)$$

El cálculo final de la distancia se realiza dependiendo de la zona del pixel en cuestión, el cálculo se realiza considerándolo como un triángulo rectángulo.

$$DA = [(P_y)(FacA) (\cos(\beta))] + dA \quad (1.10)$$

$$DB = [(P_y - P_{xA})(FacB) (\cos(\beta))] + d \quad (1.11)$$

El pixel P_y se encuentra como coordenada (x, y) después del procesamiento de la imagen en donde solo se toma en cuenta el valor de (y) .

El procesamiento de la imagen utiliza el método de Threshold (Umbral) que detecta luminosidad en un área que abarca el centro horizontal de la imagen, que con ayuda de operaciones Morfológicas es posible obtener la coordenada del Centroide que genera el punto luminoso de la incidencia del haz láser.

Para que el sistema funcione correctamente se debe tomar en cuenta que $(\alpha) < (\beta)$ y que el objeto a medir se encuentre centrado en la imagen.

Existe una relación entre los parámetros X y α , pues seleccionando adecuadamente estos parámetros el sistema es capaz de medir distancias del orden de Km o cm siempre y cuando la cámara pueda detectar el reflejo de la incidencia del haz láser, la manera de configurar el sistema para medir Km es acercar α a β sin sobrepasar y eligiendo una cantidad mayor en X , el único inconveniente es que la exactitud disminuye, en cambio para configurarlo en cm, se hace todo lo contrario y por consecuencia se mejora la exactitud.

Esta etapa de medición ocurrirá cada vez que el sistema logre centrar el objeto de interés en el centro de la imagen, esta medición ayudará a obtener la ubicación espacial del objeto y poder realizar un control sobre él.

1.3 MICROCONTROLADOR

Es un circuito de alta escala de integración que está constituido de tres componentes fundamentales: microprocesador, memoria y unidades E/S.

- Unidad Central de Proceso (CPU). El microprocesador es un circuito integrado que contiene la Unidad Central de Proceso, la cual lee los valores de la memoria, los interpreta como órdenes y los ejecuta en el orden establecido.
- Memoria de Programa. En este componente se guarda el programa o instrucciones que gobiernan la aplicación, estos valores son fijos ya que radican en una memoria EEPROM (memoria ROM eléctricamente programable).
- Memoria de datos. Esta memoria es de lectura/escritura y almacena las variables y los datos obtenidos del cálculo, la memoria es de acceso aleatorio (RAM), y por lo tanto volátil.
- Unidades de entrada y salida. El Microcontrolador se compone de pines que funcionan como entrada/salida para comunicarse con el exterior, y se agrupan en puertos.

Los microcontroladores gracias a sus características y prestaciones se han conseguido utilizar como interfaces PC-Actuador, en este sentido es como se utiliza para esta implementación, existen diversos microcontroladores en el mercado pero la gran mayoría para comunicarlo con la PC utilizan el estándar de comunicación RS232 (Comunicación Serial), pero este puerto se encuentra cada vez menos en los equipos, y las PC de nueva generación ya no lo tienen integrado, ahora se dispone solo del puerto USB, el cual ya es un estándar, en base a esto los microcontroladores se han adaptado para solucionar el problema, así es cómo surge la comunicación CDC (Clase de Dispositivo de Comunicación) en los microcontroladores.

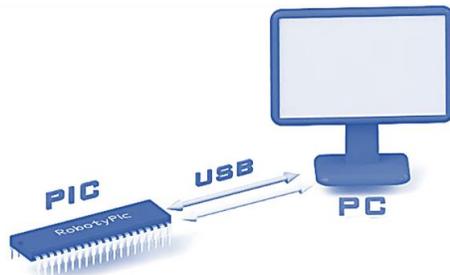


Figura 1.15: Representación general para la comunicación CDC.

El Microcontrolador utilizado para el control de actuación es un PIC18F4550 fabricado por la empresa Microchip Technology. Se eligió este chip por la capacidad de manejar Timers, así como enviar y recibir datos por medio del puerto USB de una PC, evitando el uso de convertidores USB a UART reduciendo las componentes del sistema. Asimismo forma parte de la familia PIC18 (Gama Alta), la cual posee un repertorio de 75 instrucciones, su memoria no es dividida en bancos y tiene una pila de 32 niveles que lo hace más compatible y eficiente con la programación en C.

1.3.1. Modulo USB del PIC18F4550

El modulo USB está configurado para emular la comunicación RS-232 a través de comunicación CDC o la HID (Dispositivo de Interfaz Humana), la segunda se utiliza como dispositivo de interfaz hombre computador, como por ejemplo teclados, ratones, etc., una de sus características es que no necesita drivers para que se reconozca pues utiliza los estándares del sistema operativo y puede configurarse a USB Full Speed 2.0 (12 Mb/s), su desventaja es que pocos softwares soportan esta comunicación. La clase CDC por el contrario necesita drivers (mchpcdc.inf) para su reconocimiento dentro del sistema operativo, una vez instalados estos y al conectar el Microcontrolador a un puerto USB el sistema operativo automáticamente crea un puerto COM Virtual, el cual puede ser utilizado por softwares de control tales como Matlab, LabView, Visual Basic, etc. La velocidad máxima que puede utilizar esta clase es de 80 Kbytes/s (Bauds).

Para asignar un puerto COM Virtual, el Host de la PC debe detectar el nuevo dispositivo, “interrogarlo” con el fin de adquirir información necesaria para establecer la comunicación, como consumo de corriente, el VIP y PID, etc., si detecta a continuación le asigna una dirección. Este proceso de interrogación y asignación de dirección para establecer la comunicación se le llama enumeración.

Los descriptores del PIC están almacenados en la memoria no volátil, los que utilizamos para la comunicación son los VID (Vendor ID) y PID (Product ID), estos son dos números de 16 bits que identifican y distinguen al dispositivo del resto, estos valores son asignados bajo licencia y con el fin de comercializar el producto, Microchip facilita estos números:

VID	0X04D8
PID	0X000A

Estos valores también se encuentran en el Driver (mchpcdc.inf) proporcionado por Microchip y al momento de la conexión deben coincidir con los grabados en el PIC.

En esta etapa el Microcontrolador recibe de la PC el sentido de giro para cada servomotor el cual interpreta y establece el ancho del pulso de la señal PWM, esto se logra con interrupciones del timer0 a 50Hz o 20 ms, que es la frecuencia a la que operan los servomotores.

1.3.2. Módulo TIMER

Los TIMER o temporizadores son módulos integrados en el PIC que permiten realizar cuentas tanto de eventos internos como externos, cuando la cuenta es interna se habla de temporización y cuando la cuenta es externa se habla de contador, los Timers están íntimamente ligados al uso de interrupciones.

Las interrupciones permiten a cualquier suceso interior o exterior interrumpir la ejecución del programa principal en cualquier momento. Al producirse una interrupción, la secuencia de ejecución salta automáticamente a la dirección del vector de interrupción de la memoria del programa, al mismo tiempo habilita una bandera indicando que se inició, posteriormente ejecuta la porción del programa correspondiente a la atención de la interrupción, hasta encontrar la instrucción que lo obliga a salir (RETFIE). Al encontrar dicha instrucción abandonara la interrupción, al mismo tiempo deshabilitara la bandera y retornara a la posición de memoria del programa principal desde la que salto al producirse la interrupción más una, posteriormente continua con la secuencia programada [10].

Esta etapa el Microcontrolador recibirá información, la cual analizará e interpretará con el propósito de coordinar la actuación de los motores haciendo uso de las herramientas anteriores explicadas.

1.4 SERVOMOTORES

Es un mecanismo similar a un motor de corriente directa que puede ubicarse en cualquier posición dentro de su rango de operación y controlado tanto en velocidad como en posición por medio de una retroalimentación con acoplamiento mecánico.

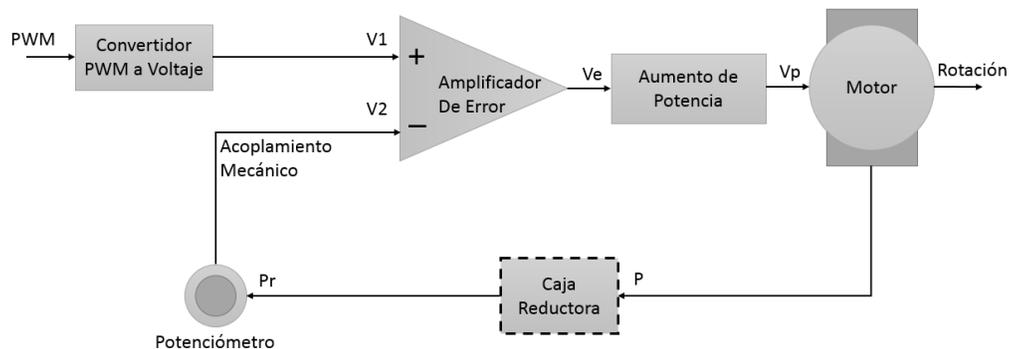


Figura 1.16: Retroalimentación interna del servomotor.

Está conformado por un motor, una caja reductora y un circuito de control, la corriente que requiere el dispositivo depende del par especificado por el fabricante y de la carga que se oponga a su libre movimiento.

- Motor. Es de corriente directa y es el componente de actuación.
- Caja Reductora. Convierte el par de salida en movimiento rotacional que puede ser interpretado por el potenciómetro, se trata de engranes interconectados.
- Potenciómetro. Supervisa el ángulo actual del motor, convierte el movimiento mecánico en diferencia de potencial que envía a comparar.
- Amplificador. Compara el voltaje deseado de la señal PWM con el real del potenciómetro, si es difiere acciona el motor hasta lograr el ángulo esperado de la señal PWM.

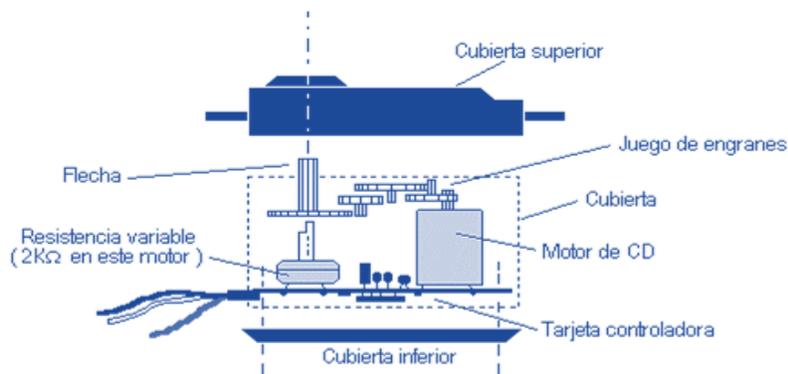


Figura 1.17: Composición interna física del servomotor.

Esta clase de motores necesitan un pulso constante de PWM (Pulso Modulado por Anchura) tienen capacidad de ubicarse y mantenerse estables en una posición o moverse a distintas velocidades, operando a una frecuencia de 50 Hz (20 ms) y variando el ciclo de trabajo entre 2.5% y 12.5% dependiendo de fabricante, es decir, el ancho de pulso en alto se encuentra entre (0.5ms – 2.5ms), con estos cambios es posible cambiar el sentido de giro, la posición y la velocidad.

Esta etapa de actuación constantemente recibe señales y reacciona para satisfacerlas, su efecto final se percibe en el movimiento del campo de visión de la cámara.

CAPÍTULO 2

ALGORITMOS DE VISIÓN ARTIFICIAL PARA DETECCIÓN Y SEGUIMIENTO

2.0 INTRODUCCIÓN

Con el incremento en la velocidad de procesamiento de las computadoras, la creación de nuevos algoritmos de visión artificial en paquetería de libre acceso y la reducción de costo y tamaño en las cámaras de alta resolución han logrado que aumenten las aplicaciones de los sistemas automáticos de seguimiento visual en la actualidad.

Determinar la posición de cualquier objeto en el espacio a partir del conocimiento de los fotones que se reflejan en el objeto, forma uno de los problemas fundamentales en la Visión Artificial. Los métodos para determinar la posición de un objeto y estimar su movimiento se continúan aplicando en diversos campos, como: la navegación de sistemas autónomos en robótica móvil, seguimiento en sistemas de vigilancia y estimación de movimiento en robótica de manipuladores.

La detección de objetos es importante en la visión artificial, porque de igual manera que los humanos (enfocan la mirada en un solo objeto a la vez despreciando el entorno que rodea al objeto), la visión artificial también reproduce esta acción, porque al eliminar la información irrelevante (entorno que rodea al objeto), reduce el cálculo computacional, para enfocarse solo en el objeto en la imagen.

Los métodos para detectar objetos se pueden clasificar en dos de manera general, los detectores de objetos conocidos y desconocidos.

Los detectores de objetos conocidos (o patrones) son los que utilizan información previamente suministrada por el programador, entre ellos los más destacables son: detección por índice de correlación y el detector cascada, el primero es eficiente pero su tiempo de computo es elevado además es necesario establecer el índice de correlación de una sola muestra que puede no ser detectada al menor cambio de iluminación, en cambio el segundo es tan eficiente como el programador lo requiera, ya que puede ser entrenado con una gran variedad de muestras lo que incrementa en gran medida la detección del objeto, llegando a ser implementado para sistemas en "Tiempo Real".

Los detectores de objetos desconocidos son los que logran detectar e interpretar alguna característica destacable de los objetos dentro del campo visible de la cámara, se denominan así pues no se proporciona una matriz donde este el objeto de interés, las características "desconocidas" pueden ser:

- Color: se encuentra en todos los objetos y se pueden distinguir por esta particularidad si el objeto tiene un color sólido, si es así se puede detectar con una simple umbralización sobre las matrices del espacio de color, para el color a detectar.

-
- Figuras primitivas, se encuentran en casi todas las figuras creadas por el hombre, tales como círculos, líneas, cuadrados, triángulos, que contenga el objeto a detectar, se pueden extraer con el algoritmo de la Transformada de Hough.
 - Texturas, se encuentran en todos los objetos con volumen o con imperfecciones (rugosos) estas son detectables con técnicas de detección de esquinas prominentes en un objeto, como el algoritmo de Harris.

El problema de detectar estas características “desconocidas” es que no se pueden diferenciar de las que corresponden al objeto en cuestión de las que pertenecen a la escena, solo son utilizables en espacios controlados y con información a priori, lo que dificulta su aplicación en lugares cambiantes o con movimiento.

Para solucionar esto existen algunos métodos para la detección de objetos desconocidos que no se basan en las características del objeto sino en su movimiento, es decir utilizan algoritmos que detectan variaciones en la luminosidad en un fondo estático producidas por objetos en movimiento (Mescla de Fondos Adaptables), o utilizan algoritmos que detectan texturas y las correlacionan en movimiento para generar vectores en un fondo estático y así detectar al objeto móvil (Flujo Óptico).

2.0.1 Métodos de seguimiento de objetos

Los métodos para seguir un objeto en consecutivos frames son diversos, estos métodos buscan un indicador que de la correspondencia entre objetos similares a través de frames secuenciales y converger en el mismo objeto (el que se desea seguir), para tal fin existen problemas a los que se enfrentan los algoritmos como al cambio de forma del objeto (rotación), alejamiento o acercamiento del objeto (traslación) y pérdida parcial o total del objeto (oclusión), pero el más importante es el cambio drástico de iluminación o la ausencia total de esta.

Los algoritmos que pueden en mayor medida hacer frente a estos desafíos son CAMShift y K-L-T, el primero sigue el histograma de color, el segundo sigue características destacadas en la textura del objeto.

Debido a las características propias del sistema es necesario utilizar algoritmos que utilicen un reducido tiempo de cómputo.

2.1 DETECTOR CASCADA

Clasificador Cascada y Entrenador de Clasificadores basado en AdaBoost. Los ingenieros Paul Viola de Mitsubishi Electric Research Labs y Michael Jones de COMPAQ CRL durante el diseño de un algoritmo de detección de rostros en una imagen, con un costo computacional muy bajo para lograr una alta tasa de detección, publicaron en 2001 un artículo que constaba de dos partes: un algoritmo de detección de objetos que emplea la clasificación cascada y un entrenador de clasificadores basado en (**Adaptative Boosting**) AdaBoost.

El algoritmo para la detección no utiliza directamente la imagen sino una representación de la misma, llamada imagen integral, que permite evaluar de manera muy rápida las características utilizadas. Una de sus ventajas es que puede calcular las características en cualquier lugar de la imagen y en cualquier escala en el mismo lapso de tiempo.

Para determinar si en una imagen se encuentra o no el patrón a seguir, el algoritmo divide la imagen integral en subregiones de tamaño variable y utiliza una serie de clasificadores (etapas), cada una con un conjunto de características visuales. Este tipo de clasificación es denominada: clasificación cascada. Este clasificador se construye seleccionando un número reducido de significativas características, utilizando el algoritmo AdaBoost como entrenador.

En cada etapa se determina si en la subregión está el patrón o no. Si la subregión es aceptada como patrón a seguir, entonces es evaluada por la siguiente etapa (que es más rigurosa) y si no se discrimina.

La clasificación en cascada garantiza la discriminación rápida de subregiones que no sean un patrón a seguir. Lo que se traduce como ahorro computacional, dado que no se procesaran subregiones de la imagen que con certeza no contenga un patrón y solo se invertirá cálculo computacional en aquellas que posiblemente si contengan. Este método tiene una tasa de verdaderos-positivos de 99.9%, y una tasa de falsos-positivos de 33.3%. [11]

2.1.1. Clasificación de Imágenes con patrones

Las imágenes que se generan durante el entrenamiento y en la aplicación se clasifican generalmente como positivas y negativas, las primeras contienen al patrón y las segundas no, sin embargo existen situaciones en las que el algoritmo llega a confundirse, estas situaciones se designan como:

- Verdaderos Positivos. Se produce cuando una muestra positiva se clasifica correctamente (el patrón realmente está en la imagen y el algoritmo lo reconoce).

-
- Verdaderos Negativos. Se produce cuando una muestra negativa se clasifica correctamente (el patrón no se encuentra en la imagen y el algoritmo lo reconoce).
 - Falsos Positivos. Se producen cuando una muestra negativa se clasifica erróneamente como positivo (el patrón no se encuentra en la imagen y el algoritmo deduce que sí).
 - Falsos Negativos. Se producen cuando una muestra positiva se clasifica erróneamente como negativo (el patrón realmente está en la imagen y el algoritmo deduce que no).

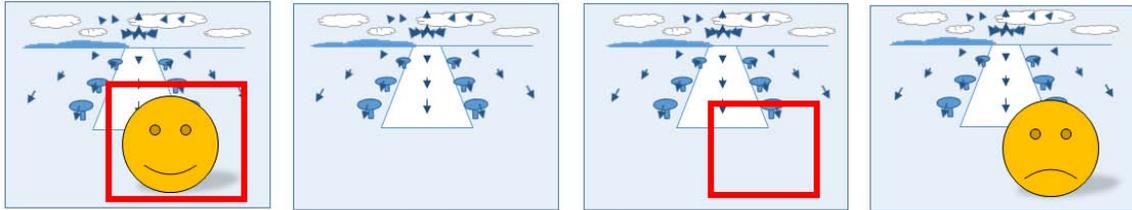


Figura 2.1: Tipos de clasificaciones; de izq. a der; Verdadero Positivo, Verdadero Negativo; Falso Positivo, Falso Negativo.

2.1.2. Extracción de características

En imágenes las características de cada objeto se extraen al aplicar ciertas funciones que permitan la representación y descripción de los objetos de interés en la imagen (patrones). La extracción de características es un paso en el reconocimiento de patrones en el cual las medidas u observaciones son procesadas para encontrar atributos que pueden ser usados como objetos a determinada clasificación [13].

El método propuesto por [11] plantea utilizar para la extracción de característica los filtros con base Haar, sin embargo existe la posibilidad de utilizar LPB (Patrones Locales Binarios) y HOG (Histogramas de Gradientes Orientados), cada uno con ventajas y desventajas.

2.1.2.1. Características HOG

El gradiente de pixel, es un vector que indica la dirección en la cual se produce un mayor cambio en la intensidad o en el color de la imagen. Los descriptores de HOG se basan en la orientación del gradiente en áreas locales de una imagen. Se obtienen dividiendo la imagen en celdas, calculando el histograma de la dirección del gradiente o de la orientación de los bordes en cada una de las celdas, se suele normalizar el contraste de cada uno de los histogramas, para hacerle frente a los cambios de iluminación o sombras.



Figura 2.2: Extracción de vectores orientados de un Drone.

2.1.2.2. Características Haar

El filtro Haar puede ser calculado eficientemente sobre la imagen integral, son selectivos en la orientación espacial y pueden ser modificados en escala y orientación. Realizan una codificación de diferencia de intensidades en la imagen, generando características de contornos, puntos y líneas, mediante la captura de contraste entre regiones.

Estas características son 'características rectángulo', existen diversas formas de dichas características, pero todas ellas se basan en la suma y diferencia de los valores de los píxeles de las áreas dentro de rectángulos.

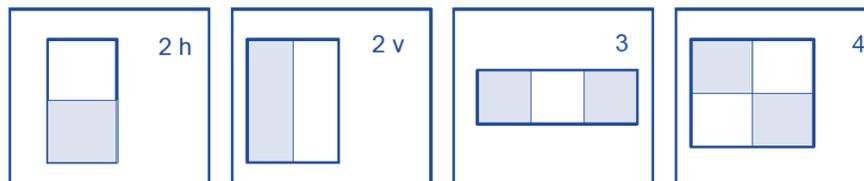


Figura 2.3: Principales características rectángulo.

El valor de la característica "Dos-Rectángulos" es la diferencia entre la suma del valor de los píxeles dentro de cada región, es decir, se suman los valores de los píxeles dentro de una región (mitad del rectángulo donde se evalúa la característica) y se le resta del valor de los píxeles de la otra región (la otra mitad del rectángulo).

El valor de la característica Tres-Rectángulos es la diferencia entre la suma del valor de los píxeles dentro de los rectángulos exteriores y el rectángulo interior.

El valor de la característica Cuatro-Rectángulos es la diferencia entre la suma del valor de los píxeles dentro de los rectángulos en una diagonal del valor y la suma de los píxeles dentro de los rectángulos de la otra diagonal.

En una posterior mejora, se añade dos tipos nuevos de orientaciones tipo Haar, aumentando su efectividad.

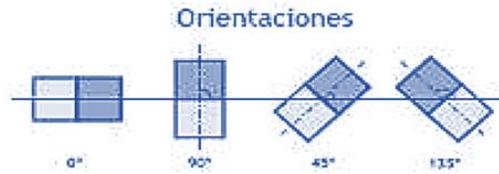


Figura 2.4: Orientaciones mejoradas para las características rectángulo.

2.1.3. Integral de la Imagen o tablas de sumatoria (SAT)

Esta imagen permite extraer de forma rápida características rectángulo a diferentes escalas ya que no se trabaja directamente con los valores de intensidad, sino con una imagen acumulativa que se construye a partir de operaciones básicas.

La imagen integral en un punto (x, y) de la imagen base, es la suma de todos los píxeles contenidos en el rectángulo formado por el origen $(0,0)$ y el punto (x, y) .

$$II(x, y) = \sum_{xi \leq x, yi \leq y} I(x_i, y_i) \quad (2.1)$$

Donde $II(x, y)$ es la imagen integral en el punto (x, y) e $I(x, y)$ es el valor de los píxeles de la imagen original.

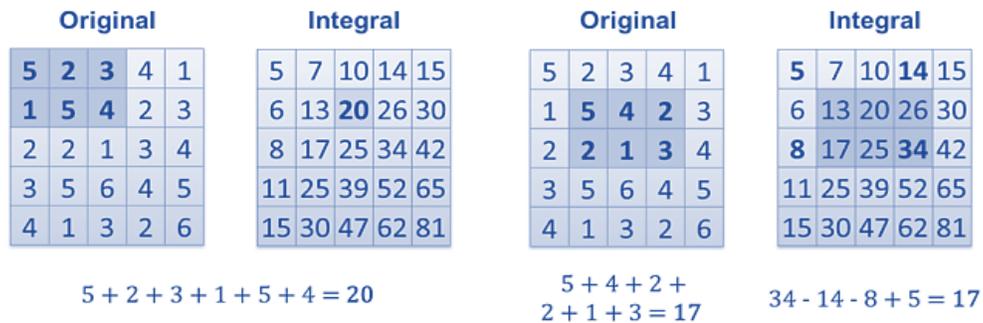


Figura 2.5: Forma de obtener la imagen integral y las características rectángulo.

Esta pequeña matriz muestra la forma sencilla de encontrar la imagen integral y las características rectángulo.



Figura 2.6: Comparativa, entre imagen integral y la original.

Utilizando la siguiente recursividad, se puede obtener la imagen integral en una sola pasada sobre la imagen original:

$$S(x, y) = S(x, y - 1) + I(x, y) \quad (2.2)$$

Con $S(x, -1) = 0$ y $I(-1, y) = 0$

$$H(x, y) = H(x - 1, y) + S(x, y) \quad (2.3)$$

Donde $S(x, y)$ es la suma hasta y de la fila x

Utilizando la imagen integral cualquier suma dentro de un rectángulo puede ser calculada en cuatro referencias de la tabla.

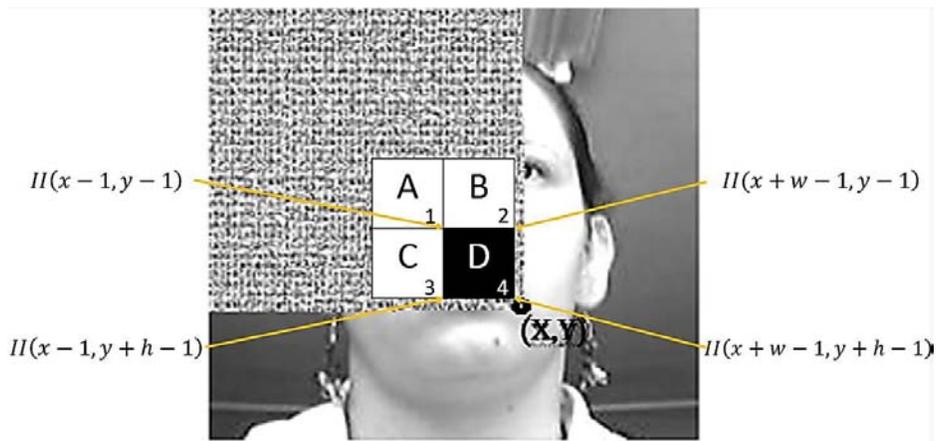


Figura 2.7: Extracción de características rectángulo.

Accediendo al resultado de la imagen integral de los puntos 1, 2, 3, 4, se puede calcular en forma sencilla y rápida la suma de los pixeles dentro de cualquiera de los rectángulos.

La suma de los píxeles dentro del rectángulo D puede ser calculada fácilmente como.

$$sumD = (4 + 1 - 2 - 3) \quad (2.4)$$

Donde:

$$\begin{aligned} sumD(x, y, w, h) = & II(x - 1, y - 1) \\ & + II(x + w - 1, y + h - 1) \\ & - II(x - 1, y + h - 1) \\ & - II(x + w - 1, y - 1) \end{aligned} \quad (2.5)$$

Por lo tanto la resta del área de dos rectángulos cualesquiera puede ser calculada en 6 referencias debido a que los rectángulos son adyacentes, 8 para el caso de 3 rectángulos y 9 para los de 4 rectángulos [12].

2.1.4. Arquitectura Cascada

La razón por la cual se utiliza, es por la capacidad de los clasificadores entrenados por AdaBoost de poder rechazar la mayoría de las sub-ventanas (débiles aprendices), dejando pasar todos los casos positivos. Estos clasificadores se ponen en las primeras etapas de manera que en pocas operaciones se pueda rechazar la mayoría de las ventanas, dejando así los clasificadores más complejos concentrarse en pocos casos.

Al saber que muy pocas de la mayoría de las ventanas a analizar contienen al objeto, la gran mayoría de las mismas deberán ser rechazadas.

La estructura del clasificador adopta la forma de un árbol de decisiones. Si el primer clasificador da por buena una ventana, esta pasa al segundo clasificador que también ha sido entrenado para producir una alta tasa de detección, sino la ventana se descarta. El segundo clasificador la evalúa, si da por buena se la pasa al tercer clasificado, sino la descarta y así continua todo el proceso.

El número de etapas del clasificador está determinado por la cantidad de imágenes que se disponen para su entrenamiento.

Se puede deducir, que en una imagen normal la gran mayoría de ventanas exploradas serán negativas. Por lo que en el proceso es deseable rechazar tantas ventanas como sea posible, lo antes posible para acelerar el proceso.

Por lo tanto a medida que avanzan los clasificadores, estos tienen cada vez una tarea más difícil, solo ven las imágenes que ya pasaron por las otras etapas, es frecuente que estos clasificadores se entrenen con imágenes que sean falsos positivos de otras etapas.

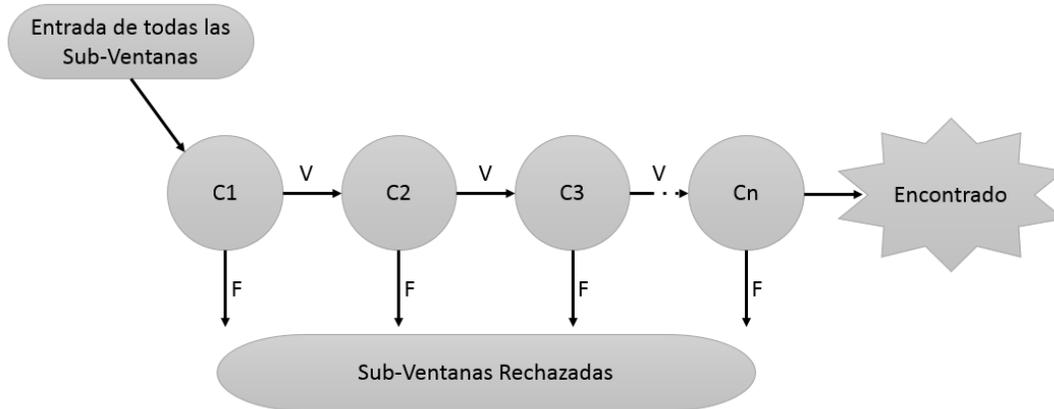


Figura 2.8: Esquema Cascada

2.1.5. Algoritmo del detector cascada

1. Se debe tener un conjunto de imágenes donde se encuentre el objeto a detectar las cuales se llaman imágenes Positivas. También se debe contar con un conjunto de imágenes donde no se encuentre el objeto y que muestren el entorno donde posiblemente aparezca el objeto a detectar, a estas se les llaman imágenes Negativas.
2. Establecer las metas de detección por etapa, por lo usual la tasa de verdaderos positivos se establece entre 85 y 95%, y la tasa de falsos positivos en el orden de menos de 10%.
3. Entrenar con el Algoritmo AdaBoost, estableciendo las características de entrenamiento (Haar – LPB - HOG) y el número de etapas. Al final del entrenamiento se conseguirá un archivo XML en cual tiene las instrucciones que se ejecutarán en cada imagen secuencial para detectar el objeto de interés.

2.1.5.1. Funcionamiento del archivo .XML

En cada etapa divide la imagen en regiones etiquetando cada una si es una posible positiva o negativa, si es etiquetada como negativa la clasificación es esa región termina, y el detector recorre la siguiente región, si es etiquetada como positiva, el clasificador la promueve a la siguiente etapa. Y continua así hasta terminara todas las etapas, al final informa de la o las regiones que han sido clasificadas como positivas en el frame actual, es decir, si encontró el objeto de interés.

2.2 DETECTOR DE MOVIMIENTO

La detección del primer plano tiene como objetivo principal, detectar cambios en la escena dentro de secuencias de imágenes.

El método fue creado para la detección de movimiento en sistemas de seguridad en cámaras sin movimiento, se basa en crear un fondo de la escena (modelo) que siempre permanecerá estática generalmente con la ausencia de objetos a detectar, una vez creado este fondo se umbralizará, es decir, se creará una imagen binaria con los objetos que se muevan entre el entorno y la cámara, al que se llamará primer plano.

El problema usual que tiene el algoritmo ocurre cuando capta movimiento “involuntario”, es decir, el movimiento que es producido por el entorno, estos movimientos lo crean los arboles con el viento o las sombras que se proyectan por el cambio de fuente o perspectiva de iluminación. Para solucionar el problema del movimiento falso se propone que el modelo del fondo sea adaptativo [14] para hacer frente a los cambios de iluminación graduales en las escenas repetitivas y de largo plazo.

Este método de detección segmenta las regiones en movimiento formadas por secuencias de imágenes, el cual debe adaptarse, es decir, debe evolucionar (actualizarse) para que los lugares donde existía movimiento se consideren ahora como fondo sin movimiento, este algoritmo debe distinguir entre sombras y movimiento de objetos en entornos cambiantes.

El método necesita un periodo de formación o aprendizaje para modelar el fondo, lo logra haciendo uso de las imágenes capturadas en ese intervalo de tiempo, consiguiendo que todos los objetos en el primer plano inmóvil se consideren permanentes, es decir, como fondo, posteriormente cualquier objeto que se mueva y este iluminado podrá ser detectado y mostrado como objeto binario.

2.2.1. Funcionamiento del detector de movimiento

La función de probabilidad gaussiana está determinada por dos parámetros: la media y la varianza, la media es una medida del promedio del brillo y la varianza es la medida del contraste para los histogramas de los tres canales del modelo de color RGB, la forma de esta función matemática se observa en la siguiente figura:

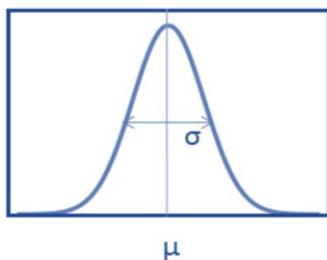


Figura 2.9: Función gaussiana.

El método se diferencia de sus antecesores por utilizar k funciones gaussianas con diferentes estados en función del tiempo, [15], aquí proponen este método para detectar los objetos del primer plano, mediante la suma ponderada de funciones Gaussianas por canal y pixel.

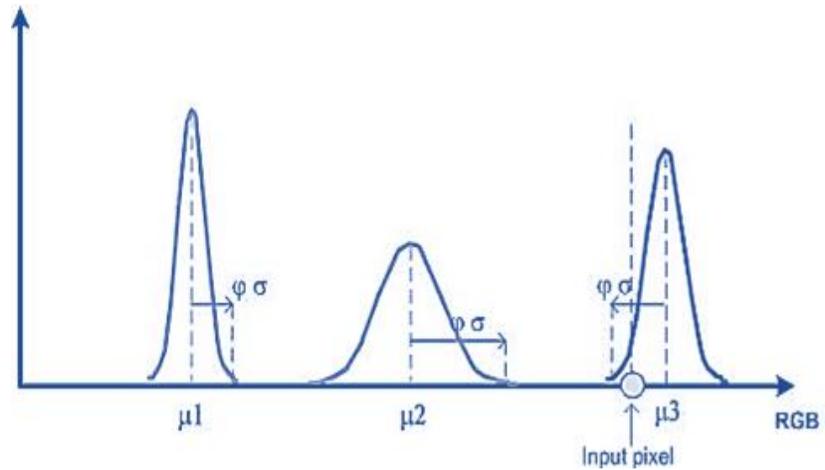


Figura 2.10: Múltiples gaussianas.

Es decir, para cada pixel se proporciona múltiples gaussianas que corresponden a los posibles estados.

Para cada pixel:

- Diferentes gaussianas modelan diferentes estados (colores) de la escena.
- Cada gaussiana puede representar un modelo de fondo o de primer plano.

La distribución de cada pixel se define por:

$$P(X_t) = \sum_{i=1}^k w_{i,t} * \eta(X_t, \mu_{i,t}, \Sigma_{i,t}) \quad (2.6)$$

Donde:

K - es el número de distribuciones gaussianas (normalmente entre 3 y 5 funciones distintas).

η - es la función de la probabilidad gaussiana

$w_{i,t}$ - son las ponderaciones (pesos) que tiene cada curva en el tiempo t .

$\mu_{i,t}, \Sigma_{i,t}$ - son la media y la varianza de cada función gaussiana en el tiempo t .

X_t - Es el valor del pixel en cuestión en el tiempo t .

Esta fórmula se utiliza para obtener el modelo del fondo y decide si el pixel en cuestión se corresponde con el modelo o con un objeto del primer plano.

Sin embargo, como las gaussianas representan tanto el fondo como el primer plano se necesitará un criterio para decidir de manera dinámica cuales modelan el fondo y cuales el primer plano. Consiguiendo de esta manera introducir los objetos del primer plano en el modelo de fondo, cuando permanezcan estáticos en la escena, se asignará la gaussiana correspondiente al fondo.

Este criterio se establece sabiendo que el fondo está modelado por las B distribuciones gaussianas de mayor peso y menor varianza como lo indica:

$$B = \operatorname{argmin}_b \left(\sum_{k=1}^b w_{i,t} > T \right) \quad (2.7)$$

Donde:

T – es el umbral de decisión (comúnmente 0.6).

B – es el número mínimo de distribuciones gaussianas para cada pixel.

Las B funciones del fondo son determinadas por las k gaussianas de cada pixel que son ordenadas por su peso $w_{i,t}$, eligiéndose las primeras, esto ocurre porque con el tiempo se va haciendo más estático y aparece con más frecuencia correspondiéndose con aquellas gaussianas que han sido utilizadas más veces.

Para decidir si el pixel analizado corresponde al fondo o no se evalúa con:

$$|x_t - \mu_{i,t}| > k\sigma_{i,t} \quad (2.8)$$

Donde:

x_t – es el pixel en cuestión del Frame t .

$\mu_{i,t}$ – es la media de la gaussiana i -ésima a evaluar.

$\sigma_{i,t}$ – es la varianza de la gaussiana i -ésima a evaluar.

k – es el umbral a especificar (normalmente es 2.5).

Si la función gaussiana en cuestión cumple para todas las gaussianas, se decide que es primer plano, ya que no encaja con el modelo aprendido hasta ese momento. En consecuencia generando una nueva gaussiana con el fin de permitir adaptar el modelo al fondo, al mismo tiempo eliminando la distribución de menor peso [15].

Si la función gaussiana en cuestión no cumple, se supone que valor encaja con el modelo de fondo (ya sea las B primeras o las posteriores de menor peso), en consecuencia se asignara como fondo.

Finalmente se actualizará el modelo de fondo a medida que los pixeles correspondan con una de las k gaussianas con las siguientes expresiones:

$$\mu_t = \rho x_t + (1 - \rho)\mu_{t-1} \quad (2.9)$$

$$\sigma_t^2 = \rho(x_t - \mu_t)^2 + (1 - \rho)\sigma_{t-1}^2 \quad (2.10)$$

Actualización de la media y la varianza (afecta sólo a la distribución gaussiana que ha encajado).

$$w_{k,t} = (1 - \alpha)w_{k,t-1} + \alpha(M_{k,t}) \quad (2.11)$$

Actualización de los pesos (afecta a todas las distribuciones que conforman el modelo).

Donde:

α - es un coeficiente de absorción (normalmente es un valor de 0.0005).

$M_{k,t}$ - es 1 para la función gaussiana que ha encajado y 0 para las que no.

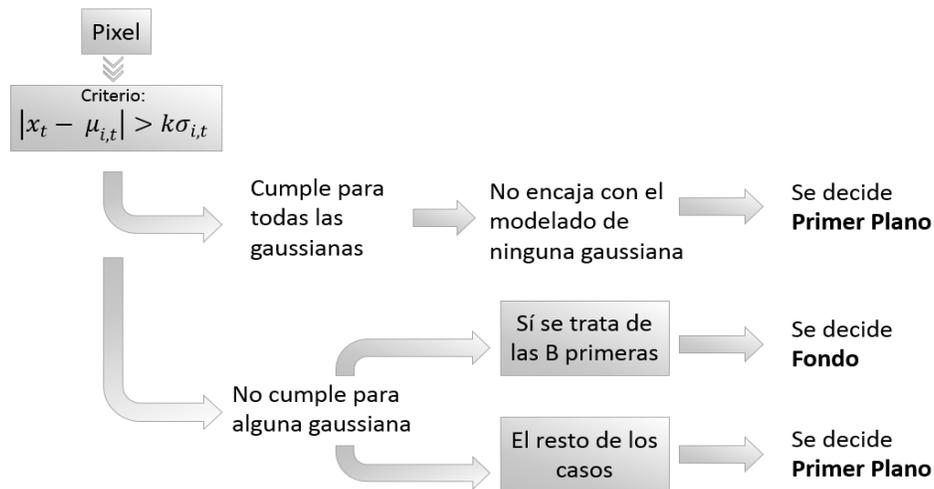


Figura 2.11: Esquema general de funcionamiento.

2.2.2. Algoritmo para detección del primer plano

1. Establecer el modelo de fondo
2. Establecer el numero inicial de frames para entrenar el modelo de fondo
3. Establecer la tasa de aprendizaje, rapidez de actualización
4. Establecer el umbral para determinar que pixeles formaran parte del fondo y del primer plano

2.3 CAMSHIFT

CAMShift (Continuously Adaptative Mean Shift - Adaptación Continua del Corrimiento a la Media). Este algoritmo fue propuesto por Gary R Bradski [16] quien hizo una mejora al método Mean-Shift (o corrimiento hacia la media) [17] para hacer frente a los cambios dinámicos en la distribución de probabilidad de color producidos entre frames consecutivos, mejorando la robustez al ruido, distorsiones y traslación del objeto seguido.

El algoritmo Mean-Shift, encuentra las modas de la densidad, es decir las zonas en las que hay una mayor densidad de datos, consta de un proceso iterativo de punto fijo el cual converge a un máximo local de la densidad obteniendo en cada paso una estimación del gradiente de dicha función de densidad a partir de la muestra o imagen de probabilidad.

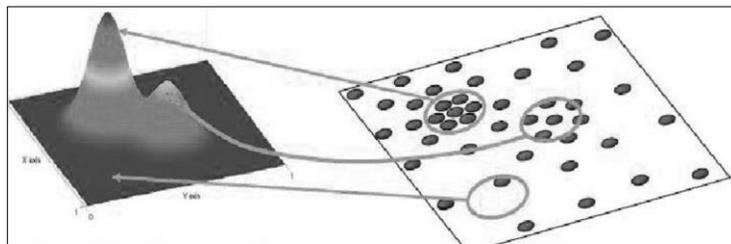


Figura 2.12: Estimación de la densidad a partir de la muestra

2.3.1. Imagen de probabilidad

Para calcularla se puede emplear cualquier método que asocie a un valor de pixel una probabilidad. El método comúnmente empleado trata de una retroproyección de histograma, esta operación consiste en generar una imagen en escala de grises donde cada pixel tendrá como intensidad el valor del histograma correspondiente al matiz de dicho pixel en la imagen procesada, y así el valor de cada pixel de esta imagen identificará la probabilidad de que dicho pixel en la imagen procesada pertenezca al objeto. Para su cálculo se suele emplear un histograma de color obtenido del canal (H) del modelo de color HSV dentro de la región de interés (ROI) que se denomina como Kernel K el cual es una matriz definida positiva y de dimensión $n \times m$.

El histograma obtenido $h(x)$ se escala de forma que se encuentre en el rango $[0, 255]$.

$$h(x) = h(x) \frac{255}{\max_{x=0}^{255}(h(x))} \quad \forall x \in [0, 255] \quad (2.12)$$

De este modo, el histograma representará la distribución de probabilidad del color del objeto a seguir, es decir, la distribución de probabilidad del objeto.

Comúnmente se utiliza el canal (H) ya que se representa los colores absolutos del espectro visual, pero se puede utilizar otro modelo o un histograma multidimensional, pero esto ocasionaría una mayor carga para el procesador.

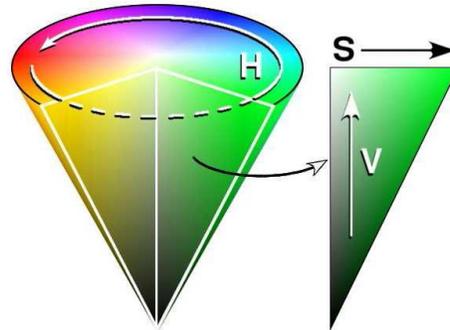


Figura 2.13: Modelo de color HSV (Hue-Saturation -Value).

2.3.2. Centro de masas (Centroide)

Para obtener el cuadro delimitador que encierra al objeto se debe calcular el centro del objeto, en la cual se calcula el nuevo tamaño y posición en cada frame, con esto se logra obtener los momentos $I(x, y)$. En el cálculo de momentos se emplea la intensidad de la imagen de probabilidad en el punto (x_c, y_c) que es la ubicación del Centroide.

El momento de orden 0 (M_{00}) y los momentos de orden 1 (M_{10}, M_{01}) se relaciona directamente con el Centroide con las siguientes formulas:

$$x_c = \frac{M_{10}}{M_{00}} = \frac{\sum_x \sum_y xI(x, y)}{\sum_x \sum_y I(x, y)} \quad (2.13)$$

$$y_c = \frac{M_{01}}{M_{00}} = \frac{\sum_x \sum_y yI(x, y)}{\sum_x \sum_y I(x, y)} \quad (2.14)$$



Figura 2.14: Comparativa de una Imagen en RGB contra una en HSV.

2.3.3. Convergencia

Se trata de un criterio de convergencia para detener el desplazamiento hacia la media. Puede estar basado en un número máximo de iteraciones, una distancia máxima entre centros de masas consecutivas. Debido a que en el caso en que un momento tenga el valor de 0 el algoritmo debe detenerse, puesto que ningún pixel tiene valor, lo que indica que el objeto se ha perdido.

2.3.4. Algoritmo para el seguimiento de histogramas

1. Seleccionar una región de interés (ROI) dentro de la imagen donde se espera se encuentre el objeto.
2. Calcular el histograma del canal Hue sobre la ventana de búsqueda en la primera imagen.
3. Calculo de la retroproyección del histograma pasado.
4. Iteración del algoritmo Mean-Shift hasta converger en el centroide de la imagen de probabilidad, almacenando la posición del centroide y el momento de orden cero.
 - a. Calcular el Centroides (x_c, y_c) en la ventana de búsqueda actual, utilizando el momento de orden 0 (M_{00}) y los momentos de orden 1 (M_{10}, M_{01}) .
 - b. Establecer el Centroides obtenido como nuevo centro de ventana de búsqueda.
 - c. Repetir a y b hasta converger.
5. En el siguiente fotograma se centra la ventana en la posición del Centroides y se actualiza su tamaño en función de momento de orden cero. Vuelve al punto (3).

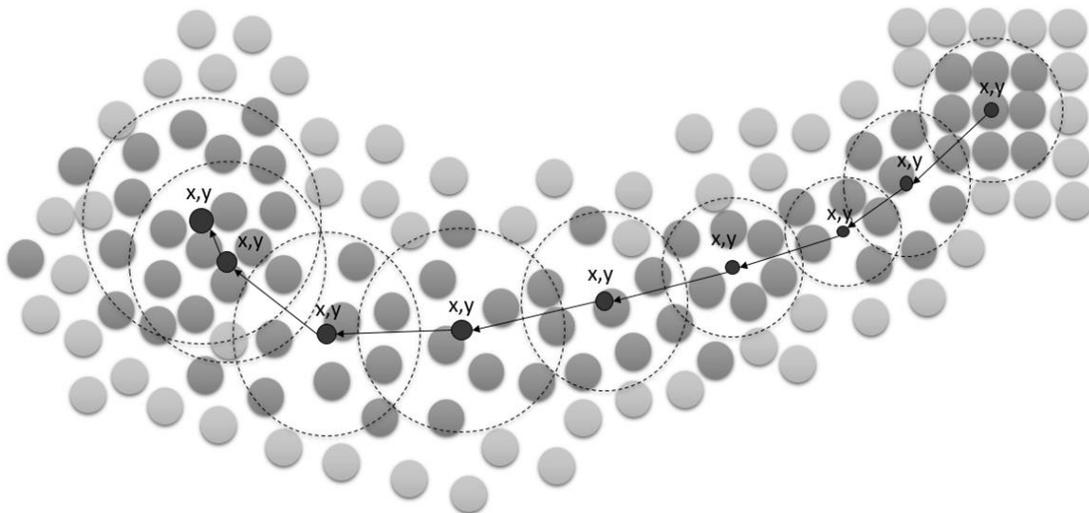


Figura 2.15: Evolución del Centroides en consecutivos frames

2.4 SEGUIMIENTO DE CARACTERÍSTICAS (KLT TRACKER)

Este algoritmo fue propuesto por Kanade - Lucas - Tomasi (KLT), funciona bien para el seguimiento de objetos que no cambian de forma y que exhiben texturas visuales destacables, El algoritmo encuentra la correspondencia de las esquinas en una secuencia de frames, sin embargo estos pueden perderse por efectos de un rápido cambio de iluminación o rotación del objeto, para superar esta inconveniente las esquinas deben ser cuidadosamente seleccionadas, de igual forma para hacer más eficiente el seguimiento el tiempo entre frames debe ser mínimo pues determinara el porcentaje de coincidencia para esquinas del mismo objeto.

2.4.1. Selección de puntos característicos

Para seleccionar estas esquinas característicos se debe considerar un criterio de selección, el cual detectará las más prominentes en la región de la imagen donde se encuentra el objeto, los cuales deben reunir:

- La distinción. Significa que el punto debe ser diferente de sus vecinos pixeles inmediatos.
- La unicidad. El punto debe ser distinguido globalmente, es decir, idealmente no debería aparecer en ningún otro punto de la imagen, puede provocar confusión en la búsqueda de correspondencia.
- La invariancia. La apariencia del punto no debe variar a consecuencia de distorsiones geométricas en su movimiento o en relación con la iluminación.
- La estabilidad. La apariencia del punto debe ser invariante respecto al punto de vista, es decir, puntos interesantes de la imagen deben corresponder a puntos de interés del objeto.

El método más utilizado para detectar las esquinas es el algoritmo "Harris", ya que encuentra rápidamente las esquinas, el procedimiento que utiliza para tal fin es, en primer lugar realizar un pre-procesamiento, es decir, realzar o suavizar la imagen para detectar los contornos más destacables de la imagen, los contornos permiten visualizar el inicio y fin de un objeto, así mismo los contornos son creados por el cambio drástico en los valores de intensidad de los pixeles en una dirección (llamado gradiente), este cambio de intensidad se calcula a partir de la primera derivada unidireccional en cada fila o columna de la matriz, una vez obtenido la imagen de contornos, se encuentran todas las esquinas, se pueden encontrar ya que las esquinas son puntos (pixeles) en la imagen donde colindan con diferentes bordes a la vez en diferentes direcciones. El inconveniente

que tiene este método es que encuentra todas las esquinas y la mayoría de ellas se hacen débiles durante el desplazamiento de la cámara.

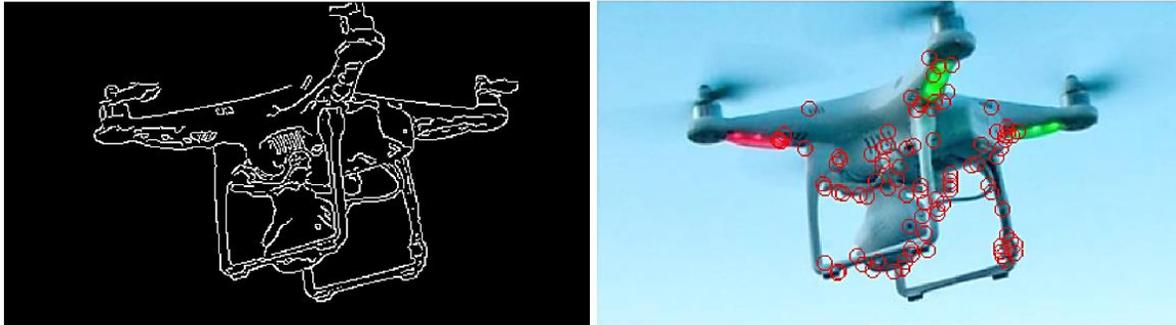


Figura 2.16: Extracción de contorno y de esquinas prominentes

Para satisfacer esta exigencia existe el algoritmo “Good Features to Track” desarrollado por Tomasi [18], el cual detecta las esquinas más características de las texturas, logrando que sean ideales para realizar un seguimiento con desplazamientos de la cámara.

2.4.2. Kanade Lucas.

Este modelo también recibe el nombre de método piramidal. El método se basa en el conocimiento de que el video se forma de imágenes secuenciales en pequeños instantes de tiempo, albergando texturas de posibles objetos, por lo tanto las esquinas de las texturas entre imágenes se encuentran fuertemente relacionadas.

Satisfaciendo la siguiente propiedad que está en función de las coordenadas $I(x, y)$ de la imagen y el tiempo t en frames posteriores:

$$I(x, y, t + \tau) = I(x - \xi, y - \eta, t) \quad (2.15)$$

Donde (ξ, η) es el desplazamiento de la esquina en el punto (x, y) entre instantes de tiempo t y $t + \tau$.

Incluso en un entorno estático bajo iluminación constante la propiedad anterior puede no ser cumplida, sin embargo en general es satisfecha en lugares donde la iluminación cambia abruptamente.

El principal problema al que se enfrenta este método es la búsqueda del máximo desplazamiento d de una esquina, de un frame a otro, el problema se agrava más aún si esta esquina no sobresale de

sus vecinas, pues en el transcurso de frames se puede confundir con los pixeles adyacentes. Como consecuencia es difícil seguir una esquina basándose en información local.

Para solucionar a estos problemas el algoritmo sigue ventanas de pixeles con suficiente textura. Desafortunadamente el patrón de intensidad puede deformarse de un frame a otro, así como entrar en oclusión (apareciendo y reapareciendo). Creando la interrogante si “realmente está siguiendo la misma ventana”. La solución que propone [19] es la vigilancia de residuos, es decir, comprobar que la nueva ventana no ha cambiado demasiado. Si es así, se desecha la ventana. El uso de pequeñas ventanas implica que con solo unos pocos parámetros pueden estimarse con fiabilidad.

La solución al máximo desplazamiento es minimizar el error residuo utilizando ventanas pequeñas.

De esta manera el algoritmo estima solo dos parámetros: el vector de desplazamiento para las pequeñas ventanas.

La ecuación que satisface el máximo desplazamiento es:

$$G\mathbf{d} = e \quad (2.16)$$

Donde:

G – matriz de coeficientes simétrica 2x2.

e – es un vector de dos dimensiones.

Para cada par de frames adyacentes la matriz G puede ser calculada a partir de un frame, mediante la estimación de gradientes. El vector e , por otro lado, se puede calcular a partir de la diferencia de dos frames junto con el gradiente calculado anteriormente sobre una ventana dada [19].

2.4.3. Algoritmo para seguimiento de esquinas

1. Obtener la ROI donde se encuentra el objeto a seguir.
2. Detectar los Puntos más sobresalientes del objeto utilizando el algoritmo “Good Features To Track”, especificando el tamaño del filtro Gaussiano para suavizado y resaltado de esquinas, y especificando la “mínima calidad” de esquinas para rechazar falsos puntos.
3. Inicializar el Seguidor configurando: Error bidireccional, Niveles piramidales, Vecindad, Iteraciones de búsqueda.
4. Comparar los puntos en cada frame, es decir, la relación con sus anteriores ubicaciones.

CAPÍTULO 3

CONTROL DE ACTUACIÓN Y PREDICCIÓN DE LA TRAYECTORIA

3.1 ESQUEMA MECÁNICO

Para poder obtener el panorama total de una escena con cámara estacionaria y utilizando un mínimo número de grados de libertad se utilizó los movimientos rotacionales Tilt y Pan, así mismo existe otro movimiento rotacional para cámara estática llamado Roll, esta maniobra no se implementó porque los algoritmos de seguimiento son robustos frente a la rotación del objeto.

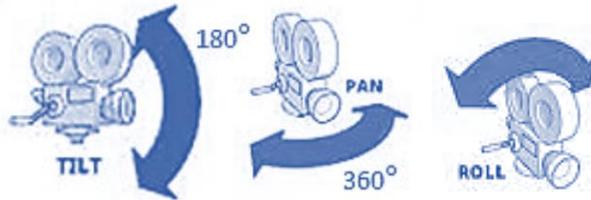


Figura 3.1: Movimientos rotacionales.

Si establecemos como origen la posición de la cámara y partiendo de ella se obtiene la descripción de un sistema de coordenadas esféricas.

Donde: θ – representa la posición angular de PAN.

φ – representa la posición angular de TILT.

ρ – representa la distancia entre la cámara y el objeto.

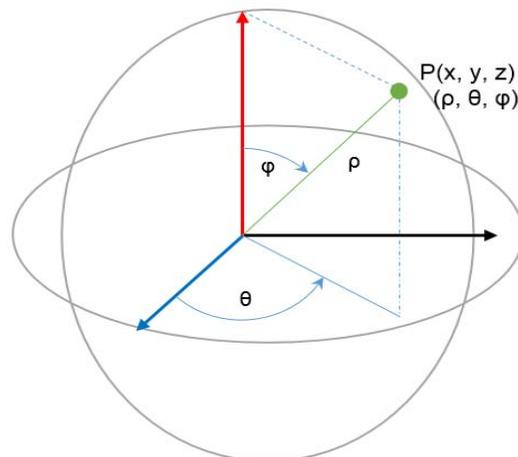


Figura 3.2: Sistema de coordenadas esféricas y cartesianas.

Conociendo cada parámetro es posible identificar la ubicación de un objeto y su posible desplazamiento, el cual se representaría como un vector de coordenadas que viaja en el espacio.

De igual forma es posible representar esta descripción esférica en un sistema de proyección, si bien esta técnica es común en la cartografía terrestre no se limita a esta, esta representación es adecuada para obtener las dimensiones reales del objeto.

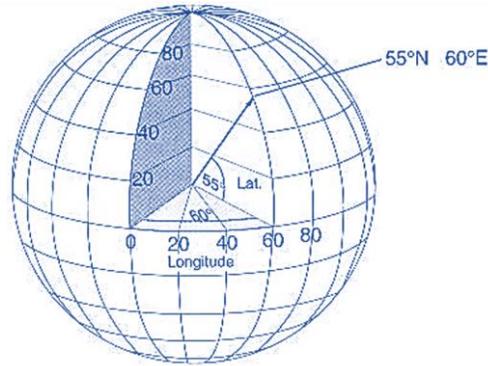


Figura 3.3: Sistema de proyección esférica.

La proyección se conoce como la transformación de un espacio tridimensional en uno bidimensional, este proceso distorsiona, la forma, superficie, distancia y dirección. La proyección plana consiste en que un sistema plano toque tangencialmente una parte de la esfera que por lo común se trata de un polo.

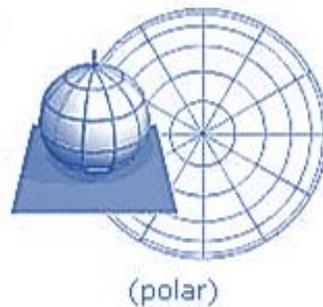


Figura 3.4: Proyección plana polar.

En esta representación plana los “paralelos” corresponden a círculos consecutivos a partir del polo, distanciados por un ángulo de apertura, los “meridianos” corresponden a líneas rectas que se interceptan en el polo con ángulos de intersección determinados por la cámara.

3.2 CONTROL

El control se define como una interconexión de componentes que forman una configuración del sistema que proporcionan una respuesta deseada, para realizar tal hazaña se debe tener un gran entendimiento de la variable a controlar.

El control al que se refiere este apartado es al que se aplica al sistema de actuación del sistema de seguimiento, a la velocidad rotacional que debe alcanzar la cámara para realizar un minucioso seguimiento, esta velocidad dependerá de la distancia a la que se encuentre el objeto de la cámara a consecuencia del siguiente fenómeno.

3.2.1. Perspectiva del movimiento

Este fenómeno ocurre en todos los objetos en movimiento y depende de la posición de quien realice la observación.

Si un objeto se mueve a una velocidad constante y pasa frente a una cámara, entrará y saldrá de su campo visual tan rápido, como se encuentre el objeto de la cámara, es decir, si el objeto se mueve a una cierta velocidad y se encuentra cerca de la cámara entrará y saldrá de su campo visual rápidamente; lo que en apariencia se puede visualizar que el objeto viaja a una velocidad mayor a la que realmente es. Si ahora consideramos que un objeto tiene la misma velocidad que el anterior y pasa a una distancia mayor respecto al observador, entrará y tardará más tiempo en salir del campo de visión, esto en apariencia se ve visualmente que viaja a una velocidad menor. Esto es consecuencia del ángulo de apertura del campo de visión de la cámara.

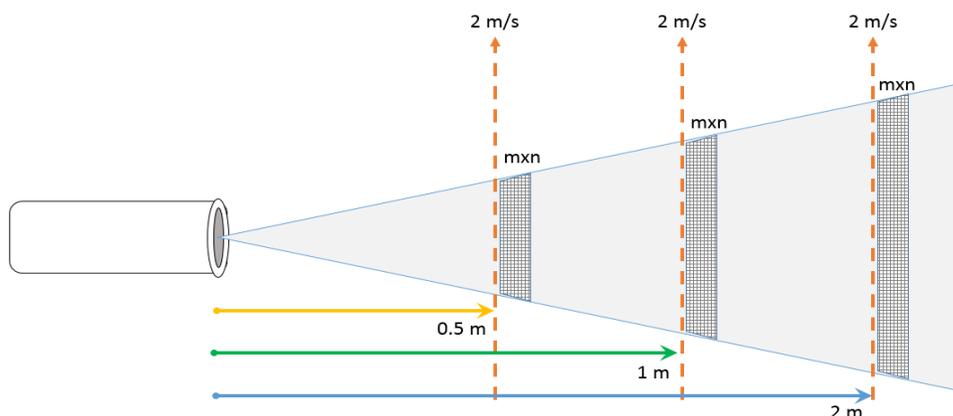


Figura 3.5: Fenómeno perspectiva del movimiento.

La noción de movimiento relativo entre los objetos estacionarios entre si y en relación con las diferentes distancias a las que se encuentran respecto del observador que se mueve a su vez respecto de los objetos es un fenómeno óptico y fue inicialmente estudiado por Euclides en su tratado de Óptica, es muy común que se presente cuando se viaja en vehículo, estando dentro de él, se puede observar como en apariencia los objetos más cercanos tienden a moverse más rápido que los que se encuentran a un a distancia mayor [7].

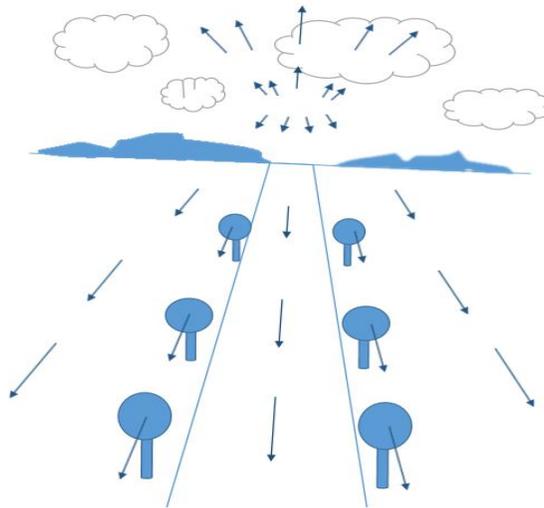


Figura 3.6: Campo de movimiento de un observador o de una cámara que se mueve respecto a la escena.

Ahora, si la cámara es capaz de moverse a una velocidad similar a la del objeto, sin importar la distancia a la que se encuentre, en la imagen capturada parecerá que el objeto se encuentra sin movimiento aparente y podrá realizarse la consecuente etapa, que es tratar de centrar el objeto en la imagen, pues con esto es posible realizar la medición de la distancia, predicción de la ubicación futura y la adaptación de la velocidad.

3.2.2. Objetivo de control

Esto es a lo que se quiere llegar con el sistema de control, es decir, el objetivo es lograr que el objeto se encuentre en todo momento en el centro de la imagen capturada, para poder desprestigiar los efectos de distorsión geométrica mencionados anteriormente y que los algoritmos de seguimiento visual puedan funcionar convenientemente.

Considerando lo anterior es necesario utilizar un sistema de control que trate de moverse a la velocidad del objeto con el fin de no perder las características visuales que siguen los algoritmos de visión. Se requiere un control que llegue de manera sobre-amortiguada, y no ocurran oscilaciones

como lo sería en una sub-amortiguación, pues estas ocasionaran que se pierdan las características visuales seguidas.

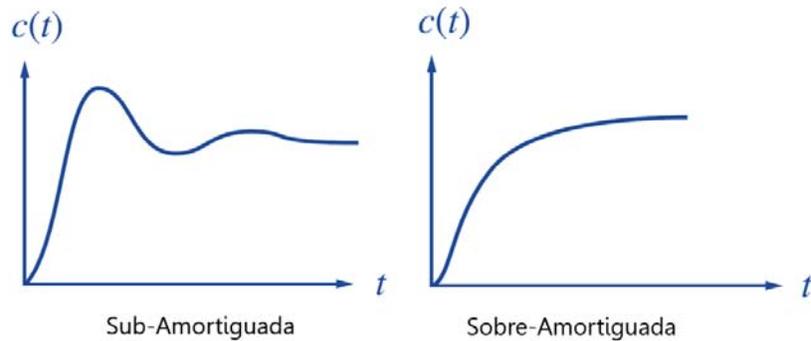


Figura 3.7: Respuestas comunes a la entrada escalón.

Se pensó en un inicio utilizar un control PID (Proporcional Integral Derivativo) Discreto pero experimentalmente se observó que ocurrían las oscilaciones al tratar de alcanzar el (Set Point), produciendo imágenes borrosas, y cuando el objeto cambió de velocidad o sentido empeoraba el resultado, esto se debió a la inercia y el tiempo de respuesta (muestreo-actualización).

3.2.3. Control Adaptativo

La opción de control que se prefirió utilizar fue el Control Adaptativo en el tiempo Discreto que por definición “Cambian el comportamiento del controlador conforme a nuevas circunstancias”, “Un controlador adaptativo es aquel que estima los parámetros del sistema y calcula la señal de control basándose en dichos parámetros” [20].

El esquema básico de control adaptativo está compuesto de dos lazos que evolucionan a diferente velocidad, uno es un bucle de retroalimentación ordinaria que evoluciona rápidamente, la otra que es más lenta corresponde a los cambios de los parámetros, es decir, a la velocidad con la que los parámetros del regulador son modificados.

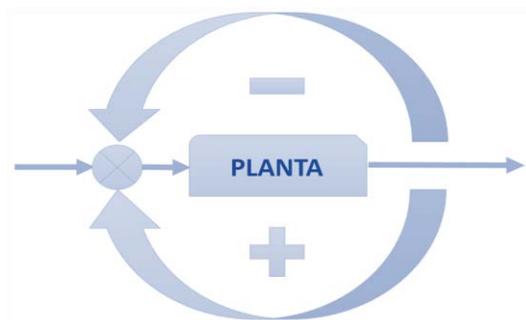


Figura 3.8: Esquema general del control adaptativo.

El esquema básico de control adaptativo está compuesto de un lazo principal de retroalimentación negativa, en el que actúa, al igual que en los sistemas convencionales un controlador, y de otro lazo en el que se mide un cierto índice de funcionamiento, el cual es comparado con el índice deseado y se procesa el error en un mecanismo de adaptación que ajusta los parámetros del controlador y actúa sobre la señal de control.

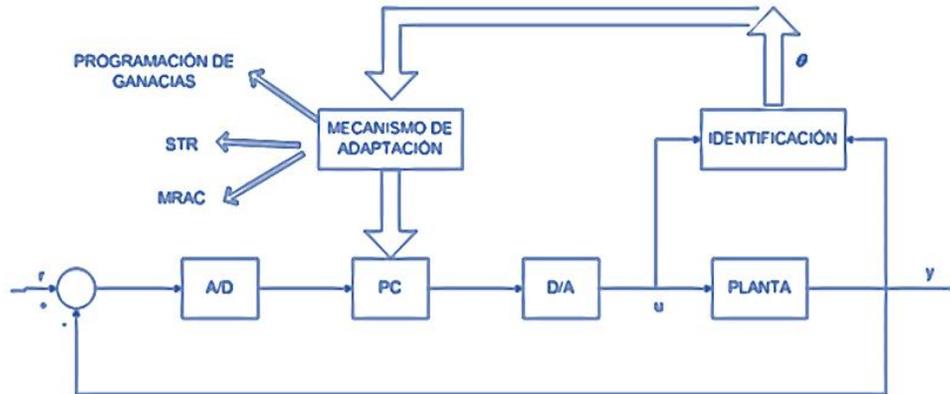


Figura 3.9: Control adaptativo con diferentes mecanismos de adaptación.

Existen diversos tipos de controladores que se denominan adaptativos como lo son por Modelo de Referencia (MRAC) o Auto-Sintonizable (STR), no obstante el que se empleara en este estudio será por “Ajuste de Tabla” o “Programación de Ganancia” (Gain Sheduling).

3.2.3.1 Control Adaptativo por programación de ganancias

En muchos casos es posible encontrar variables medibles que se relacionan bien con cambios en la dinámica del proceso, estas variables pueden ser utilizadas para cambiar los parámetros del controlador; este es el enfoque al que se denomina Programación por Ganancia porque el sistema fue originalmente utilizado para medir la ganancia y luego cambiar, es decir, establecer el controlador para compensar los cambios en la ganancia del proceso.

El sistema de control cambiará los parámetros en función de las condiciones de operación previamente programados, puede ser implementado como una función o mediante la búsqueda en una tabla.

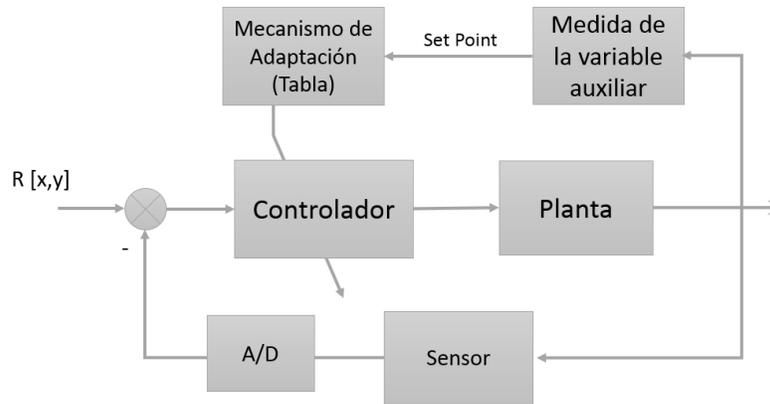


Figura.3.10. Esquema de control adaptativo por programación de ganancias.

El problema principal en el diseño de sistemas con ajuste por tabla es encontrar las variables auxiliares apropiadas para el cambio. Esto normalmente se hace basándose en el conocimiento físico del sistema. Cuando se determinan las variables auxiliares, los parámetros del controlador se calculan para distintos puntos de operación, usando algún método de diseño adecuado. El controlador es sintonizado o calibrado para cada punto de operación, se debe prestar atención a la transición entre diferentes puntos de operación. Sin embargo tiene la ventaja de que los parámetros del controlador pueden cambiar rápidamente en respuesta a la rapidez de los cambios que reflejen los parámetros del proceso [20].

3.3 PREDICCIÓN

La predicción de sucesos futuros ha fascinado a la humanidad desde tiempos muy antiguos, la predicción requiere de un arduo estudio de los sucesos del pasado y de las relaciones actuales, para que a partir de ello, intentar pronosticar un suceso futuro.

3.3.1. Horizonte de Predicción

El horizonte de predicciones es el límite práctico y válido para alguna predicción. En los sistemas complejos con gran cantidad de variables que cambian de forma compleja y errática, es imposible predecir su comportamiento más allá de un punto determinado; este punto es conocido como horizonte de predicción. Más allá de este punto los resultados son inexactos. (Edward Lorenz, 1960).

También se define como el periodo de tiempo (contabilizado desde un instante determinado en que realizamos la predicción) que determina el momento futuro para el que realizamos las predicciones.

Para esta aplicación en particular se debe considerar el tiempo en que alcanzara el proyectil al objetivo, esto a su vez está íntimamente determinado por la distancia que recorrerá.

Ahora si tomamos en cuenta que se utilizara un haz láser que viaja aproximadamente a la velocidad de la luz ($300 \times 10^6 \text{ m/s}$), se puede despreciar el tiempo de alcance, sin embargo ahora se debe considerar el retraso inherente que existe en la transmisión y procesamiento de datos para una predicción.

3.3.2. Filtro de Kalman

Se trata de un algoritmo de procesamiento de datos el cual estima los estados del sistema discreto de forma óptima y recursiva, de manera que minimiza el índice del error. Se denomina Óptimo porque minimiza un criterio determinado y porque incorpora toda la información que se le suministra para determinar el nuevo filtrado, y Recursivo porque no necesita mantener los datos previos, lo que facilita su implementación en tiempo real.

Covarianza

La covarianza indica la medida del grado en que dos variables aleatorias se mueven hacia la misma dirección o en direcciones opuestas, es decir, si dos variables aleatorias se mueven a la misma dirección se indicará que tiene covarianza positiva. Si tienden a moverse en direcciones opuestas se indicara que tienen covarianza negativa. De igual forma, la covarianza se define como el valor que se espera de la media aritmética de los productos de las desviaciones de cada una de las variables en relación a sus medias respectivas.

El algoritmo de Kalman se compone de tres partes fundamentales (un modelo y dos fases).

1. Modelo del sistema

Describe la evolución en el tiempo discreto de la cantidad que se desea estimar, esta cantidad es expresada mediante un vector de estado x_k . La transición entre estados ($x_k \rightarrow x_{k-1}$) se describe por la matriz de transición A_k , y la adición de ruido gaussiano blanco W_k que tiene una media cero en el proceso y está asociada a la matriz de covarianza Q formulado de la siguiente forma:

$$x_k = Ax_{k-1} + Bu_k + W_{k-1} \quad (3.1)$$

$$x_{k+1} = Ax_k + Bu_k + W_k \quad (3.2)$$

Donde:

x – Estado del sistema.

k – Tiempo.

u – Entrada del sistema.

W – Vector, Ruido en el proceso.

z – Valor de medición.

V – Vector, Ruido en la medición.

A – Matriz $n \times n$ describe la transición entre estados.

H – Matriz $n \times m$, relaciona la medición con el estado x .

B – Matriz $m \times 1$, relaciona la entrada de control con el estado x .

2. Modelo de la Medición

Relaciona el vector de medida z_k con el estado del sistema x_k a través de la matriz de medición H_k y la adición de un ruido gaussiano blanco V_k que tiene una media cero en la medición y está asociado a la matriz de covarianza R . Representado de la siguiente forma:

$$z_{k+1} = Hx_k + V_k \quad (3.3)$$

3. Modelo a Priori

Describe el conocimiento previo del vector de estado en el instante inicial $x(0)$, el valor esperado y su matriz de covarianza $P(0)$ respecto a las variables aleatorias W_k y V_k , se asume que son independientes entre ellas, que son ruido blanco y que tienen distribución de probabilidad normal.

En la práctica las matrices de covarianza de los ruidos de proceso Q y de la medición R , podrían cambiar con el paso del tiempo o la medición, sin embargo se asumen que son constantes. El ruido del sistema W_k puede considerarse que se genera en su interior o bien que se introduce en la entrada del sistema, y el ruido de medición V_k es el error que se comete al medir la salida (en los sensores) como se muestra en el diagrama.

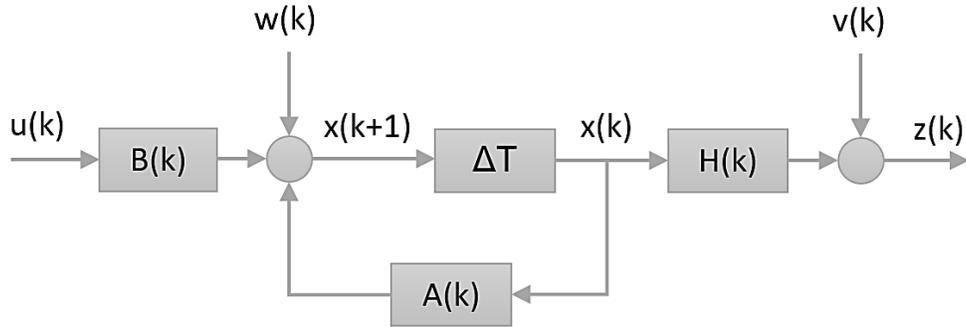


Figura 3.12: Representación del filtro de Kalman en diagrama a bloques.

Como se puede observar en el diagrama la dificultad radica en estimar el valor óptimo del vector de estado x_k considerando medidas ruidosas más la predicción del estado partir del conocimiento de su dinámica.

3.3.2.1. Principio computacional del Filtro de Kalman

El filtro de Kalman logra estimar a partir del proceso anterior valiéndose de una retroalimentación, es decir, estima el proceso en algún instante en el tiempo y luego obtiene la medición de la retroalimentación.

Se define $\hat{x}_{\bar{k}}$ como un estado a priori, estado declarado en el paso k a partir del conocimiento del proceso antes del paso k

Se define \hat{x}_k como el estado a posteriori, estimado en el paso k a partir de la medida z_k :

$$\text{Error a priori} \quad e_{\bar{k}} \equiv x_k - \hat{x}_{\bar{k}} \quad (3.4)$$

$$\text{Error posteriori} \quad e_k \equiv x_k - \hat{x}_k \quad (3.5)$$

$$\text{La covarianza a priori estimada del error es:} \quad P_{\bar{k}} = E[e_{\bar{k}} (e_{\bar{k}})^T] \quad (3.6)$$

$$\text{La covarianza a posteriori estimada del error es:} \quad P_k = E[e_k (e_k)^T] \quad (3.7)$$

Medida de innovación o residual, esta ecuación calcula un estado estimado a posteriori \hat{x} como una combinación lineal de un estado estimado a priori $\hat{x}_{\bar{k}}$ y una diferencia entre una medida real z_k y una medida de predicción $H\hat{x}_{\bar{k}}$.

$$\hat{x} = \hat{x}_{\bar{k}} + K(z_k - H\hat{x}_{\bar{k}}) \quad (3.8)$$

La matriz K de $n \times m$ es seleccionada para ser el factor de Ganancia de Kalman o el factor de mezcla que minimiza la ecuación de covarianza del error a posteriori.

$$K_k = P_{\bar{k}} H^T (H P_{\bar{k}} H^T + R)^{-1} \quad (3.9)$$

En esta ecuación se puede observar que conforme la covarianza del error en la medición R aproxima a cero, la medida actual z_k es más y más “confiable”, mientras que la medición predicha $H\hat{x}_{\bar{k}}$ es menos y menos “confiable”.

Por otro lado conforme la covarianza estimada de error a priori $P_{\bar{k}}$ se aproxima a cero la medida actual z_k es menos y menos “confiable” mientras que la medida predicha $H\hat{x}_{\bar{k}}$ es más y más “confiable” [21].

3.3.2.2. Algoritmo del Filtro de Kalman Discreto

El filtro de Kalman estima un proceso utilizando una continua retroalimentación, es decir, estima el proceso en algún instante en el tiempo y luego obtiene la retroalimentación por medio de los datos medidos, los cuales tienen ruido.

Las ecuaciones utilizadas se pueden dividir en dos grupos:

Las que actualizan el tiempo o de predicción, son las responsables de la predicción hacia delante (en el tiempo) del estado al instante k tomando como referencia el estado en el momento $k - 1$ y de la actualización intermedia de la matriz de covarianza del estado Q (error de covarianza estimado), para así obtener las estimaciones a priori para el siguiente paso en el tiempo $k + 1$.

Y las que actualizan los datos medidos o de corrección, son las responsables de la retroalimentación, es decir, incorporan una nueva medición dentro de la estimación a priori con lo cual se llega a una estimación mejorada del estado a posteriori.

Así el, el filtro de Kalman funciona por medio de un mecanismo de proyección y corrección al pronosticar el nuevo estado, su incertidumbre y corregir la proyección con la nueva medida.

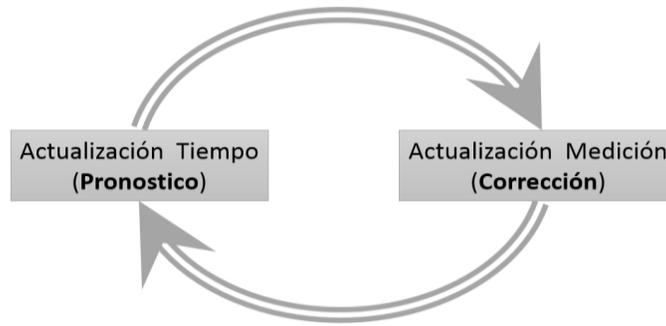


Figura 3.13: Recursividad del filtro de Kalman.

Ciclo del Filtro de Kalman

El primer paso consiste en generar un pronóstico del estado hacia delante en el tiempo tomando en cuenta toda la información disponible en ese momento.

El segundo paso, se genera un pronóstico mejorado del estado, de tal manera que el error es minimizado estadísticamente.

Las ecuaciones utilizadas para la actualización del tiempo o predicción son:

$$\hat{x}_{\bar{k}} = A\hat{x}_{k-1} + Bu_k \quad (3.10)$$

$$P_{\bar{k}} = AP_{k-1}A^T + Q \quad (3.11)$$

Se puede observar como las ecuaciones anteriores pronostican las estimaciones del estado y la covarianza desde $k - 1$ a k . La matriz A relaciona el estado en el momento previo $k - 1$ con el estado al momento actual k , esta matriz puede cambiar para los diferentes instantes en el tiempo k , Q representa la covarianza de la perturbación aleatoria del proceso que trata de estimar el estado.

Las ecuaciones utilizadas para la actualización de la medida o corrección son:

$$K_k = P_{\bar{k}}H^T(HP_{\bar{k}}H^T + R)^{-1} \quad (3.12)$$

$$\hat{x} = \hat{x}_{\bar{k}} + K(z_k - H\hat{x}_{\bar{k}}) \quad (3.13)$$

$$P_k = (I - K_kH)P_{\bar{k}} \quad (3.14)$$

La primera tarea durante la corrección de la proyección del estado es el cálculo de la ganancia de Kalman K_k , este factor de ganancia es seleccionado de tal forma que minimice la covarianza del error de la nueva estimación del estado.

El siguiente paso es tomar una medición del proceso en el instante k y obtener el valor de z_k , esta medida actualizada del proceso permite obtener una estimación a posteriori de estado.

El paso final se calcula la estimación a posteriori de la matriz de covarianza del error.

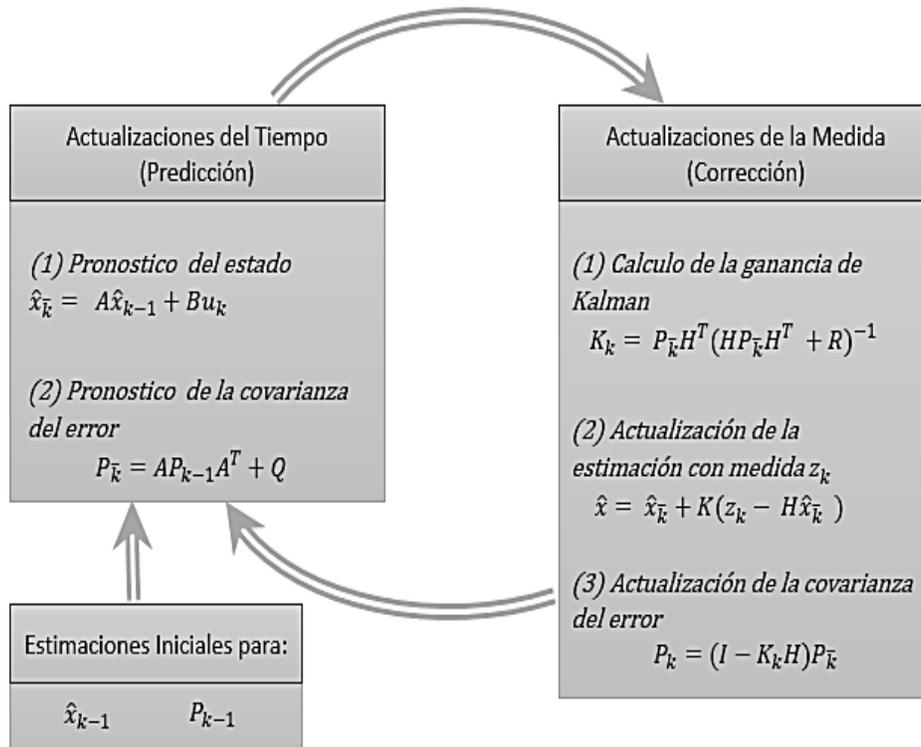


Figura 3.14: Esquema Discreto del filtro de Kalman.

Después de cada par de actualizaciones, (Predicción y Corrección), el proceso es repetido tomando como punto de partida las nuevas estimaciones del estado y de la covarianza del error. Esta naturaleza recursiva es una de las características atractivas del filtro de Kalman, y lo hace viable para implementaciones prácticas. El filtro de Kalman condiciona recurrentemente la estimación actual en todas las últimas medidas.

3.3.3. Filtro de Kalman en Imágenes

Para hacer uso del filtro de Kalman en imágenes se debe conocer las coordenadas cartesianas dentro de la imagen donde se encuentra el Centroide del objeto de interés $p_0 = (x_0, y_0)$, estas dos variables determinan el orden del sistema, como se puede observar el sistema es de cuarto orden, considerando como variables de estado la posición y velocidad relativa de cada coordenada.

En el caso de la velocidad, una vez que se obtiene a posición del objeto, puede ser calculada como:

$$v = \frac{p}{t} \left[\frac{\text{pixeles}}{\text{frame}} \right] \quad (3.15)$$

Donde p indica los pixeles que se desplazó el objeto.

Para encontrar la velocidad es necesario encontrar la siguiente posición del objeto $p_1 = (x_1, y_1)$, y con una resta obtenemos el desplazamiento en pixeles del objeto, obteniendo así las condiciones iniciales del filtro de Kalman.

Utilizando un modelo lineal del sistema. Los estados están definidos por:

$$X(k) = \begin{bmatrix} x(k) \\ y(k) \\ v_x(k) \\ v_y(k) \end{bmatrix} \quad (3.16)$$

Donde:

$x(k)$ Y $y(k)$ representan la posición en el instante k .

$v_x(k)$ Y $v_y(k)$ representan la velocidad en el instante k .

Las ecuaciones del sistema estarán dadas por:

$$x(k) = x(k - 1) + v_x(k - 1) \quad (3.17)$$

$$y(k) = y(k - 1) + v_y(k - 1) \quad (3.18)$$

$$v_x(k) = v_x(k - 1) \quad (3.19)$$

$$v_y(k) = v_y(k - 1) \quad (3.20)$$

En un instante k la posición es igual a la posición en el instante $(k - 1)$ más un desplazamiento (velocidad por tiempo, con $t = 1$).

El modelo del sistema está definido por:

$$\begin{bmatrix} x(k) \\ y(k) \\ v_x(k) \\ v_y(k) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x(k - 1) \\ y(k - 1) \\ v_x(k - 1) \\ v_y(k - 1) \end{bmatrix} + \begin{bmatrix} u_1(k - 1) \\ u_2(k - 1) \\ u_3(k - 1) \\ u_4(k - 1) \end{bmatrix} \quad (3.21)$$

La matriz de covarianza $Q(k)$ que modela el ruido en el proceso:

$$Q(k) = \begin{bmatrix} \sigma_Q^2 & 0 & 0 & 0 \\ 0 & \sigma_Q^2 & 0 & 0 \\ 0 & 0 & \sigma_Q^2 & 0 \\ 0 & 0 & 0 & \sigma_Q^2 \end{bmatrix} \quad (3.22)$$

Donde: σ_Q^2 es un parámetro.

La medida está dada por:

$$z(k) = \begin{bmatrix} z_x(k) \\ z_y(k) \end{bmatrix} \quad (3.23)$$

Donde:

$$z_x(k) = x(k) \quad (3.24)$$

$$z_y(k) = y(k) \quad (3.25)$$

EL modelo de mediciones está definida como:

$$\begin{bmatrix} z_x(k) \\ z_y(k) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x(k) \\ y(k) \\ v_x(k) \\ v_y(k) \end{bmatrix} + \begin{bmatrix} v_1(k) \\ v_2(k) \end{bmatrix} \quad (3.26)$$

La matriz de covarianza $R(k)$ que modela el ruido en la medición:

$$R(k) = \begin{bmatrix} \sigma_R^2 & 0 \\ 0 & \sigma_R^2 \end{bmatrix} \quad (3.27)$$

Donde: σ_R^2 es un parámetro.

3.3.3.1. Parámetros y sintonización

El mayor problema para implementa este filtro es la falta de información, pues al inicio los parámetros del modelo son desconocidos. No obstante el modelo se asume lineal si el tiempo de muestreo es suficientemente rápido. En esta situación el filtro comienza a predecir en base a estados unitarios, continúa así hasta que empieza a corregirse con la toma de mediciones.

El modelo de medida está disponible desde que se entrega las ubicaciones (coordenadas) del objeto.

Como se expuso la ganancia del filtro de Kalman K_k es el peso relativo de la predicción del modelo y de las medidas, en su ecuación se observa que depende de las matrices de covarianza del ruido del sistema $Q(k)$ y de la medida $R(k)$.

Estos parámetros (Q y R) pueden ser obtenidos fuera del sistema.

Los parámetros de la matriz R se pueden encontrar de manera práctica mediante la toma de muestras de mediciones buscando la varianza en el ruido o perturbación, también se pueden obtener por prueba y error buscando hallar el valor ideal para el proceso, la prueba y error se realiza observando la ecuación de la ganancia de Kalman.

Si las perturbaciones en las mediciones son grandes entonces R será grande y en consecuencia la ganancia será pequeña, lo que indica que la medición en el cálculo siguiente \hat{x} no tendrá mucho peso y no será muy confiable.

Por el contrario si R tiene asignado un valor pequeño, se considera que las perturbaciones en la medición son pequeñas y en consecuencia la ganancia será grande y estará dando mayor peso o credibilidad en las mediciones cuando se calcule el siguiente \hat{x} .

Determinar el parámetro Q que representa el ruido en el proceso es complejo y por lo general es difícil de conocer, una forma de establecerlo es hacer una correspondencia con la covarianza R , que dependerá del tipo de sistema a implementar.

Si los parámetros de R son más pequeñas que los de Q , significa que el sistema es muy ruidoso y genera más incertidumbre que las medidas, resultando que el filtro de Kalman ignore la predicción del modelo y confié casi exclusivamente en las medidas.

Contrariamente, si los parámetros de Q son más pequeños que R , se expresa que las medidas poseen mayor incertidumbre que la predicción, resultando que el filtro ignore las medidas y confié casi enteramente en la predicción del modelo del sistema [22].

3.4 ANGULO DE ADELANTO Y UBICACIÓN ESPACIAL

Este parámetro está íntimamente relacionado con la posición del objetivo, velocidad, condiciones atmosféricas y el tiempo de vuelo del proyectil (ToF).

Las condiciones para calcular el ángulo de adelanto son:

- a. El objetivo debe seguir una línea recta y no debe acelerar durante el tiempo en que vuela el proyectil (ToF).
- b. La distancia del objetivo respecto a la cámara no debe cambiar durante el (ToF).
- c. Para cortas distancias las condiciones atmosféricas pueden ser ignoradas.

Estas condiciones son aceptadas por el sistema debido a que se predice en mínimos tiempos futuros, es decir, la aceleración y la distancia no pueden cambiar drásticamente, no más rápido que el (ToF) de la incidencia del haz láser.

Ahora sabiendo que el sistema no opera totalmente en tiempo real, existe un pequeño retraso en la transmisión de datos, será necesario calcular el ángulo de adelanto, para algunos cortos instantes futuros.

Considerando la posición del objeto en coordenadas esféricas son: (ρ, θ, φ) , donde (ρ) es la distancia al objetivo, (θ) es el ángulo horizontal y (φ) es el ángulo vertical.

Y además sabiendo que el algoritmo de predicción (Filtro de Kalman) provee (x, y) y de igual forma la triangulación laser proporciona (ρ) , con los que se puede calcular (θ, φ) se podrán convertir a coordenadas cartesianas con las siguientes ecuaciones:

$$\begin{aligned} X_1 &= \rho_1 \text{ sen } \varphi_1 \cos \theta_1 \\ Y_1 &= \rho_1 \text{ sen } \varphi_1 \text{ sen } \theta_1 \\ Z_1 &= \rho_1 \cos \theta_1 \end{aligned} \quad (3.28)$$

Estas coordenadas sirven para ubicar la posición del cañón respecto a las coordenadas de la cámara, donde las coordenadas predichas se compensan a consecuencia de la distancia entre ambos dispositivos (ΔD) , suponiendo que se encuentren nivelados, es decir, que no exista variación en z .

$$P(x + \Delta x, y + \Delta y, z) \quad (3.29)$$

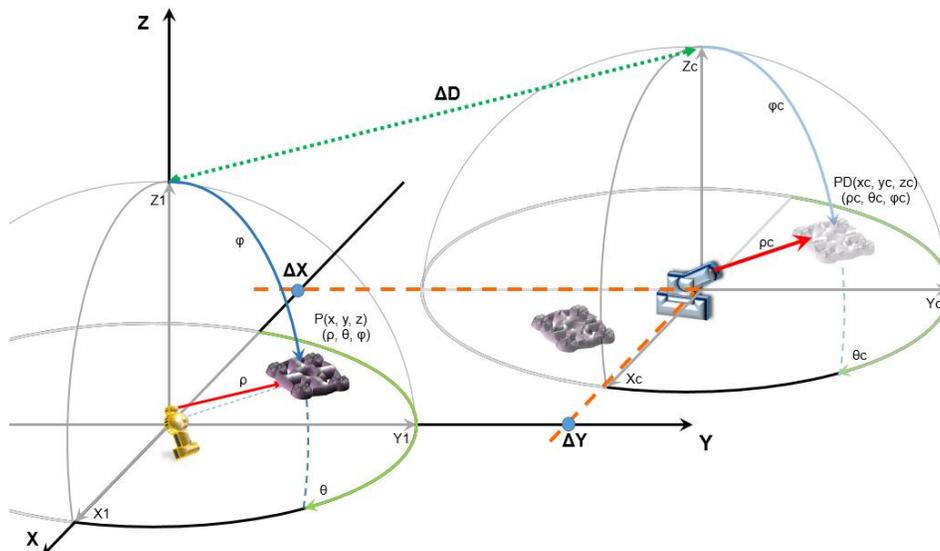


Figura 3.15: Ubicación espacial entre el sistema de seguimiento y el de derribo.

3.4.1. Ubicación espacial (Bosquejo)

En esta etapa es donde todo lo calculado cobra significado, ya que conociendo la ubicación espacial del objeto es posible realizar la proyección de un plano ortogonal a la ubicación actual y a partir de esta imagen “proyectada” se puede calcular las dimensiones reales del objeto así como la distancia máxima de percepción en (x, y) , al mismo tiempo conociendo su futura posición en el plano de la imagen “proyectada”, es posible predecir la ubicación real espacial del objeto $P(x_f, y_f, z_f)$, así como la velocidad instantánea que posee.

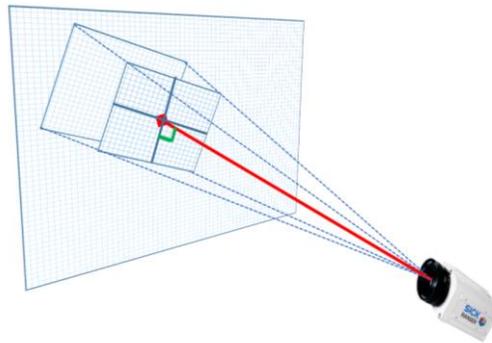


Figura 3.16: Proyección ortogonal de la imagen respecto al haz láser.

Se dice que es ortogonal por las características de la triangulación láser donde el cateto opuesto determina la proyección perpendicular a esa distancia, la proyección que se realiza es plana, es decir, no tiene efectos de distorsión geométrica debido a la incidencia del haz en el centro de la imagen (donde la distorsión es mínima).

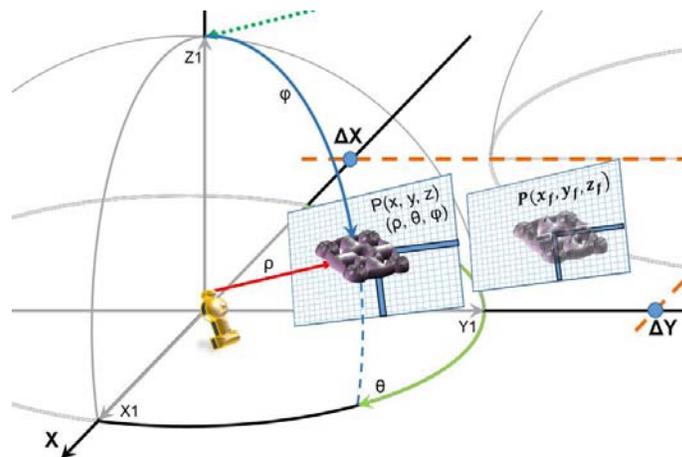


Figura 3.17: Esquema de predicción de movimiento espacial.

Estas coordenadas cartesianas espaciales obtenidas se deben convertir a coordenadas esféricas nuevamente, debido a que el sistema de derribo opera describiendo una esfera y es aquí donde se calcula el ángulo de adelanto que debe poseer el sistema.

El ángulo de adelanto se calcula en base a los datos arrojados de la predicción espacial para hacer efectivo el derribo, y se utilizan en ambos actuadores del sistema de derribo.

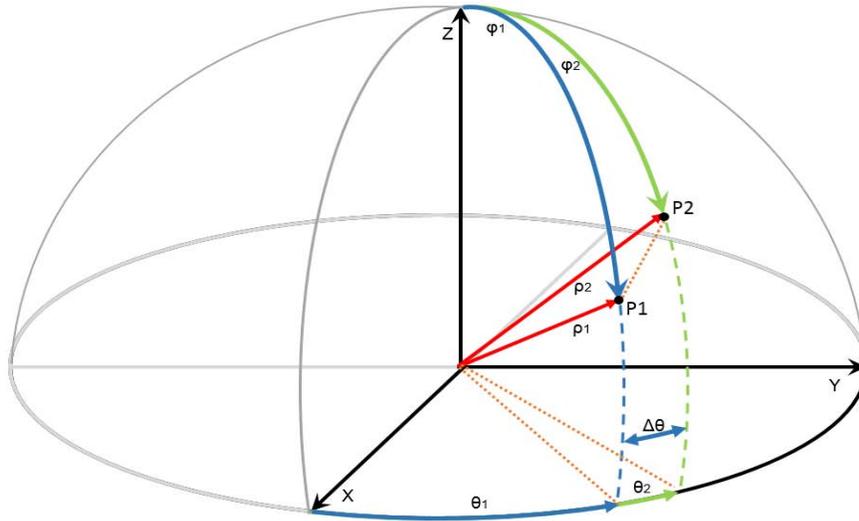


Figura 3.18: Representación de los puntos para el análisis del ángulo de adelanto.

3.4.2. CALCULO ÁNGULO DE ADELANTO

Si bien, conociendo que el (ToF) es despreciable ya que el haz laser viaja casi a la velocidad de la luz $300 \times 10^6 \text{ m/s}$, se tomara en cuenta el retraso que existe en la trasmisión de datos, así como la velocidad del objeto v_0 .

Asumiendo que la ubicación próxima es semejante en t tiempos futuros cortos se puede considerar que:

$$\rho_1 = \rho_2 = \rho \quad \text{y} \quad \varphi_1 = \varphi_2 = \varphi \quad (3.30)$$

Se obtiene con la siguiente ecuación el ángulo de adelanto en theta donde tendrá que apuntar el sistema de derribo [23]. El procedimiento es similar para calcular phi adelantado.

$$\Delta\theta = \cos^{-1} \left(1 - \frac{v_0^2 ToF^2}{2\rho^2 \text{sen}^2\varphi} \right) \quad (3.31)$$

CAPÍTULO 4

PROPUESTA DEL SISTEMA DE DERRIBO

4.0 INTRODUCCIÓN

En este capítulo se propondrá el sistema de derribo el cual controlará el tránsito del espacio aéreo o terrestre en determinada zona.

En un principio se pensó en adaptar el cañón de derribo en el mismo sistema de seguimiento, pero con las posibles vibraciones que tendría la expulsión del proyectil, se descartó inmediatamente, debido a que la cámara en un movimiento brusco perdería el objetivo. Posteriormente se descartaron los sistemas de expulsión de proyectil por razones que se explicaran en el siguiente apartado, así que se decidió usar láser. El cual podría ser acoplado mecánicamente al sistema de seguimiento sin que provoque vibraciones y en consecuencia la pérdida visual del objeto.

No obstante se propone implementar el sistema independiente del seguidor debido a que se piensa utilizar la metodología de "robots cooperativos", la cual consistirá en tener no solo un sistema de derribo, sino varios situados de manera estratégica que actuando de manera colaborativa tendrán mayor éxito en el derribo del objetivo, siguiendo el esquema de control centralizado.

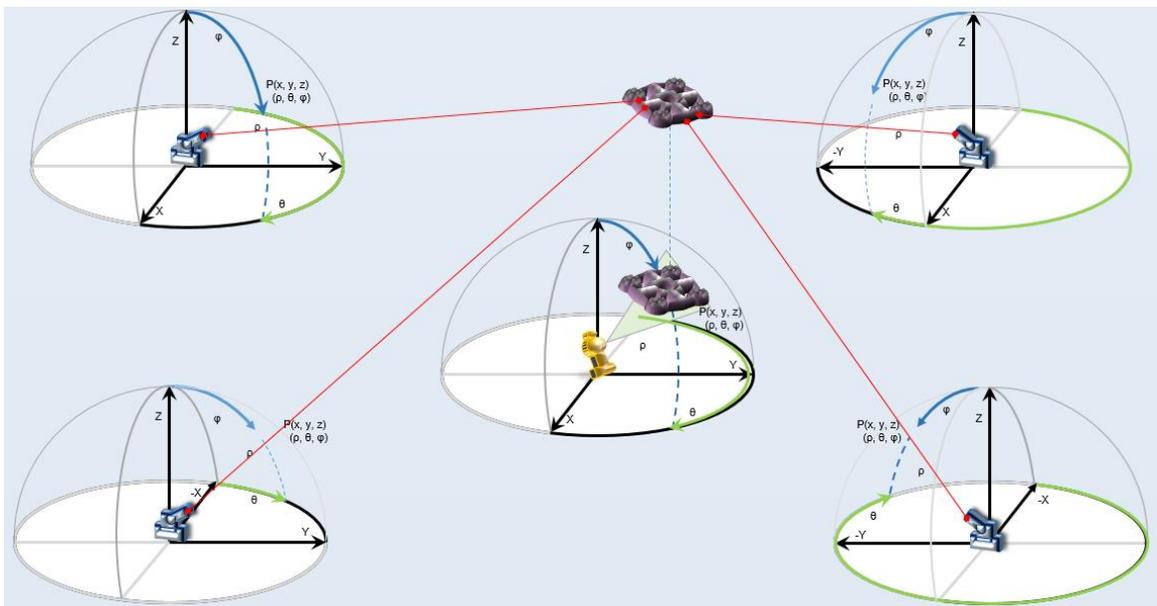


Figura 4.1: Sistema propuesto para un eficaz derribo aéreo.

El sistema de derribo contará con similar sistema de actuación al de seguimiento, es decir, tendrá los mismos grados de libertad y se mantendrá en todo momento comunicado con el sistema de seguimiento para actualizar el sentido de movimiento así como la velocidad rotacional en sus actuadores.

4.1 ESTADO DEL ARTE

Los sistemas de derribo aéreo son de carácter militar y durante el transcurso de la historia han evolucionado junto con la tecnología.

Los que aun predominan son los de tiro parabólico y proyectil autopropulsado, pero recientemente se han empezado a desarrollar los que emplean ondas electromagnéticas como el láser; cada sistema presenta características particulares y por lo tanto retos específicos en su control.

4.1.1 Tiro Parabólico

El tiro parabólico es una posible solución y se desprende de ella bastantes tópicos de estudio, pero presenta diversas dificultades que lo hacen difícil de implementar con éxito, estos problemas pueden dividirse en internos y externos.

Problemas Internos

Son las características que tomamos en cuenta para obtener una óptima dirección y velocidad de disparo. Estas van desde la correcta elección del tipo de proyectil considerando su aerodinámica y material de construcción, también está involucrado el cañón de disparo, su longitud, anchura y el estriado interno del cañón que dará estabilidad al vuelo, pero sobre todo el agente que dará la potencia de salida y determinara su máxima velocidad definiendo el rango de distancia efectiva [24].

Problemas Externos

Aquí consideramos as condiciones que enfrentará el proyectil al hacer contacto con el ambiente, las cuales modificaran su trayectoria, estas variables son, la gravedad, la ventisca que mermara la distancia de vuelo, densidad del aire que creará resistencia al vuelo, curvatura y rotación de la tierra que solo son consideradas en distancias grandes [24].

4.1.2 Proyectil autopropulsado

El proyectil autopropulsado (misil) es todo un problema de implantación debido a la naturaleza no lineal de su control, el desarrollo de un sistema de autopropulsión y además se descarta por la naturaleza de los Drones, los cuales no generan el suficiente calor para ser detectados térmicamente, la mayoría son fabricados de materiales polímeros lo que lo hacen indetectables para radares, por lo tanto no hay forma de guiarlos hasta su objetivo.

4.1.3 Armas de energía dirigida (DEW)

Los programas dirigidos a la utilización de la tecnología láser en el armamento comenzaron a tomar forma en la primera década después de los primeros éxitos de la tecnología láser en los años '60, y avanzó enormemente con la llegada de los láseres gaseosos hacia 1975, así como los láseres bombeados por impulsos.

Las tecnologías de armas electromagnéticas representan una clase de aplicaciones de alto nivel basadas en efectos que producen la energía electromagnética concentrada, estos sistemas se ejemplifican mediante aplicaciones tales como: láseres de alta energía (High-Energy Lasers = **HEL**), dispositivos de microondas de alta potencia (High-Microwave devices = **HPM**) y municiones de radio frecuencia (Radio Frequency Munitions = **RFM**) [27].

El uso del láser como técnica de derribo se ha desarrollado con éxito en otros países, es una técnica reciente y novedosa, ya que no se rige por los problemas anteriormente explicados. Esta técnica no utiliza un objeto "físico" para colisionar el objetivo, más bien utiliza la concentración de energía.

Y que por ley física todo fotón que incide en la materia aumenta la vibración de sus moléculas en la superficie que toca, es decir, genera un aumento de la temperatura.

4.1.4 Sistemas Análogos

- Telémetros Láser.

Son sistemas que miden el tiempo que tarda un impulso láser en llegar a un objeto/objetivo, ser reflejado por el mismo y volver a su punto original; con el objetivo de obtener las distancias a blancos terrestres o aéreos. Estos se encuentran montados externamente sobre vehículos de combate blindados o carros de combate; tales como el MOLEN de ZEISS Optronik (Alemania). [27]

- Sistema Láser Aerotransportado (ABL)

Se trata de un sistema de un sistema antimisiles láser dentro de un Boeing 747 (Airborne Laser = ABL), utiliza un control de haz de baja potencia y un sistema de dirección de tiro para seguir y priorizar objetivos, así como de un láser táctico de alta potencia para destruirlos. La mitad trasera del avión alberga el láser de alta energía, mientras que la mitad delantera contiene el control de haz/sistema de dirección de tiro. [27]

- SKY GUARD

Es un sistema desarrollado por Northrop Grumman para combatir misiles antiaéreos, cohetes y munición de artillería, utilizando un láser de alta potencia (Tactical High Energy Laser = THEL),

pensado para proteger en aeropuertos el arribo y despegue de aviones, su costo se encuentra alrededor de 30 millones de dólares. [28]

4.2 ABLACIÓN TÉRMICA LÁSER

La ablación térmica surge de la interacción física láser-materia, en esta interacción una parte de la energía es reflejada y otra parte es absorbida, esto depende de la longitud de onda en la radiación láser, la densidad de potencia, la duración del pulso y sobre todo las propiedades de la superficie.

Cuando la superficie logra absorber energía del haz láser en función de la longitud de onda se producen los siguientes efectos:

- Efectos térmicos.
Están asociados a un incremento de la temperatura debido a la absorción de la energía láser, para conseguir ese incremento térmico se utilizan láser de pulso largo o continuo de tal manera que dependiendo de la duración y la intensidad de los mismos, se tienen distintos efectos macroscópicos que dependiendo del material van desde vaporización, carbonización y fusión [25].
- Efectos Fotoquímicos
Se produce la ruptura de los enlaces entre las moléculas del material. Puede lograrse al utilizar láseres ultravioleta cuyos fotones son muy energéticos, suficientes para romper el enlace químico de la molécula del material [25].

Ahora considerando la duración del pulso láser se describen los casos:

- Pulso Largo.
La radiación láser produce una absorción óptica inicial, acompañada de un rápido aumento de la temperatura y de la presión sobre la superficie, seguida de una expansión de gas.
- Pulso Corto.
En este caso se experimenta una ablación fotomecánica y la mayor cantidad de calor transferida al material se va con la fracción de materia extraída.

4.2.1. Ventajas del láser

Velocidad. La velocidad que tiene para hacer contacto con el objeto es comparable a la luz, sin embargo para ocasionar cambios significativos en el material de contacto debe concentrar energía por un tiempo determinado.

Material de desecho. La ablación láser genera relativamente menos desechos, solo el objeto que debe ser eliminado, en comparación con otros sistemas.

4.2.2. Mecanismo del proceso

El primer paso para producir un efecto duradero sobre una superficie, es la absorción de la radiación láser. La forma en la que un rayo láser interactúa con un material depende de parámetros del haz tales como la longitud de onda y la potencia, y las propiedades físicas y químicas del material. Cuando un haz láser encuentra un obstáculo en su viaje parte de la energía es reflejada (reflectividad) y el resto es absorbido (capacidad de absorción), suponiendo que no existe transmisión a través del material.

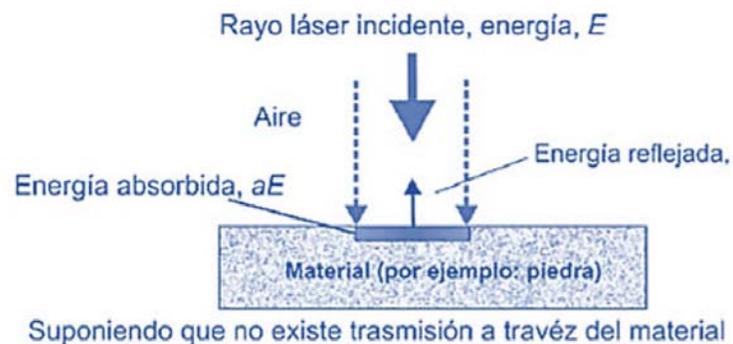


Figura 4.2: Forma en que incide el haz láser

La absorción depende de dos variables principales:

- Las propiedades del material.
- La longitud de onda de la radiación láser.

La capacidad de absorción también depende de las irregularidades de la superficie. Una superficie irregular, poco reflectante, absorbe una mayor proporción de la energía que una superficie uniforme del mismo material, porque presenta una zona de superficie mayor a incidir pues tiene mayores ángulos de incidencia.

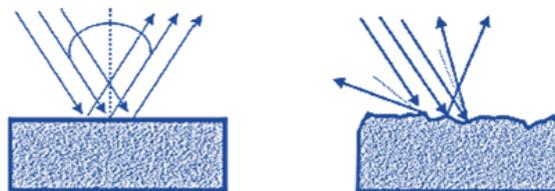


Figura 4.3: Variaciones en las superficies de incidencia

4.2.3. Efectos producidos por la absorción de radiación láser

Existe una relación entre la longitud de onda y la energía de un fotón, conforme la longitud de onda se reduce, la energía del fotón aumenta. Concluyendo que los fotones ultravioletas con longitud de onda corta son más energéticos que los infrarrojos con longitud de onda más larga, pasando por los espectros visibles del ojo humano.

Las diferencias entre energías y longitudes de onda producen diversos efectos en la superficie del material. A longitudes de ondas cortas, puede producir ruptura molecular sin generar calor. A longitudes de onda más largas (en el espectro visible e infrarrojo) las energías de los fotones son demasiado bajas para romper enlaces directamente y la energía se convierte en calor.

Haciendo coincidir la longitud de onda y la potencia del haz con las propiedades ópticas y térmicas del material, se puede controlar el aumento de temperatura para conseguir un resultado específico.

Considerando una duración prolongada del haz sobre la superficie, si aumentamos la densidad de potencia y con el consecuente aumento de temperatura, la superficie conducirá finalmente a cambios de fase como fusión y vaporización [25].

Si se eleva la temperatura por encima del punto de fusión, la región irradiada de la superficie se fundirá y después se re solidificará, una vez que el pulso láser haya concluido.

Si aumentamos la densidad de potencia, se producirá vaporización tan pronto como el punto de ebullición de la superficie sea excedido. La vaporización inducida por pulsos cortos se ha descrito como "Explosión Térmica" [26]; la ablación de material desde una superficie genera una presión de retroceso sobre dicha superficie.

Para conseguir un efecto duradero sobre la superficie, la potencia del haz debe ser suficientemente alta para que la energía absorbida sea la necesaria y producir así un cambio en el material.

CAPÍTULO 5

IMPLEMENTACIÓN Y RESULTADOS

5.0 INTRODUCCIÓN

El algoritmo general se muestra en el anexo, a continuación se muestra de manera simplificada: se encuentra dividido en tres etapas, cada una con funciones diferentes y dependientes uno de otro para conseguir el objetivo final.

La primera parte **(A)** consta de la detección y seguimiento en la imagen del objeto, para lograr la detección es necesario que el sistema este constantemente escaneando los posibles lugares donde aparecerá el objeto, el objeto se puede detectar cuando atraviesa el campo de visión de la cámara ya sea por el movimiento o por información a priori de su modelo a seguir. Una vez detectado se utilizarán dos métodos para su seguimiento.

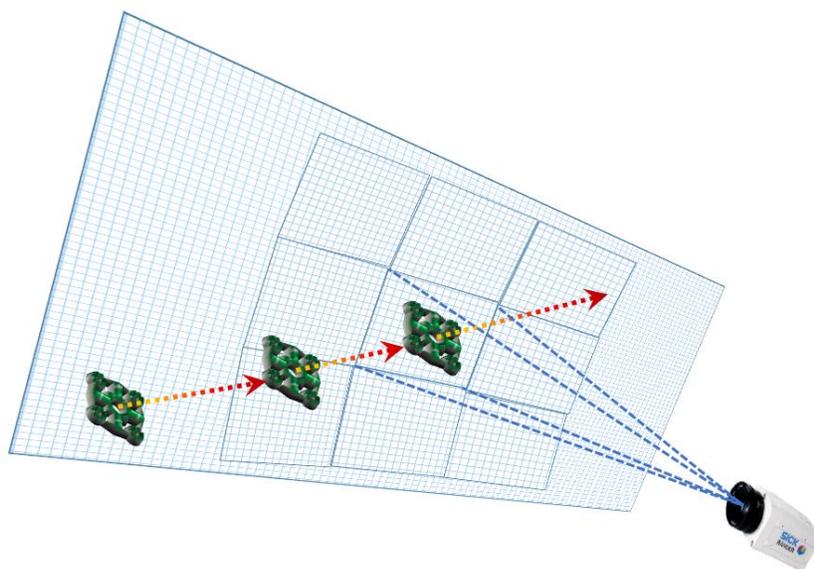


Figura 5.1: Representación de la detección y seguimiento de objetos.

La segunda parte **(B)** consiste en extraer las coordenadas del objeto seguido y en base a ellas se controlara la actuación mecánica del sistema, es decir se decidirá la velocidad rotacional de ambos actuadores así como su sentido de giro, con el fin de mantener el objeto centrado y poder obtener su distancia respecto a la cámara, una vez adquiridos estos valores es posible realizar una predicción del movimiento con el fin de posicionar de manera eficiente el sistema de derribo.

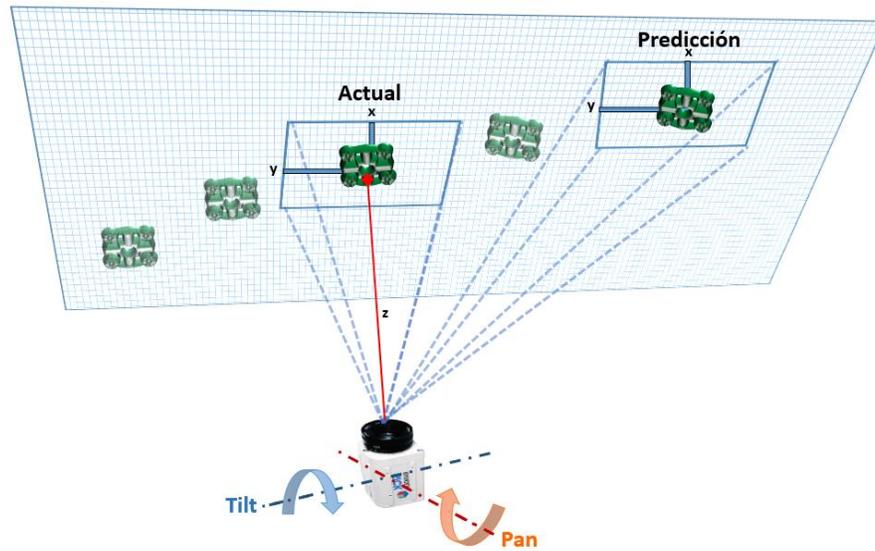


Figura 5.2: Representación de la forma en que mide la lejanía y realiza la predicción.

La tercera parte **(C)** consiste en el sistema de derribo, como sea mencionado anteriormente, no se realizara físicamente, solo se propondrá un posible sistema de derribo, este sistema continuamente estará actualizando la posición angular de ambos actuadores y accionando el haz láser hasta lograr el derribo del objeto o conseguir alejarlo de la zona restringida a su vuelo.

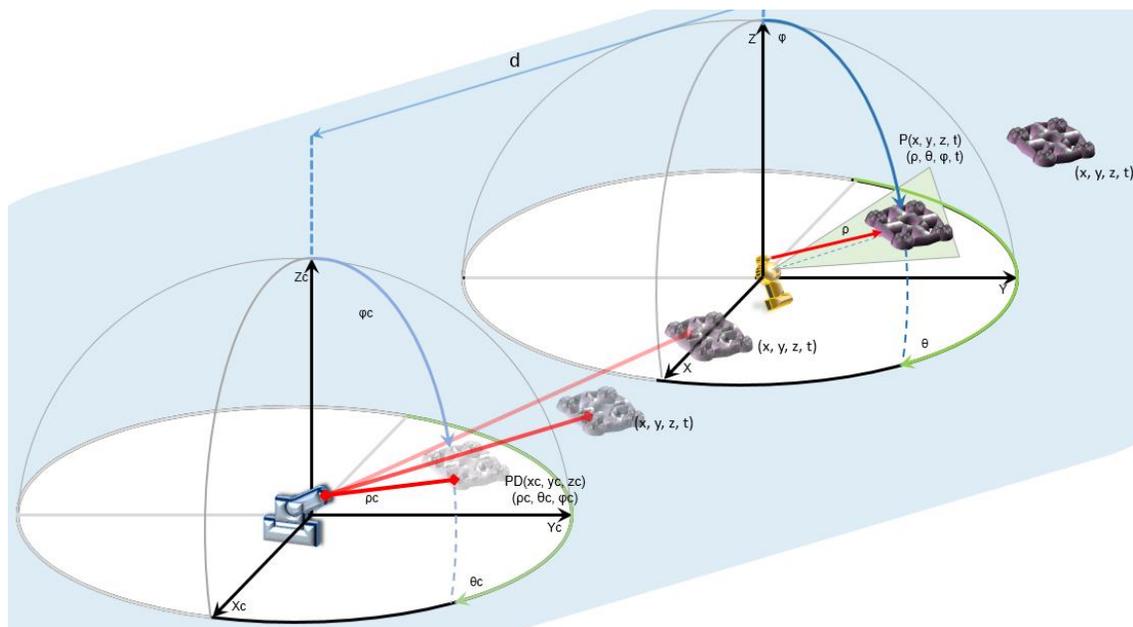


Figura 5.3: Propuesta de sistema de derribo.

En los siguientes apartados se describirá a detalle el funcionamiento de cada etapa.

5.1 DISEÑO DE SOFTWARE

Como se mencionó inicialmente el software utilizado es Matlab, es un software muy potente debido a que se especializa en el manejo de matrices y conociendo que las imágenes son matrices es excelente para la implantación, además cuenta con funciones predefinidas que facilitan la comunicación con dispositivos externos; de igual forma se emplearon funciones de los toolbox de procesamiento de imágenes y de visión artificial.

Se diseñó una interfaz gráfica de usuario (GUI), para ejecutar el algoritmo general paso a paso y facilitar su uso. Consta de dos ventanas, la primera es la principal, y en ella se visualizara en todo momento los frames capturados, la segunda es la auxiliar y servirá para visualizar las detecciones de movimiento y de re-detección.

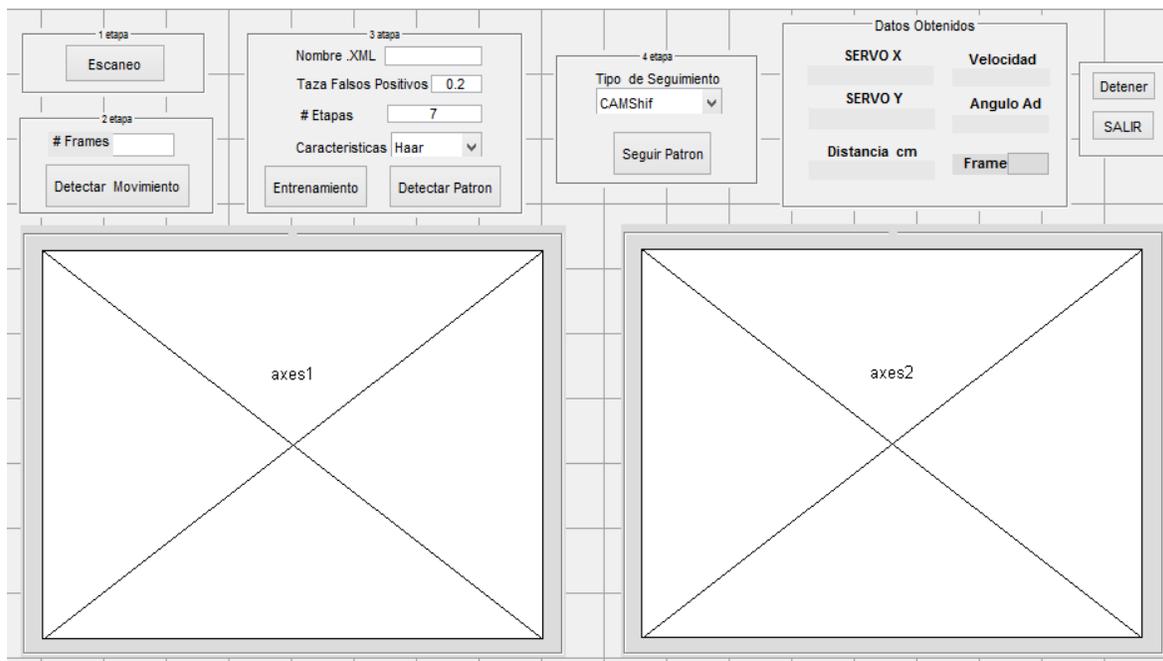


Figura 5.4: Interfaz gráfica de usuario.

La interfaz está dividida en etapas las cuales producen el algoritmo general. Al momento de abrir la interfaz inicializa la Comunicación Serial y gestiona la adquisición de imágenes, por tal proceder es necesario tener previamente conectados los dispositivos a la PC.

Se visualiza en ella la cantidad de frames que se están capturando. Se diseñó con botones esenciales de control de ejecución: el botón “Detener”, como su nombre lo indica detiene la ejecución de los algoritmos de detección y de seguimiento, el botón de “Salir”, aborta todas las operaciones, cierra la comunicación Serial y Limpia el búfer de la cámara, finalmente cierra el programa.

La primera fase cuenta con un botón el cual está programado para realizar el escaneo automático del entorno almacenando en una carpeta imágenes negativas (donde no hay objeto).

La segunda etapa cuenta con un botón el cual iniciará la detección de movimiento, así mismo se puede configurar la cantidad de frames que detectaran el objeto en movimiento, con el fin de crear en otra carpeta imágenes positivas (donde está el objeto).

La tercera etapa consta de dos pasos, el primero entrena automáticamente un detector cascada con las imágenes anteriormente obtenidas, a este entrenamiento se le puede configurar para ser más efectiva la detección. El segundo paso comprueba correcto el funcionamiento del entrenamiento, sino se satisface se puede corregir el entrenamiento.

La cuarta etapa tiene programada el seguimiento del objeto, en esta se puede elegir el tipo de seguimiento que más convenga, así mismo tiene programado la predicción del movimiento, la elección del sentido y velocidad para los actuadores, también el procedimiento para calcular la distancia y el ángulo de adelanto.

Finalmente tiene un apartado donde se visualiza los valores obtenidos durante el seguimiento.

La resolución que se empleó para la captura de imágenes es relativamente baja de 352x288 pixeles, sin embargo es suficiente para lograr los objetivos planteados, la cámara se ajusta a la cantidad de luz en el ambiente, por tal motivo, la velocidad de toma de captura (fps) varía dependiendo la exposición a la luz que tenga, a baja luminosidad la cámara trata de extraer mayor cantidad de fotones, es decir, tarda más en cargar los sensores matriciales y en consecuencia la velocidad disminuye, a una gran exposición de luz la velocidad aumenta alrededor de 15fps.

PRIMERA PARTE

5.1.1. Escaneo del Entorno

La cámara que se utilizó es de apertura angular, es decir, sólo detectan un fragmento de un entorno, existen otras cámaras llamadas panorámicas las cuales logran detectar 180 grados de un entorno su desventaja está en la deformación geométrica que hace al tomar la imagen.

La cámara de apertura angular solo percibe una porción del entorno, por tal razón es necesario efectuar una exploración del entorno donde posiblemente aparezca el objeto, conociendo que solo se detectará por primera vez el objeto en movimiento, es decir, cuando están en pleno vuelo, reducirá la búsqueda a solo los cuadrantes positivos de z en coordenadas cartesianas o el primer cuadrante positivo de φ en coordenadas esféricas.

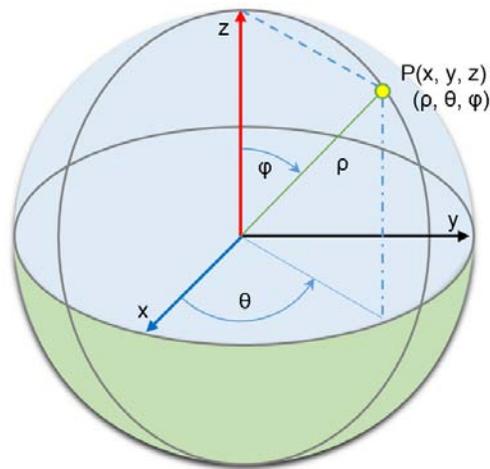


Figura 5.5: Cuadrantes de escaneo φ y z

La cámara tiene ángulos de apertura de aproximadamente (25x25) grados, por tal circunstancia se explorará a través secciones con el fin de cubrir todo el espacio visible.

El movimiento en Pan (x o θ) lo realizará el actuador utilizando la totalidad de su movimiento (180 grados), los cuales se pueden cubrir en ocho secciones de 22.5 grados, consiguiendo redundancias en las observaciones.

El movimiento en Tilt (y o φ) requiere una rotación de solo 90 grados, los cuales se pueden cubrir en cuatro secciones de 22.5 grados, consiguiendo redundancias en las observaciones.

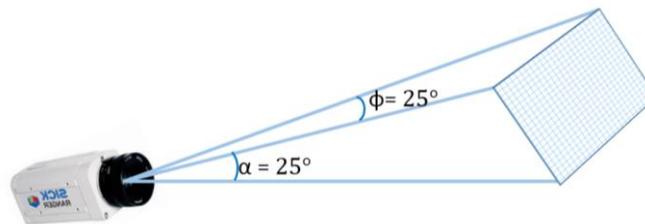


Figura 5.6: Ángulos de apertura internos en la cámara

Como se muestra en la proyección (cuadro punteado) habrá capturas que abarcará fragmentos de otras secciones, esto se debe a la naturaleza esférica del sistema, esta consecuencia más que afectar, ayudará a reducir los errores de la no linealidad en la apertura del campo de visión.

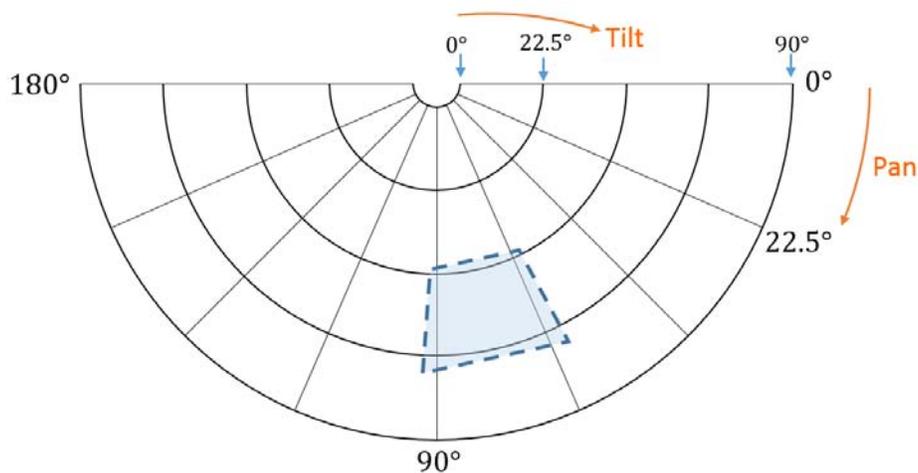


Figura 5.7: Zonas de escaneo mapeadas.

La exploración del entorno se puede efectuar de dos maneras, cuando se busca detectar un patrón en el entorno o cuando se desea detectar el movimiento de un objeto desconocido.

En el primero el sondeo será en tiempos continuos y constantes hasta coincidir objeto con el patrón, de esta manera se pueden extraer imágenes negativas y almacenarlas en una carpeta con el objetivo de utilizarlas para un entrenamiento.

El segundo será en tiempos regulares los cuales durarán periodos cortos, en estos periodos se adaptara la cámara al entorno que percibe, enseguida permanecerá un momento estacionada; posteriormente pasará a otra sección hasta cubrir todo el entorno, continuará así hasta que un objeto atraviese el campo de visión, es decir detectará el movimiento del objeto en cualquier parte del entorno.

Diseño de Software para el procesamiento de Imágenes.

5.1.2. Detección

5.1.2.1. Detección de movimiento

Con esta función se detectan los objetos por medio de su movimiento haciendo posible obtener las coordenadas del cuadro delimitador que lo encierra dentro de la imagen original, consiguiendo imágenes donde el objeto de interés sea señalado, despreciando el entorno que está a su alrededor.

La función que se implementó es (Vision.ForegroundDetector), se configuro los parámetros de la siguiente manera:

	Parámetro	Configuración
a	AdaptLearningRate	True
b	NumTrainingFrames	40
c	LearningRate	0.007
d	MinimumBackgroundRatio	0.5

a. Se habilita la tasa de aprendizaje adaptativa para que actualice el fondo ante cambios graduales en el entorno.

b. La cantidad de frames iniciales para la creación del fondo depende de la velocidad de captura de imágenes (fps), lo recomendable es de 3 a 5 segundos de entrenamiento.

c. La tasa de aprendizaje depende de los falsos movimientos que existan en el entorno, para entornos muy estáticos se puede establecer alrededor de 0, para entornos relativamente cambiantes se puede aumentar pero se corre el riesgo de no detectar objetos que efectivamente se mueven, lo recomendable es establecerlo menor a 0.01.

d. El umbral de detección se establece para especifica que pixeles serán el fondo o el primer plano, es decir, un valor cercano a 1 se hará mayor énfasis a la intensidad de luminosidad, valores cercanos a 0 será muy propenso a detectar sombras.

Este detector umbraliza la imagen capturada, es decir, el procesamiento se realizará sobre imágenes binarias, cuando se detecta un objeto se convierte en una estructura binaria (unos) y el fondo en (ceros), la estructura binaria por efectos de iluminación y de adaptabilidad del primer plano tiende a dividirse por grietas (donde existe un cambio drástico de iluminación) creando diversos objetos en el proceso de etiquetación, esto afecta al tratar de obtener el cuadro delimitador que lo encierra, pues encierra a los numerosos etiquetados, para solucionar el problema se utilizó operadores morfológicos, la estructura de referencia para aplicar los operadores fue de tipo cuadrada con diferentes dimensiones, primero se aplicó la erosión para eliminar el posible ruido, después se dilató la estructura para engrosar y conectar las partes aisladas, en seguida se volvió a erosionar para reducir a el tamaño al original.

Una vez realizado lo anterior se obtuvieron las coordenadas del cuadro delimitador de la estructura binaria con la función (blobAnalyzer).Obteniendo imágenes positivas para el entrenamiento cascada.

Imágenes positivas

Estas imágenes se obtuvieron automáticamente después de obtener las coordenadas del cuadro delimitador, estas coordenadas se aplicaron sobre la imagen original extrayendo solo la región del cuadro delimitador con la función (imcrop), se les asigno un nombre diferente a cada una y fueron almacenadas en una carpeta.

Imágenes Negativas

Se obtuvieron automáticamente cuando no se detectó ningún movimiento, se obtienen para mapear la zona donde no se encuentre el objeto en movimiento, estas imágenes se almacenaron en una carpeta y se les asigno un nombre diferente a cada una.

5.1.2.2. Entrenamiento de Patrones

El detector de patrones es sensible a los cambios de rotación de los objetos en 3D, es decir, solo se entrenará con un solo lado del objeto, no es factible entrenar un detector manteniendo todas las orientaciones, para detectar la totalidad de orientaciones del objeto es necesario y recomendable realizar el entrenamiento para cada lado.

Como se señaló al inicio para detectar patrones se necesita de un “banco” de imagines positivas y negativas, una vez que disponemos de estas, es posible hacer el entrenamiento para la detección de un patrón, para tal fin se utilizó la función (trainCascadeObjectDetector), para su correcto funcionamiento de manera automática requiere de un arreglo tipo estructura de dos campos que señalen los objetos en las imágenes positivas, en el primer campo de esta estructura se debe especificar el nombre y la dirección donde se almacena la imagen positiva, en el segundo campo se debe especificar las coordenadas del cuadro delimitador de la imagen en cuestión. La función se configuro de la siguiente manera:

	Parámetro	Configuración
a	Nombre	'prueba.xml'
b	positiveInstances	(array struct)
c	Imagenes_negativas	(path)
d	ObjectTrainingSize	'auto'
e	NegativeSamplesFactor	2
f	FalseAlarmRate	0.2
g	TruePositiveRate	0.995
h	NumCascadeStages	7
i	FeatureType	'Haar'

-
- a. Es el nombre con el que será llamado el archivo .XML, el cual tiene las especificaciones para detectar el objeto, el archivo se almacena en la carpeta actual (donde está el programa principal).
 - b. Es una variable de tipo (arreglo de estructuras), se crea cuando se capturan las imágenes positivas, para aumentar la robustez de detección se pueden editar las imágenes existentes añadiéndoles rotación o ruido, así como variando el brillo o contraste.
 - c. Aquí se indica la ruta de la carpeta que contiene las imágenes negativas, se recomienda tener un gran número de estas, más que las positivas. También es importante suministrar imágenes negativas que contengan objetos similares al de interés, así se mejora la precisión.
 - d. El tamaño de imágenes puede variar si se toman a diferentes resoluciones, si fuesen de distinto tamaño se pueden redimensionar a un tamaño único para todas. Se configuro para tomar las dimensiones reales de detección.
 - e. Factor que multiplica la cantidad de imágenes negativas por el número de muestras positivas usadas en cada etapa.
 - f. Tasa de falsos positivos en cada etapa, es el porcentaje de muestras negativas entrenadas incorrectamente clasificadas como muestras positivas. Establecerlo en valores bajos, incrementa el rigor para que las muestras positivas sean promovidas a las siguientes etapas y en ocasiones el entrenamiento requiere más muestras negativas o de menos etapas para completar el entrenamiento, aumentarlo el valor del parámetro disminuye la precisión de detección, es decir, detectará bastantes falsos positivos.
 - g. La tasa mínima de verdaderos positivos por etapa debe ser alto, pues determina cuantas muestras positivas no serán discriminadas como muestras negativas, sin embargo aumenta el tiempo de entrenamiento.
 - h. La cantidad de etapas es determinada directamente por la cantidad de muestras positivas y negativas, también depende de la tasa de falsos positivos y de la tasa de verdaderos positivos. Incrementar el número de etapas aumenta la exactitud de detección así como el tiempo de entrenamiento.
 - i. El tipo de característica que se extraerá del objeto dependerá de su estructura y/o textura que posea en su superficie, la extracción de características tipo Haar se recomienda para superficies que tengan cambios drásticos (contraste) de color o de intensidad, las características tipo Hog se recomienda para estructuras que tengan en su superficie variación de texturas o que no sean estructuras solidas (tengan huecos en la superficie).

Las siguientes fórmulas determinan la relación que existe entre el número de etapas, tasa de falsos y verdaderos positivos. Se puede ver que aumentando más etapas reduce la tasa global de falsos positivos, pero también reduce la tasa global de verdaderos positivos.

$$Tasa\ de\ Falsos\ Positivos^{Numero\ de\ etapas} \quad (5.1)$$

$$Tasa\ de\ Verdaderos\ Positivos^{Numero\ de\ etapas} \quad (5.2)$$

En general si se cuenta con “bancos” de imágenes en la orden de cientos, es recomendable aumentar el número de etapas y la tasa de falsos positivos, al contrario si se tiene una pequeña cantidad en el “banco” de imágenes se debe reducir el número de etapas y la tasa de falsos positivos.

Estableciendo un alto número de etapas con pocas imágenes ocasionará que el entrenamiento se detenga hasta consumir todas las muestras que dispone, por ejemplo se puede establecer 10 etapas y detenerse en la etapa 5, esto pasa porque la función no genera suficientes muestras negativas, si se quiere ahora entrenar con 5 etapas posiblemente terminará el entrenamiento, pero no será el mismo que cuando se estableció 10 etapas, debido a que el número de muestras positivas y negativas usadas es vuelto a calcular por el nuevo número de etapas.

Una vez que se configuran los parámetros, la función procede a entrenarlo, los valores establecidos determinarán el tiempo en que terminará.

5.1.2.3. Detección de Patrones

Una vez conseguido el archivo .XML, se utiliza la función (vision.CascadeObjectDetector) que implementa el archivo entrenado, se configura de la siguiente manera:

	Parámetro	Configuración
a	ClassificationModel	'prueba.xml'
b	MergeThreshold	25
c	MinSize	[50 50]
d	MaxSize	[200 200]
e	ScaleFactor	1.0018

- a. Nombre y ubicación del archivo .XML.
- b. Determina el umbral para despreciar las falsas detecciones alrededor del objeto de interés.
- c. Área mínima del objeto a detectar, debe ser igual o mayor al tamaño de las imágenes utilizadas en el entrenamiento.

d. Área máxima del objeto a detectar, conociendo que el objeto no es tan grande ni estará demasiado cerca de la cámara, el valor debe ser menor al tamaño de la imagen.

e. Este factor determina el ajuste de la búsqueda para múltiples escalas del objeto, se establece utilizando la siguiente fórmula:

$$ScaleFactor = \frac{size(I)}{size(I) - 0.5} \quad (5.2)$$

Donde I es la imagen y $size$ es la función que obtiene el tamaño de la imagen.

El detector escala la región de búsqueda en incrementos entre la mínima y máxima área utilizando la siguiente relación:

$$Region\ de\ busqueda = ((Training\ Size) * (ScaleFactor^N)) \quad (5.3)$$

Dónde: *Training Size* es el tamaño de las imágenes utilizadas durante el entrenamiento.

N : es el incremento actual.

Entendiendo la relación entre estos parámetros y conociendo los límites del tamaño del objeto es posible reducir el tiempo de cómputo.

Al utilizar este detector es posible encontrar el objeto en cualquier parte del entorno que previamente se proporcionó. Esta función regresa las coordenadas del cuadro delimitador que encierra el objeto de interés que ha localizado, con estas coordenadas se podría realizar un seguimiento mecánico del objeto, pero al realizar tal desplazamiento de la imagen ocurren discontinuidades en la adquisición de las coordenadas, (perdida por instantes del objeto), creando retardos. Para dar solución a tal escenario, se utilizara algoritmos de seguimiento, los cuales tiene la facultad de reducir el cómputo si se compara con el sistema de detección.

5.1.3. Seguimiento

Para realiza el seguimiento de objetos se deben identificar las coordenadas localizadas del objeto en el frame actual, las cuales indicarán la ubicación inicial del proceso de seguimiento, esto se realizara con el detector cascada previamente entrenado.

Además se utilizarán diferentes métodos de seguimiento que operan en circunstancias diferentes con el fin de comparar los resultados.

5.1.3.1. Seguimiento de color

Como se mencionó al inicio el algoritmo CAMShift sigue el histograma de color en el modelo HSV, es decir, debemos transformar la imagen original que está en el modelo RGB y pasarlo al modelo HSV con la función (rgb2hsv) y seleccionar solo el área donde se encuentra el objeto de interés el cual conviene tenga un color sólido que contraste con su entorno, para ello se implementó la función (vision.HistogramBasedTracker), esta debe inicializarse configurando los parámetros de la siguiente forma:

	Parámetro	Configuración
a	H	Seguidor
b	I	ImHSV
c	R	Coordenadas
d	N	12

- a. Es la variable a la que se designó el seguidor.
- b. Se trata de la imagen en modelo HSV que contiene el objeto de interés.
- c. Son las coordenadas del cuadro delimitador que encierra al objeto de interés proporcionadas en un vector de cuatro espacios [*x, y, ancho, alto*].
- d. Es el parámetro que indica cuantas iteraciones se realizarán para encontrar la máxima correspondencia con el objeto del frame anterior, un valor reducido ocasiona que el seguidor pierda rápidamente el objeto, sin embargo un valor alto producirá un tiempo computacional alto.

Dado que el algoritmo es iterativo, esta función además devuelve las nuevas coordenadas del cuadro delimitador del objeto seguido, también devuelve el porcentaje de correspondencia respecto al histograma de color original para cada frame, el cual es utilizado para conocer si el seguidor está realmente siguiendo el objeto original, pues como se indicó en la descripción del algoritmo, este tiende a “evolucionar” con cambios graduales que sufre el objeto en frames consecutivos, tales como la iluminación , rotación , etc., ocasionando que siga objetos similares que se encuentren cercanos al objeto de interés, este parámetro auxiliará cuando el porcentaje de correspondencia se a menor al 50%, indicando que posiblemente el objeto no sea el original.

Una vez hecho patente la pérdida del objeto, se puede volver a detectar y reasignar las coordenadas del cuadro delimitador a un nuevo seguimiento. Esto es posible con la función (initializeSearchWindow) donde se introducen las nuevas coordenadas; se trata de un reinicio de seguimiento.

5.1.3.2. Seguimiento de texturas

Como se mencionó en un principio este seguidor es sensible a las texturas que detecta, así que se debe indicar la región de interés (ROI) en la que extraerá las esquinas características. Esto se realizará con el detector cascada.

La detección de esquinas características se realizara con el método de Tomasi “Good features to track” el cual esta implementado con la función (detectMinEigenFeatures); se configuró de la siguiente forma:

	Parámetro	Configuración
a	I	Imagen de Intensidad
b	MinQuality	0.01
c	FilterSize	5
d	ROI	Cuadro delimitador

a. Es la imagen donde se encuentra el objeto de interés, debe ser en escala de grises, pues se extrae de ella la variación de intensidad en los pixeles (Gradiente).

b. La mínima calidad de esquinas permitida, este parámetro con un valor alto (0.9) remueve la mayoría de las falsas esquinas, es decir, solo extrae las más fuertes, pero si las texturas de los objetos son “suaves” puede no encontrar ninguna esquina, por el contrario un valor bajo detectara la mayoría de las esquinas en la textura indicada.

c. Este parámetro indica el tamaño del filtro Gaussiano, que por sus características tiene la habilidad de “suavizar” la textura, es decir, eliminar las esquinas más “débiles”, un valor alto puede ocasionar que distorsione las estructuras presentes en la imagen, se recomienda un valor bajo.

d. Es la región de Interés, es decir, son las coordenadas del cuadro delimitador que encierran al objeto de interés, las cuales son proporcionado por el detector Cascada en un vector de cuatro espacios $[x, y, ancho, alto]$.

Configurando apropiadamente los parámetros es posible encontrar las esquinas más destacadas que serán ideales para el seguimiento, la función entrega una matriz conteniendo las coordenadas de cada esquina $[x, y, \dots, x_n, y_n]$.

Una vez conseguidas las esquinas, se utilizarán para realizar el seguimiento de la textura del objeto, es decir, se encontrarán la correspondencia de esquinas similares entre frames.

Para lograr esto se utilizó la función (vision.PointTracker), en esta función se configuran los parámetros que restringieran las esquinas a seguir, de la siguiente manera:

	Parámetro	Configuración
a	MaxBidirectionalError	3
b	NumPyramidLevels	4
c	BlockSize	[15, 15]
d	MaxIterations	30

a. Error Bidireccional. Es la distancia en pixeles desde la ubicación original hacia la ubicación final después del seguimiento hacia atrás, se utiliza para anular los puntos que estén a mayor distancia que el error bidireccional. Usándolo de manera adecuada puede eliminar los puntos que no pueden ser seguidos de manera fiable. Esto ocurre cuando el objeto se desplaza en gran cantidad de pixeles entre frames, es decir, cuando la cámara es estática o por otro lado cuando la cámara es dinámica y no logra la misma velocidad del objeto. Estableciéndolo en valores altos aumenta el tiempo de cómputo, se recomienda valores mínimos.

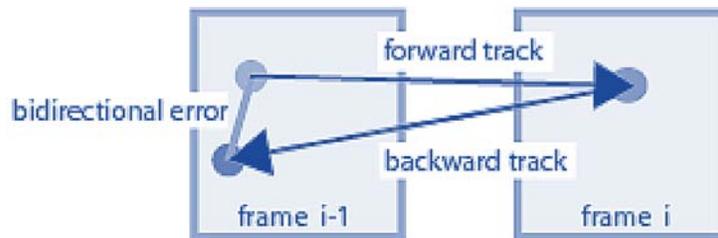


Figura 5.8: Error Bidireccional

b. Nivel Piramidal. Esto surge por el cambio de escala del objeto en una secuencia de frames (escalamiento del objeto). Cada nivel de pirámide está formado por un factor de dos en ancho y alto, en comparación con el nivel superior. El seguidor de puntos comienza el seguimiento de cada punto en el nivel de resolución más baja, y continúa el seguimiento hasta converger. El objeto transmite el resultado de ese nivel al siguiente nivel como la inicial estimación de las ubicaciones de los puntos. De esta manera, el seguimiento se refina cada nivel, hasta la imagen original. El uso de los niveles piramidales permite al seguidor manejar grandes movimientos de pixeles entre frames dificultando la pérdida del objeto, ya que se pueden abarcar mayores distancias que el tamaño de vecindad original, El aumento del factor piramidal permite un escalamiento considerable a una rapidez mayor, pero el costo computacional aumenta.

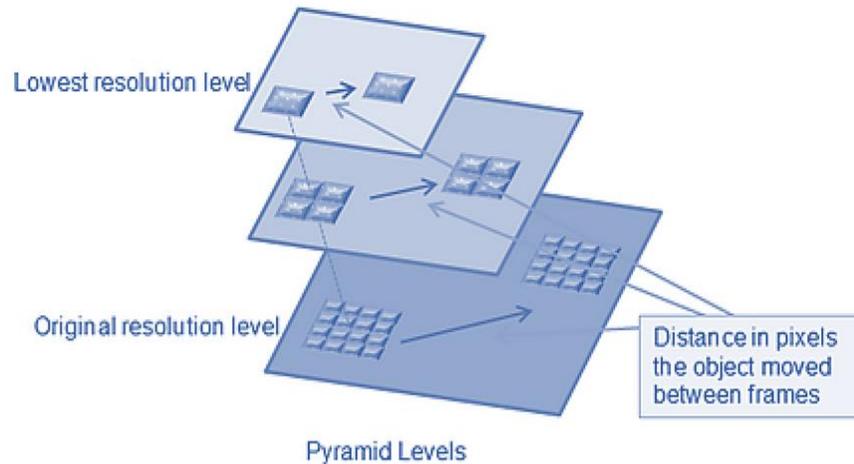


Figura 5.9: Diferentes niveles piramidales

c. Tamaño de Vecindad. Representa la vecindad alrededor de cada pixel (esquina) en que se realiza el seguimiento. Esta vecindad define el área para el cálculo de la matriz de gradiente espacial, es decir es la ventana de búsqueda para encontrar el pixel que más coincide del frame anterior, se especifica con un vector [ancho, alto], que comprende la ventana de búsqueda. Aumentar el valor del parámetro favorece el rápido desplazamiento del objeto, pero aumenta el tiempo de cómputo.

d. Iteraciones de Búsqueda. El algoritmo realiza una búsqueda iterativa para la nueva ubicación de cada punto hasta la convergencia. El algoritmo típicamente converge dentro de las 10 primeras iteraciones.

Una vez especificados adecuadamente los parámetros del seguidor de esquinas, se debe proporcionar las coordenadas de las esquinas previamente localizadas, así como la imagen donde se encuentra el objeto de interés.

Cada vez que se proporciona un nuevo frame la función busca la correspondencia entregando los siguientes valores [esquinas, esquinas validas, porcentaje], es decir, la función entrega las nuevas ubicaciones de las esquinas encontradas en una matriz $M \times 2$ $[x y]$, asimismo entrega una matriz binaria $M \times 1$ del mismo tamaño al de las esquinas indicando la validez de las esquinas entregadas, es decir, si han sido o no realmente seguidas.

Una esquina puede convertirse en invalida si sale de la imagen, también si su error bidireccional es mayor que el umbral establecido.

Igualmente devuelve el porcentaje de correspondencia para cada punto en una matriz $M \times 1$, estos valores corresponden al grado de similitud entre la vecindad alrededor de la ubicación anterior y la nueva ubicación de cada punto.

Permitiendo conocer la autenticidad del seguimiento, es decir, si el tamaño de la matriz (número de esquinas) es menor a 2 o si el porcentaje de la mayoría de las esquinas es menor al 70%, se puede deducir que el objeto posiblemente no sea el original o que ha cambiado de perspectiva, por lo tanto se desearía reiniciar el seguimiento de las esquinas.

Estas situaciones ocurren durante el seguimiento, es decir, el objeto que es seguido puede rotar y mostrar un lateral (cara) distinta a la que se extrajeron las esquinas, perdiéndose las mismas o ajustándose ellas si es que tiene en esa otra cara similares texturas a la original, causando que las esquinas reduzcan su porcentaje de semejanza.

Una vez hecho patente la pérdida del objeto, es necesario volver a detectar el objeto y extraer las esquinas para reasignarlas al seguidor, lo que es posible con la función (setPoints), en ella se indican con una matriz $M \times 2$ las ubicaciones de las nuevas esquinas.

SEGUNDA PARTE

5.1.4. Predicción de movimiento

Lograr la predicción del movimiento físico de los objetos en el plano con video dinámico (cámara motorizada), es complicado, pues en esta aplicación el sistema de seguimiento siempre tratará de centrar el objeto en la imagen, es decir, no cambiará gradualmente sus coordenadas (dentro de la imagen) con el tiempo como lo haría en un video estático.

La solución de este problema es utilizar una matriz (imagen) auxiliar, tomando como punto de referencia el centro de ambas imágenes, de la original (dinámica) y de la auxiliar (estática), después ir construyendo el desplazamiento gradual (Tendencia) en la imagen estática, con los desplazamientos parciales que ocurren en la dinámica.

El vector de desplazamiento creado en la estática es el esperado por el algoritmo de Kalman, pues con estos desplazamientos continuos es posible pronosticar y corregir el movimiento físico en el plano y sobre todo -predecir en t tiempos posteriores a la ubicación corregida del objeto.

Una vez conseguido este desplazamiento "virtual", es necesario referenciarlo ahora a la imagen dinámica para obtener las coordenadas predichas respecto al centro.

Para lograr tal hazaña de predicción es necesario configurar adecuadamente el Filtro de Kalman, es decir, indicar las condiciones iniciales de posición y velocidad del objeto (información a priori), así como todas las particularidades que contribuirán a la correcta predicción. Quedando de la siguiente manera:

Modelo del estado de transición

$$\mathbf{a.} \quad A(k-1) = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.4)$$

Modelo de mediciones

$$\mathbf{b.} \quad H(k) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \quad (5.5)$$

Modelo de control

$$\mathbf{c.} \quad B(k) = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} \quad (5.6)$$

Covarianza del ruido en medición

$$\mathbf{d.} \quad R(k) = \begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix} \quad (5.7)$$

Covarianza del ruido en el proceso

$$\mathbf{e.} \quad Q(k) = \begin{bmatrix} 0.001 & 0 & 0 & 0 \\ 0 & 0.001 & 0 & 0 \\ 0 & 0 & 0.001 & 0 \\ 0 & 0 & 0 & 0.001 \end{bmatrix} \quad (5.8)$$

Covarianza del error a posteriori

$$\mathbf{f.} \quad P(k-1) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.9)$$

Estos parámetros son implementados con la función (`vision.KalmanFilter`):

a. El modelo está definido por la cantidad de variables que depende el estado medido, en este caso se trata de las posiciones y velocidades $[x, y, \dot{x}, \dot{y}]$, resultando en una matriz de 4x4, una vez establecido este parámetro no es posible cambiarlo,

b. Es una matriz donde se ingresan los valores medidos en este caso son $[x, y, 0, 0]$ también se puede ingresar las velocidades que adquiere el objeto entre frames.

c. El modelo describe las entradas de control para cada estado del sistema, es un vector de cuatro filas, $[x; y; Vx; Vy]$

d. Se recomienda un valor reducido, no próximo a cero, para que la medida actual sea cada vez más y más confiable

e. El ruido en el proceso se considera mínimo por la naturaleza digital del sistema y menor a los parámetros de R (se confiara más en la predicción que en las medidas)

f. Es la matriz que contiene la tendencia al cambio de los estados, los adquiere recursivamente

La forma de predecir en t tiempos futuros es utilizando el siguiente algoritmo.

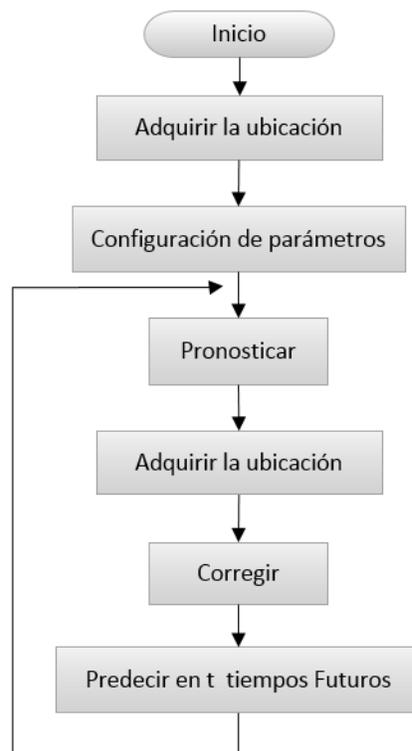


Figura 5.10: Algoritmo de Kalman implementado.

Las coordenadas que se entregan en el modelo de medición son las del Centroide del objeto seguido después de haberse centrado en la imagen dinámica, ya que este centro será la referencia, a partir de aquí es cuando se comienza a predecir el movimiento, con ayuda de la imagen auxiliar, después de cada nueva predicción se devolverán las coordenadas en el mismo formato en que se obtuvieron,

es decir, respecto al centro de la imagen dinámica, utilizando no solo para efectos de visualización, sino que también para el cálculo del ángulo de adelanto.

Como se indicó en un inicio, se trata de un proceso cíclico donde constantemente se predice con la función (predict), y esta predicción es constantemente corregida por medio de la actualización de las mediciones con la función (correct), es decir, además del seguimiento de los algoritmos de visión artificial se utiliza el filtro de Kalman para eliminar ruido en la obtención de datos realizando un seguimiento más refinado, así mismo cuando ocurra la ausencia de estos datos, es decir, cuando no están disponibles (en oclusión), el filtro predecirá la ubicación del objeto físico en ese momento.

Después de disponer de las mejoras hechas al seguimiento, se predice la ubicación en t lapsos de tiempos, estos tiempos obedecen al tiempo de muestreo, es decir, de los frames por segundo en que se capturan, se puede decir que predice en frames futuros, se utiliza la función (predict (t)), donde t indica el número de frames futuros.

El horizonte de predicción que otorga con fidelidad es reducido, de algunos frames futuros, pero son suficientes para la aplicación, si exigimos al sistema predecir un horizonte amplio su fiabilidad reduce, y empeora cuando ocurre oclusión pues no se corrige con las medidas del objeto, predice a ciegas.

Estas coordenadas estimadas en t tiempos son las que seguirá el sistema de derribo, es decir, se utilizaran para calcular el ángulo de adelanto para coordinar el posicionamiento de los actuadores y lograr una minuciosa puntería.

El para el correcto funcionamiento del ángulo de adelanto es esencial obtener la velocidad de desplazamiento, esta se obtiene de la “tendencia corregida” que corresponde a la velocidad instantánea del objeto.

5.1.5. Diseño del Software para la triangulación láser

El sistema de triangulación láser se acoplo de tal manera que el láser triangulara verticalmente, en apariencia se deduce que conviene la triangulación en sentido horizontal (existe mayor resolución en la imagen) pero las características físicas no lo permitieron.

Se configuró el acoplamiento cámara-láser con las siguientes propiedades:

	Parámetro	Configuración
a	x	7.2 cm
b	β	24.24°
c	P_x	288 pixeles
d	α	(11.6 – 20.87 – 22.48)°

a. La distancia entre el lente láser y el centro focal de la cámara se eligió con el propósito de obtener doscientos centímetro como distancia máxima de medición, debido a que la cámara se configuro con baja resolución teniendo como consecuencia que una distancia mayor a esta el objeto sea casi imperceptible.

b. La forma de obtener este ángulo beta es de manera experimental utilizando una triangulación rectangular teniendo como referencia la ubicación del reflejo, obteniendo los catetos del mismo. Se utilizó la siguiente formula:

$$\beta = \tan^{-1}\left(\frac{co}{ca}\right) \quad (5.10)$$

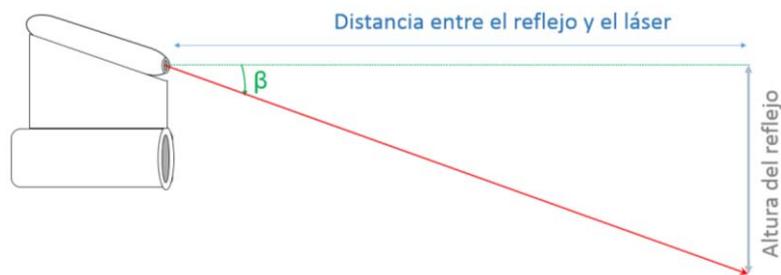


Figura 5.11: Forma de obtener el ángulo del láser.

c. Se configuro de esta manera ya que se tomo en cuenta la resolución de la imagen (352x288) que se utilizará en los algoritmos de vision artificial.

d. El angulo de apertura alfa de la cámara se obtiene de la misma manera que beta, solo se considera a partir del centro focal como triangulo rectangulo. Utilizando la siguienet formula:

$$\alpha = \tan^{-1}\left(\frac{co}{(ca)(2)}\right) \quad (5.11)$$

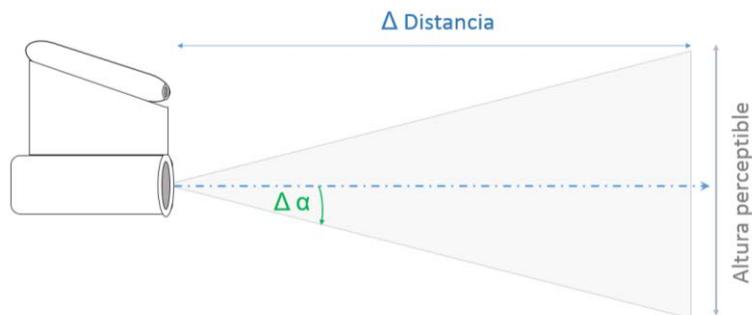


Figura 5.12: Forma de obtener el ángulo de apertura en la cámara.

En consecuencia de tal configuración las distancias mínima, máxima y centro focal son:

$$dA = \frac{x}{\tan(\alpha) + \tan(\beta)} = 11 \text{ cm} \quad (5.12)$$

$$dB = \frac{x}{\tan(\beta) - \tan(\alpha)} = 198 \text{ cm} \quad (5.13)$$

$$d = x(\tan(\beta')) = 16 \text{ cm} \quad (5.14)$$

Se comprobó experimentalmente que la cámara tiene un ángulo de apertura no lineal en las coordenadas y , es decir, la cámara tiende a aumentar la apertura de su campo de visión con forme aumenta la distancia de percepción, por tal motivo la validez de las ecuaciones anteriores se cumplen cuando se corrige la curvatura por medio de una linealización.

La desviación angular (Da) que se midió entre 198 y 11 centímetros fue de:

$$Da = (\alpha \text{ min}) - (\alpha \text{ max}) = 10.88^\circ \quad (5.15)$$

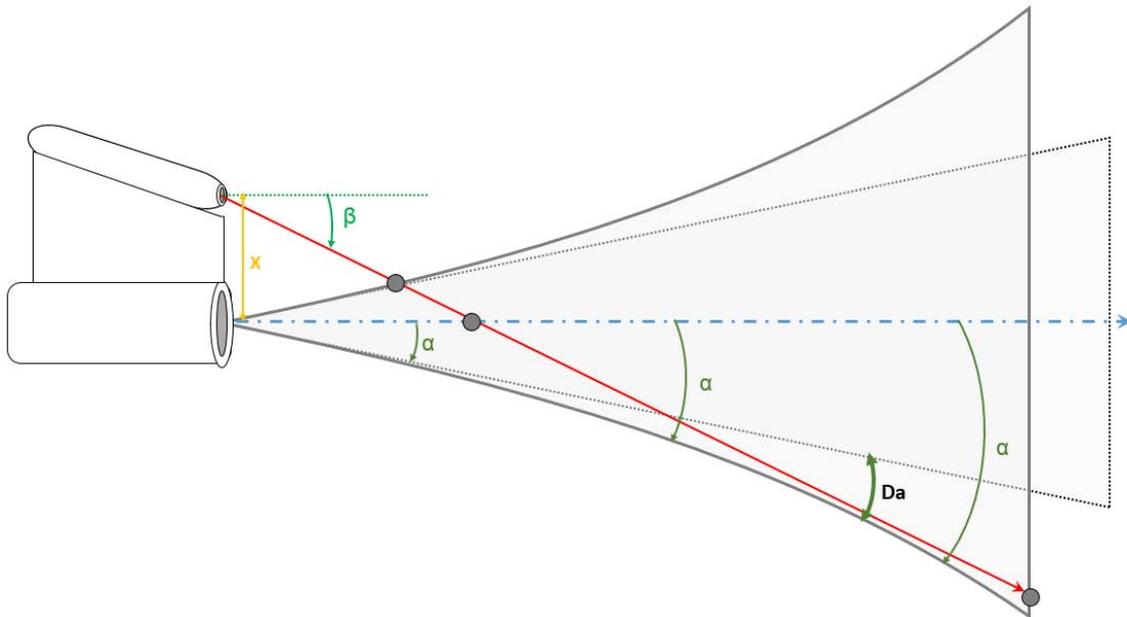


Figura 5.13: No linealidad presente en la apertura de la cámara.

Para tal fin de linealización, se creó dos factores de corrección de manera experimental, es decir, se calculó en la distancia mínima, media y máxima su respectivo ángulo de apertura (α). Y a partir de estos ángulos se escaló respecto a los pixeles de la imagen en y . Con el propósito de que a cada pixel seleccionado le corresponda un ángulo (α) de la no linealidad como se observa en la imagen.

Tabla de valores Medidos			
	Distancia	Ángulo de apertura	Coordenada y
Mínima	11 cm	11.6°	0 pixeles
Media	104.5 cm	20.87°	269.3 pixeles
Máxima	22.48 cm	22.48°	288 pixeles

Los factores se calculan como:

$$Fc1 = \frac{\Delta\alpha 1}{coord(y)med} \quad (5.16)$$

$$Fc2 = \frac{\Delta\alpha 2}{(coord(y)med) - (coord(y)max)} \quad (5.17)$$

Donde:

$$\Delta\alpha 1 = \alpha med - \alpha min \quad (5.18)$$

$$\Delta\alpha 2 = \alpha max - \alpha med \quad (5.19)$$

Estos factores afectan directamente la obtención de los catetos y por consecuencia el cálculo final de la distancia. Asegurando cambiar en cada pixel el ángulo de apertura con las siguientes ecuaciones.

$$\alpha compensada 1 = (P_y * Fc1) + \alpha min \quad (5.20)$$

$$\alpha compensada 2 = ((P_y - coord(y)) * Fc2) + \alpha med \quad (5.21)$$

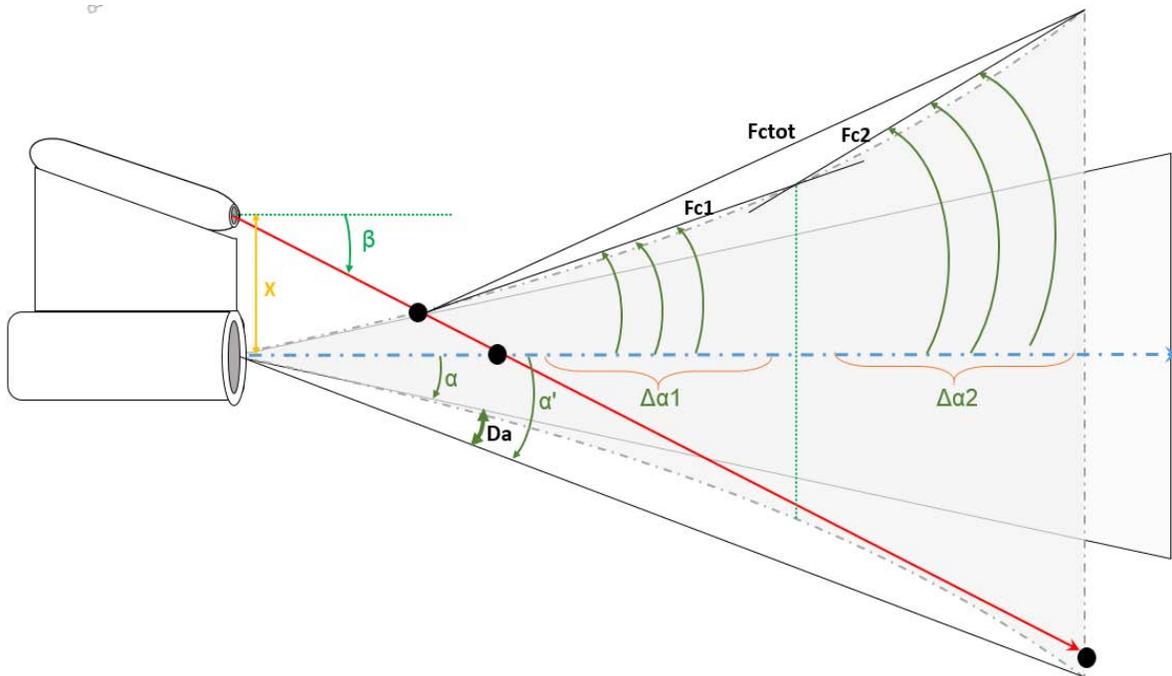


Figura 5.14: Linealización del ángulo de apertura.

En un inicio se pensó en linealizar desde el ángulo de apertura mínimo hasta la apertura máxima pero se observó que por la característica curvada de la no linealidad ocurría una desviación lineal (error) entre la medida calculada y la real, este error se incrementó gradualmente agravándose más en la distancia media (F_{ctot}).

Para darle solución se decidió utilizar dos linealizaciones partiendo de la distancia media (donde se produjo el mayor error), y en consecuencia se obtuvieron dos factores de corrección (F_{c1} y F_{c2}), esto redujo considerablemente la desviación lineal (error) en la medición. Como se puede apreciar en la figura.

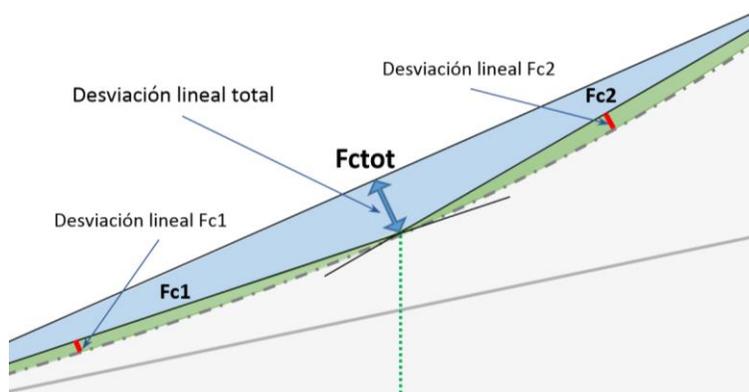


Figura 5.15: Comparativa entre una y dos linealizaciones.

De la misma manera se intuyó que reduciendo el ángulo de apertura media (pocos grados) se sería capaz de mejorar la medición y al mismo tiempo que reducir el error. Como se puede observar el error sin esta mejora tiende a aumentar y a disminuir, con la mejora se pretende mantener el error sin tener cambios tan bruscos.

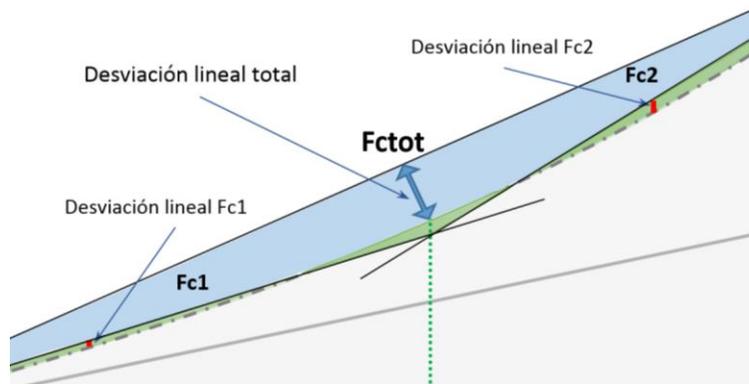


Figura 5.16: Mejora en a linealización.

La obtención de la distancia inicia cuando el objeto está centrado en la imagen, si esto acontece, se toma un frame y se convierte en una imagen de intensidad con la función (rgb2gray), es de esta manera ya que el reflejo del haz laser en un punto es luminoso, se visualizara como una región con niveles de intensidad altos (blancos), posteriormente se extrae solo la región central de la imagen, porque de ante mano sabemos que el objeto está centrado y podemos despreciar el resto de la imagen, agilizando el cálculo computacional y evitando falsos reflejos.

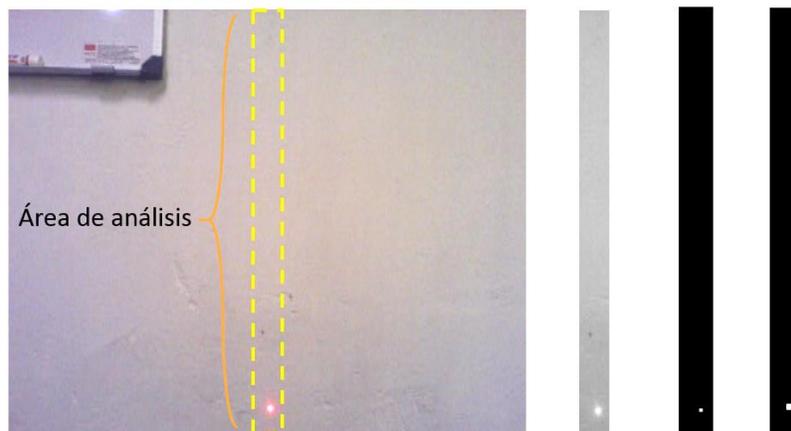


Figura 5.17: Procesamiento de la imagen donde se obtiene a zona de medición.

Una vez obtenida el área central de la imagen de intensidad se convierte a binaria aplicando un umbral en el histograma de intensidad, el umbral se aplica en la región más luminosa del histograma para eliminar todos los posibles falsos luminosos, es decir, la mayoría de los pixeles con intensidad luminosa menor a este umbral se convertirán en ceros (negro) y los mayores al umbral serán unos (blanco).

El umbral es del 90% del histograma y se utilizó la función (im2bw).

También se aplicaron filtros morfológicos para mejorar la calidad de la estructura binaria.

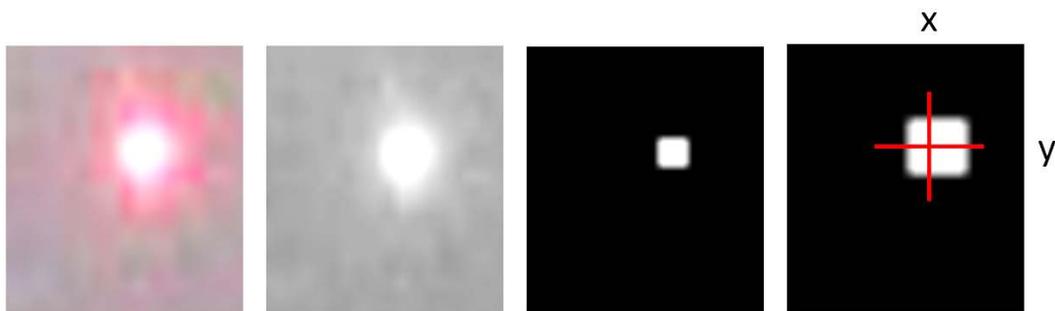


Figura 5.18: Forma en que se obtuvo las coordenadas del punto luminoso.

La siguiente condición espera a que el haz laser refleje sobre la superficie del objeto, si esto ocurre, se extrae el Centroide de la estructura binaria con el fin de obtener las coordenadas de la posición (x, y), esta coordenada se utilizara para el cálculo final de la distancia.

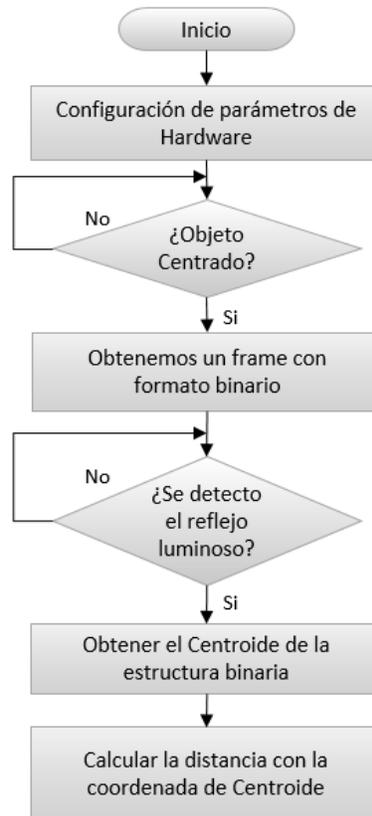


Figura 5.19: Algoritmo para obtener la medición de la lejanía del objeto.

5.1.6. Diseño de la tabla de ganancias del control adaptativo

El objetivo del sistema de control es lograr centrar el objeto en la imagen por medio de las coordenadas (x, y) matriciales que existen en el arreglo de pixeles y en consecuencia realizar el control de actuadores para que jamás (idealmente) deje de seguir el objeto.

Como previamente se ha mencionado la razón por la cual se quiere centrar es porque de esta manera es posible realizar una predicción de la posición futura con cámara dinámica, de la misma manera es la única forma de realizar la medición indirecta con triangulación láser, y sobre todo se será posible realizar un seguimiento fiable adaptándose al movimiento que tenga el objeto.

Como se puede observar en el esquema de control, la variable a controlar es la posición deseada $D [x, y]$ del objeto dentro en la imagen, esta posición se desea central en ambos ejes, sin embargo las entradas al sistema son cambiantes (dinámicas) con el tiempo, y algunas veces cambia la entrada antes de alcanzar el estado estable, estos variados cambios son consecuencia de la naturaleza del objeto en movimiento, a esto se le conoce como variable de entrada.

La variable de salida es la posición real del objeto $R [x, y]$, cuya magnitud se mide

La manera de obtener estos parámetros es utilizando el sensor CDD Discreto el cual tiene la capacidad de entregar valores entendibles para la planta, a esto se le conoce como sistema de medición, que tiene como propósito producir un efecto en la salida al cambio de valor en la variable controlada.

En control la planta es el sistema que se encarga de procesar la variable a controlar, en esta aplicación se trata de procesador de imágenes en él se obtiene la posición del objeto dentro de la imagen.

El controlador es el sistema encargado de adaptarse para corregir o limitar la desviación de la variable medida con referencia al valor previamente seleccionado (centro de la imagen), el sistema está conformado por los elementos necesarios para efectuar la variación de velocidad rotacional de los servomotores.

El mecanismo de adaptación se consigue por medio de la medida de la posición actual del objeto dentro de la imagen, seleccionando la velocidad adecuada para alcanzar el objetivo del control.

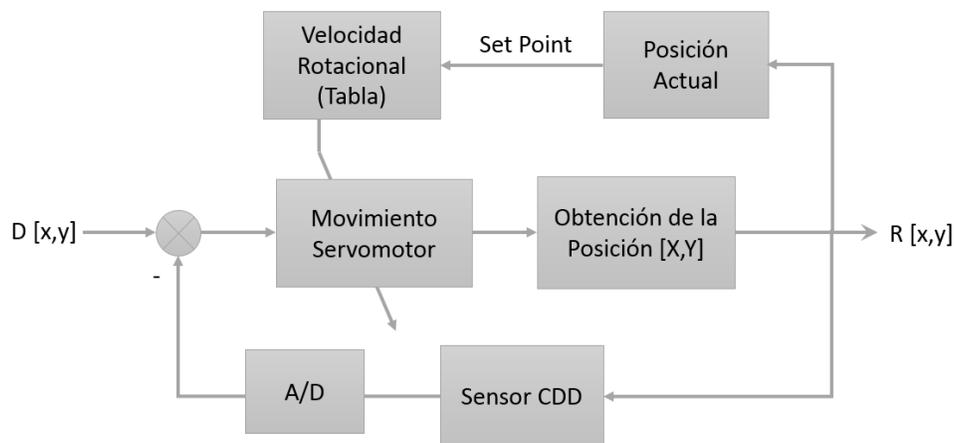


Figura 5.20: Representación particular del esquema de control adaptativo para el sistema.

Sabiendo que el sistema de actuación (Microcontrolador-servomotor) se configuró con la finalidad de que por pulso enviado el sistema de actuación se moviera un grado, se creará una tabla de correspondencia (ganancias), conteniendo los estados de adaptabilidad para gestionar los pulsos y por lo tanto los grados de movilidad en los actuadores.

La tabla está en función de la posición del objeto respecto al centro, logrando una respuesta sobreamortiguada en el control de variable (posicionamiento), eliminando el seguimiento brusco.

La figura muestra la manera de cómo se consiguió la tabla de ganancias. Las ganancias se obtuvieron a partir de la división de la imagen en (x, y) , de esta manera se tiene cuadrantes que en función de la localización del objeto en estos se obtiene la dirección para cada actuador y además su velocidad adaptada.

Comprendiendo que a cada posición le corresponde una velocidad en ambos ejes, el funcionamiento es el siguiente:

Conforme se acerca la ubicación del objeto de interés al centro disminuye la velocidad, de igual forma cuando el objeto aumenta la rapidez de su movimiento tenderá a alejarse del centro y en consecuencia ocupar celdas que envíen un pulso de habilite una velocidad mayor forzándolo a regresar al centro de la imagen. Es decir, se adaptara la velocidad rotacional al cambio de posición trasnacional del objeto.

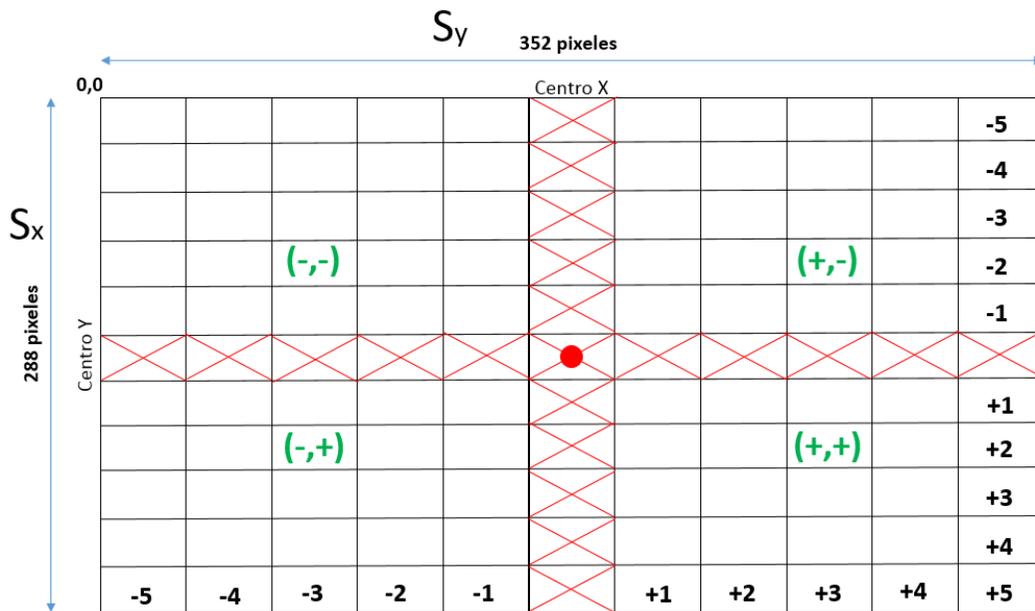


Figura 5.21: Ganancias aplicadas.

Sabiendo que el sistema siempre intentará centrar el objeto, cuanto más alejado se encuentre del centro, significara que está aumentando su velocidad, por lo tanto el sistema reaccionara aumentando la velocidad rotacional del sistema de actuación por pulso enviado.

El siguiente esquema especifica la manera en que se implementó, en él se observa ambas retroalimentaciones propias del control adaptativo.

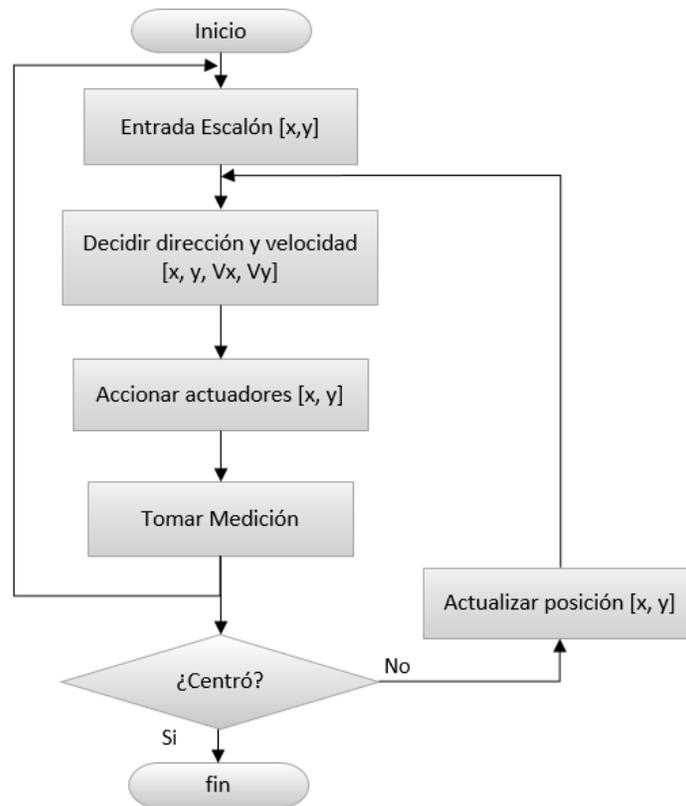


Figura 5.22: Algoritmo para implementar el control adaptativo.

5.1.7. Diseño del Software para el control de servomotores

La programación se realizó en lenguaje C utilizando el compilador CCS (Custom Computer Services). Este compilador ha sido diseñado específicamente para Microcontroladores PIC obteniendo la máxima optimización del compilador con estos dispositivos producidos por la compañía Microchip Technology. Dispone de una amplia librería de funciones predefinidas, comandos de procesado. También provee drives para la comunicación con diversos dispositivos.

La estructura general del algoritmo

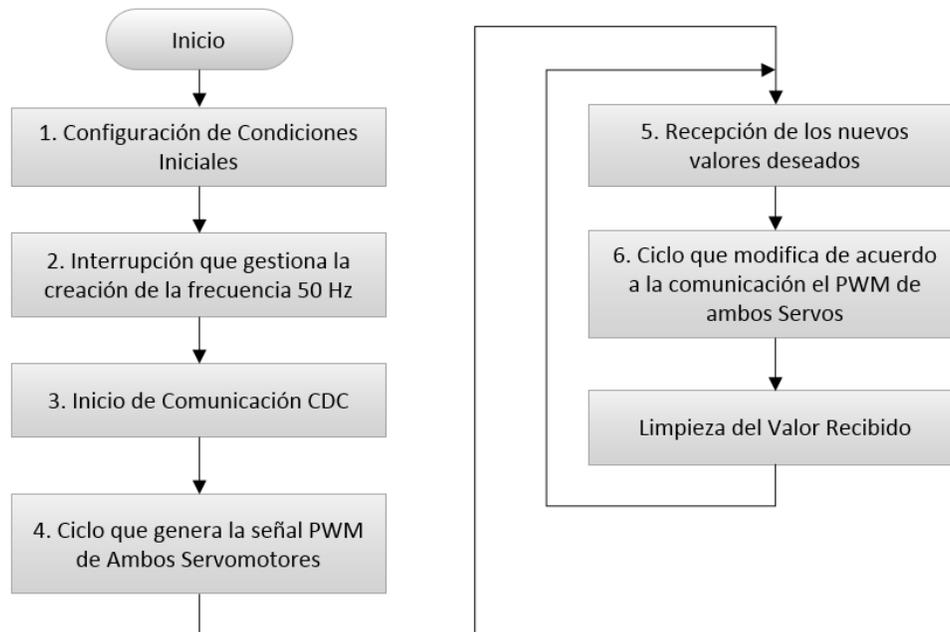


Figura 5.23: Algoritmo general para accionar el sistema de actuación.

1. Inicialmente se configuró la frecuencia de oscilación habilitando los fusibles internos para operar a 48Mhz, se invocaron las librerías USB_CDC, se configuran los pines del Microcontrolador como salidas, se delimita la cantidad mínima y máxima de tics que duraran los pulsos en alto, además se establece prioridad a la interrupción Timer0 (RTCC) sobre la comunicación para asegurar la precisa frecuencia.

2. Posteriormente se gestiona la interrupción del Timer0 para generar los ciclos de 50Hz y en cada uno de ellos se habilita una bandera que da pie al inicio de la señal PWM. Se explicara más a detalle en el siguiente apartado.

3. Para iniciar la comunicación USB_CDC se llamó a las funciones que habilitan la comunicación (usb_cdc_init), así mismo se “pregunta” continuamente si el dispositivo está conectado al puerto USB con la función (usb_task), si lo anterior es cierto, se administra la conexión “consultando” periódicamente al Host (PC) si el dispositivo esta enumerado (reconocido) con la función (usb_enumerated), todo lo anterior ocurre cuando la conexión ha sido exitosa y está operando con normalidad con lo que es posible enviar y recibir paquetes de datos, en esta aplicación solo recibimos datos y utilizamos la función (usb_cdc_kbhit) para saber si hemos recibimos algún carácter en el buffer, si sucedió leemos ese carácter y lo asignamos a una variable, y podemos disponer de ella para cambiar los parámetros de funcionamiento.

4. Una vez llegado a este punto se construye a partir del ciclo de 50Hz la señal PWM, es decir, en base a los tiempos y la bandera de fin de ciclo, se activa la señal en alto y se mantiene en este estado por un instante, posteriormente es puesto en bajo de acuerdo a la posición que deseemos. Se explicara más a detalle en el siguiente apartado.

5. Se reciben los datos y de acuerdo a su valor se modifica el ciclo de trabajo, variando la posición radial de los actuadores en distinto sentido y a distinta velocidad en ambos servomotores, después de ser utilizado el último dato recibido se elimina, esperando el nuevo dato.

5.1.7.1. Creación de frecuencia

Como se mencionó anteriormente en el capítulo 1, los servomotores operan a 50Hz o 20ms y con un ciclo de trabajo que varía entre 2.5% - 12.5% es decir necesita de pulsos constantes en la orden de microsegundos controlados puntualmente.

Los servomotores operan entre (0 – 180) grados y para cada grado tiene asignado un ancho de pulso en alto, el ancho de pulso por defecto en la mayoría de los servomotores varía entre (0.5 – 2.5) ms, a partir de este rango se escala para obtener la resolución de movimiento, es decir, para 0.5 ms, la posición en la terminal de actuación será de 0 grados, por el contrario si el pulso en alto es de 2.5 ms, la posición en la terminal de actuación será de 180 grados.

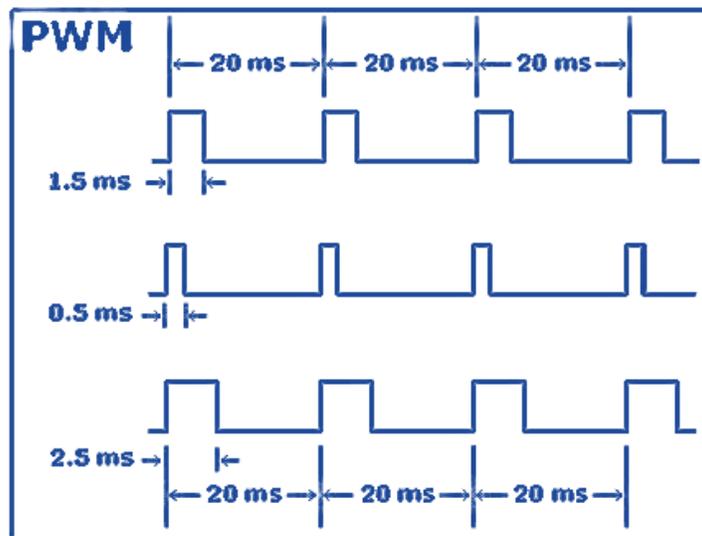


Figura 5.24: Anchos de pulso PWM permitidos.

El Microcontrolador cuenta con dos módulos PWM fácilmente configurables, pero sabiendo que el sistema necesita de dos actuadores y al tratar de utilizar y coordinar ambos módulos PWM se crean retrasos que hacen inexacto el movimiento, para su correcto funcionamiento necesitan recibir pulsos

en tiempos precisos. Por tal razonamiento se decidió utilizar Interrupciones por desbordamiento del Timer0.

El Timer0 es un temporizador/contador de (eventos internos/eventos externos) respectivamente con una resolución de 8/16 bits con la capacidad de generar interrupción por desbordamiento, es decir, puede contar “tics” del reloj externo y desbordarse cuando se cumpla una condición y de esta manera controlar la duración de los pulsos en estado alto o bajo, reproduciendo la modulación de una señal PWM de manera precisa.

Las interrupciones por desbordamiento del Timer0 se realizan cada 20ms y sabiendo que cada servomotor utiliza un tiempo máximo en alto de 2.5ms, es posible llegar controlar hasta 8 simultáneamente, sin embargo la aplicación solo requiere de dos.

Para poder realizar con eficacia lo anterior, reside todo en el cálculo de los tiempos, conociendo que el Microcontrolador opera a 48Mhz y que la frecuencia de conteo es una cuarta parte de la frecuencia de reloj ($f_{osc}/4$) en los Microcontroladores PIC, se consigue un frecuencia de 12Mhz (o 83.3ns), la cual no es suficiente para realizar el desbordamiento.

El ciclo maquina se ejecuta en 4 ciclos reloj La frecuencia de oscilación (F. Osc):

$$T_{maq} = \left(\frac{1}{F. Oscilador} \right) \times 4 \quad (5.22)$$

Sin embargo es posible insertar un pre-escalador, que es un divisor de frecuencia programable el cual puede escalar la frecuencia en 2, 4, 8, 16, 32, 64, 128 o 256, y así obtener el conteo de tics precisos para realizar el desbordamiento y por consecuente la interrupción del Timer0.

Para realizar el cálculo de desbordamiento, se utiliza la siguiente formula:

$$Interrupcion\ TMR0 = 4 \left(\frac{1}{F. Oscilador} \right) (256 - Valor\ TMR0) (Preescalador) \quad (5.23)$$

Donde:

F. Oscilador: Frecuencia determinada por el reloj externo.

Valor TMR0: Valor pre-cargado al conteo para desbordar en menos tics.

Pre-escaler: Divisor de frecuencia.

A una frecuencia de 48Mhz, con pre-escaler de 1:128, el Timer0 configurado a 8 bits y con una precarga de 0 tics se desborda cada 2.73066ms; como se utiliza con una resolución de 8 bits, es decir, $2^8 = 256$ tics debe contar para desbordar y cada tic por consiguiente tiene una duración de 10.666us, empleando la siguiente formula:

$$tic = Interrupcion \frac{TMRO}{2^8} \quad (5.24)$$

Con lo anterior se puede intuir que el desbordamiento o (RTCC) es menor al requerido que es de 20ms, por tal razón es necesario realizar desbordamientos continuos y para llegar exactamente a 20ms se deberá cerrar el ciclo con un valor precargado.

En esta configuración se necesitó 7 desbordamientos continuos más uno con 179 tics de pre-carga para el ajuste fino y así generar los 20ms.

$$\begin{aligned} 20ms &= (7 \times RTCC \text{ completas}) + (1 RTCC - 179 \text{ tics}) \\ 20ms &= (19.11ms) + (0.89ms) \end{aligned}$$

Se puede obtener diferentes resoluciones modificando el preescalador y la frecuencia del oscilador.

5.1.7.2. Creación de PWM

Para lograr la creación de PWM, el algoritmo contempla un ciclo donde se cuente y genere la frecuencia óptima para los servomotores, cuando alcanza el inicio del ciclo de 50Hz establece en alto el pin de salida, al mismo tiempo habilita una bandera que indica que el pin esta en alto, con lo que automáticamente inicia la comparación del conteo del Timer0 con el valor que deseamos que dure el estado positivo (pin en alto) de la señal PWM, cuando el valor deseado es rebasado, automáticamente el pin de salida se establece en bajo y permanecerá así hasta terminar el tiempo faltante para cubrir los 20ms.

Después de completar los 20ms vuelve a comenzar el pin en alto, con lo que mantiene al servo en la misma posición, hasta que el valor deseado sea cambiado.

El valor deseado tiene que fijarse en cantidad de tics, de esta manera logra compararse con el conteo de tics que se utilizan para el desborde. Como arriba se mencionó cada tic dura 10.66us, ahora conociendo que el servomotor opera de 500us a 2500us tiene un rango de operación de 2000us y la resolución del PWM se calcula como:

$$Resolución PWM = \frac{2500us - 500us}{10.666us} = 187.511 \quad (5.25)$$

Sabiendo que el servo opera de (0 – 180) grados y que la resolución del PWM es 187.511, se puede conocer la resolución rotacional que tendrá el actuador por aumento de un solo tic en el ancho de pulso, se calcula como:

$$Resolución\ Rotacional = \frac{180^\circ}{187.511} = 0.96^\circ \quad (5.26)$$

Con esto es posible divisar que el servomotor se moverá aproximadamente un grado por tic ($0.96^\circ \approx 1^\circ$); con lo anterior ahora es posible determinar la velocidad rotacional a la que girará para seguir al objeto.

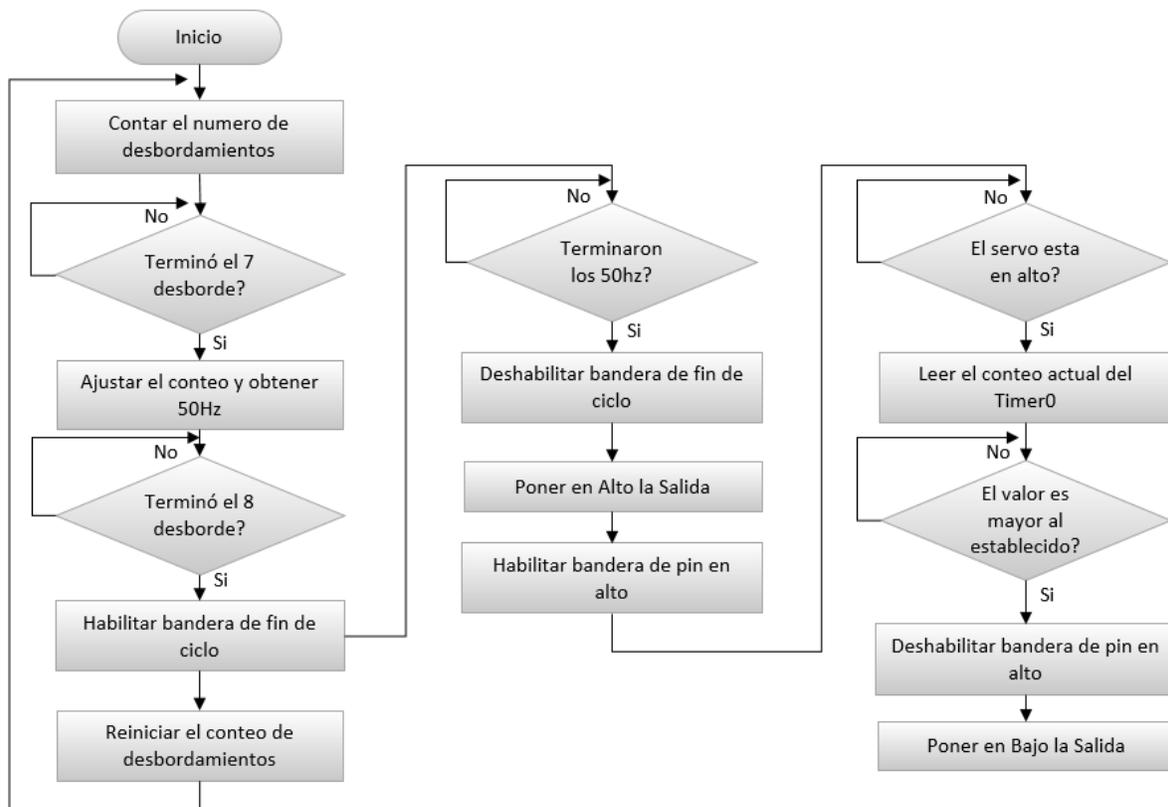


Figura 5.25: Algoritmo para crear pulsos PWM mediante interrupciones.

5.2 DISEÑO DE HARDWARE

5.2.1. Tarjeta de interfaz

La tarjeta recibe instrucciones del CPU y controla el funcionamiento de los servomotores con el Microcontrolador PIC18F4550, para logara tal hazaña es necesario acondicionar y configurar el Microcontrolador.

El sistema de adquisición puede ser alimentado con el voltaje otorgado por el puerto USB o por un voltaje externo de 5V, por las propiedades de interfaz PC-Actuador, se utilizara la alimentación por USB. Para la correcta operación en esta configuración requiere polarizarse el pin VDD con el mismo voltaje proporcionado del USB ($5V_{dc} - max500mA$), Además es necesario conectar un capacitor de $470nf$ con el fin de mantener el voltaje entre ($3V - 3.6V$) en la comunicación USB.

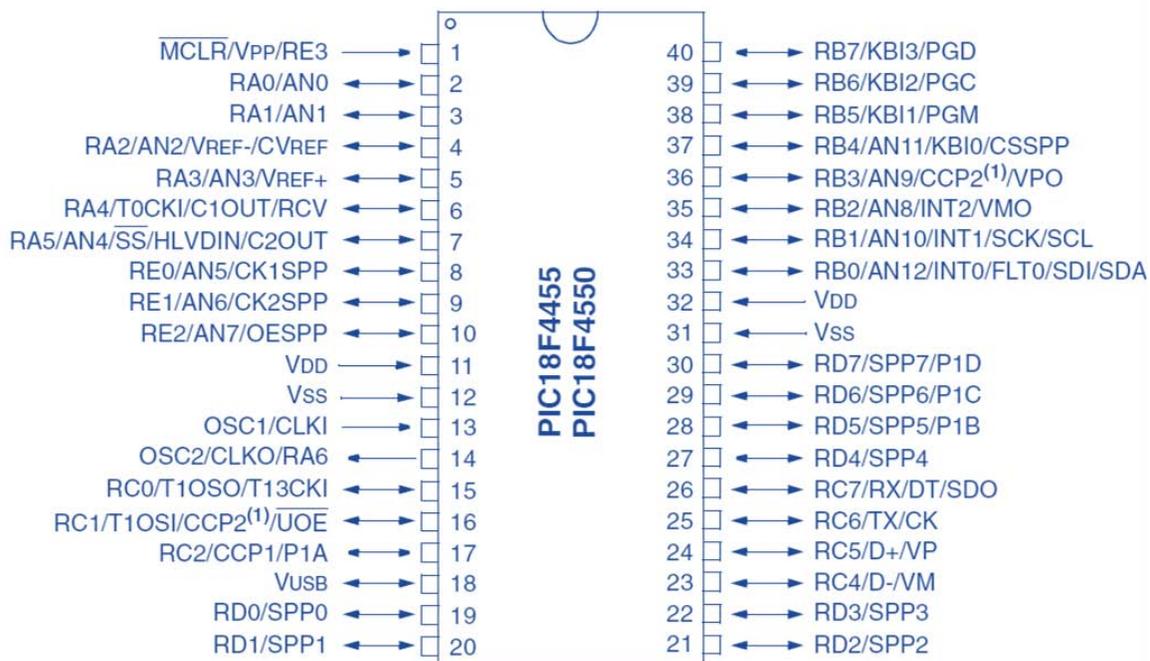


Figura 5.26: Patillaje del Microcontrolador 18F4550.

El Microcontrolador necesita un circuito oscilador que le indique la velocidad que debe trabajar, este módulo crea una señal de reloj (o tren de pulsos) a determinada frecuencia, que a la vez sincroniza su funcionamiento.

El Microcontrolador se configuro de tal manera que funcionara a la máxima velocidad a la que puede operar que es de 48Mhz.

Para ello en las terminales OSC1 y OSC2 se conectó en paralelo un cristal de 20Mhz y dos capacitores de 15pf para cada terminal con referencia a tierra como lo indica la hoja de datos que proporciona el fabricante.

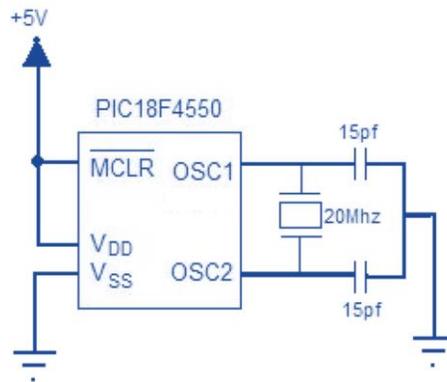


Figura 5.27: Circuito de oscilación.

El Microcontrolador puede generar una señal de reset por software si se le aplica un cero lógico en pin MCLR, aprovechando esta característica se incluyó en el diagrama un sistema de reseteo por push button el cual se acondiciono para evitar falsos disparos. Esta función auxilia al sistema cuando por alguna razón se tenga que reiniciar la comunicación.

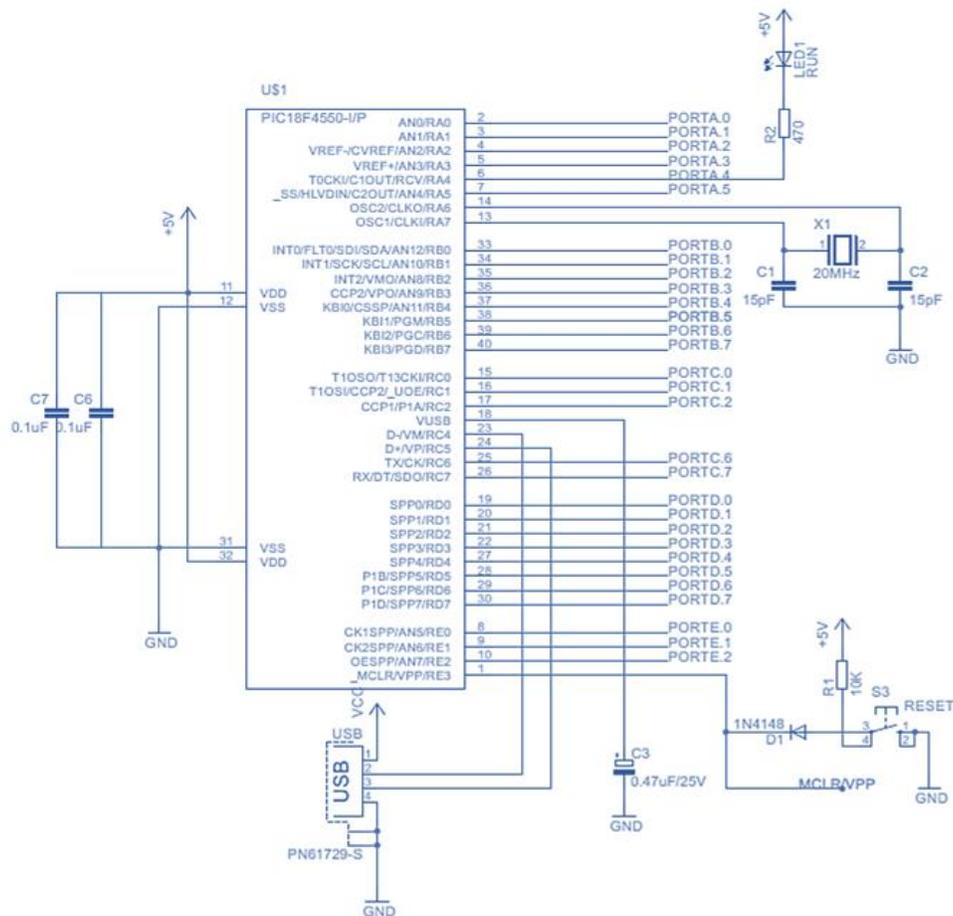


Figura 5.28: Circuito esquemático de la interfaz.

El diagrama esquemático se realizó en Eagle® (**E**asily **A**pplicable **G**raphical **L**ayout **E**ditor), apoyándose en su extensa biblioteca y sobre todo por la facilidad de crear y manipular Diagramas PCB's.

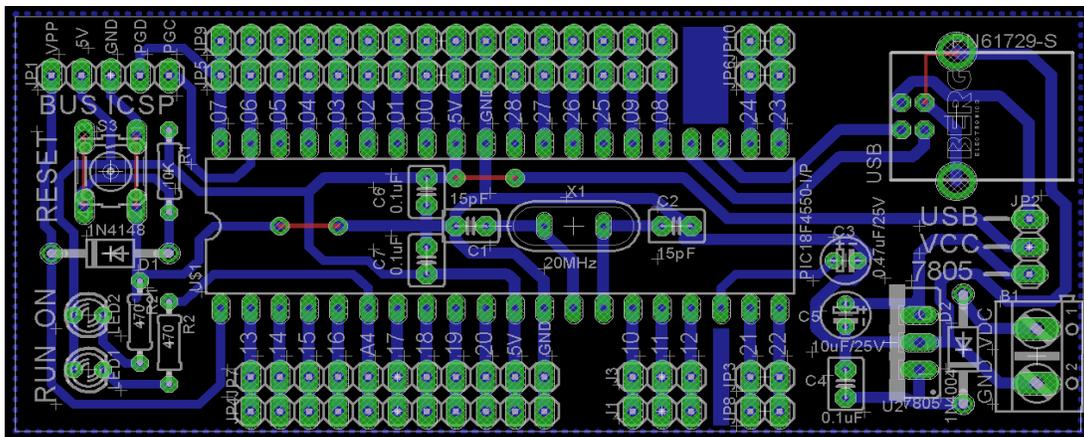


Fig.5.29. Diagrama PCB.

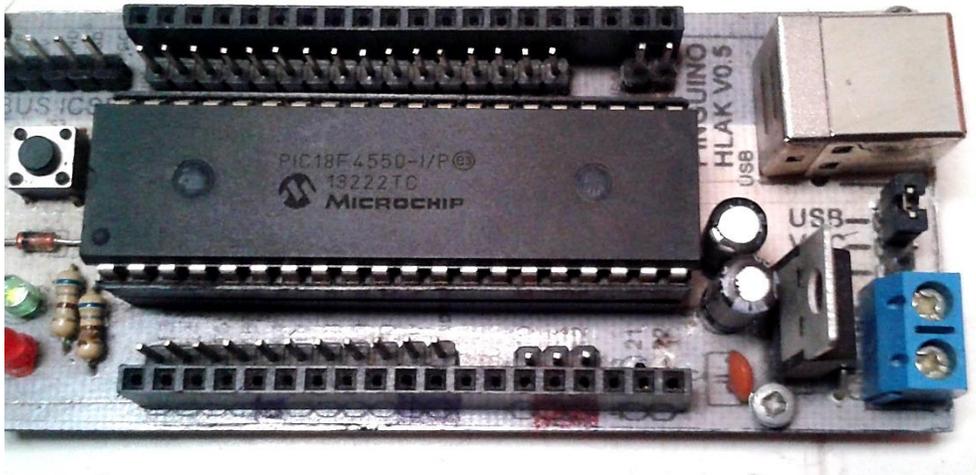


Figura 5.30: Circuito interfaz final.

5.2.2. Circuito acoplador para el láser

Como es sabido los microcontroladores están limitados para manejar cargas eléctricas de forma directa.

La máxima corriente de salida a nivel bajo por el puerto B es 100mA y a nivel alto es 150mA

Estos niveles de tensión permiten trabajar con cargas de bajo consumo como leds, displays o LCD, pero para activar cargas de mayor consumo es necesario utilizar transistores o en el mejor caso optoacopladores.

Los optoacopladores son dispositivos que se componen de un diodo LED y un fototransistor en un solo encapsulado, de manera que cuando se acciona el diodo emita luz, esta irradie el fototransistor y opere en corte o saturación.

La entrada (circuito diodo) y la salida (circuito fototransistor) están completamente aisladas.

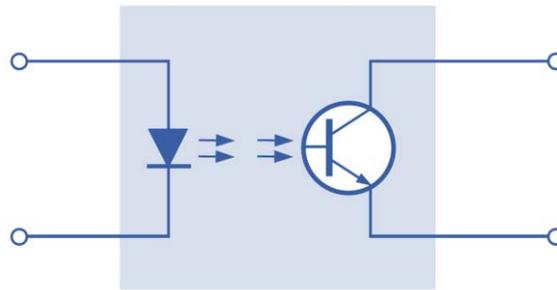


Figura 5.31: Esquema del opto acoplador

Un parámetro clave en los optoacopladores es la CTR (relación de transferencia de corriente). La CTR es una indicación del grado en que una señal es acoplada eficientemente desde la entrada hasta la salida y se expresa como la relación de un cambio de la corriente en el LED al cambio correspondiente de la corriente en el fototransistor. Normalmente se expresa como un porcentaje [9].

La aplicación común de estos dispositivos es para aislar secciones de un circuito que son incompatibles en términos de voltaje o corrientes requeridas. En esta aplicación se trata de corrientes incompatibles.

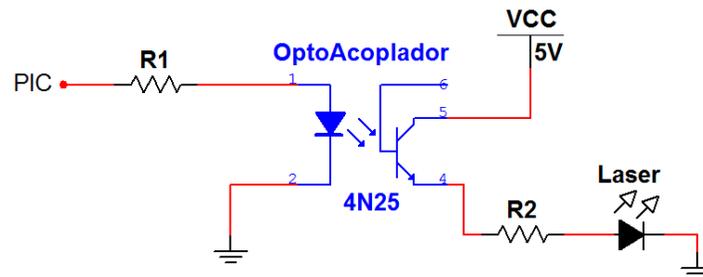


Figura 5.32: Diagrama esquemático del circuito de acoplamiento.

5.2.3. Actuadores

Los servomotores empleados son de tipo digital ambos con un torque de 15 kg, sus engranajes son metálicos, operan dentro de (500-2200) us creando un ángulo de operación de 180° y la velocidad máxima es de $0.18\text{seg}/60^\circ$, la única diferencia se encuentra en el número de baleros, el servomotor de movimiento Pan es de un balero, mientras que el de movimiento Tilt es de dos baleros.



Figura 5.33: Servomotores empleados.

Acoplamiento del sistema de actuación

El sistema de seguimiento consta de dos grados de libertad accionados por servomotores que están debidamente acoplados dando estabilidad al movimiento en cada uno, estos acopladores son de tipo metálico y están diseñados específicamente para los actuadores empleados.

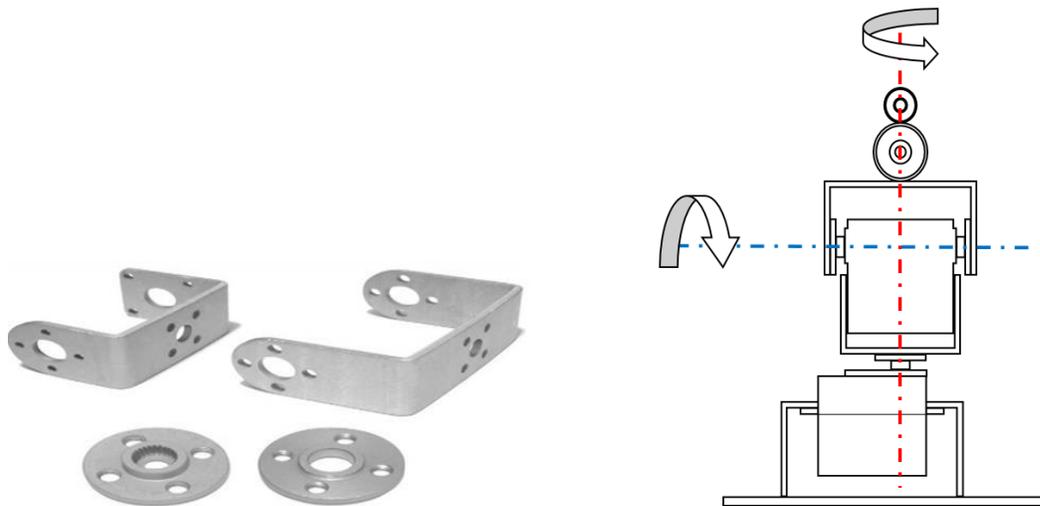


Figura 5.34: Acoplamiento final.

5.2.4. Cámara

Se utilizó un Webcam que tiene integrado sensores de tipo CMOS, alcanzando con optima iluminación una velocidad de 30FPs, el tamaño máximo de captura es de 1280x720 pixeles, así mismo cuenta con comunicación USB.



Figura 5.35: Cámara empleada.

5.2.5. Láser

Como se mencionó es de tipo diodo (semiconductor), con una longitud de onda 660nm (espectro rojo), su máxima potencia es $< 1mW$.



Figura 5.35: Láser empleado.

5.3 EXPERIMENTO

Este objeto en particular es perfecto para el experimento pues sus laterales tienen la misma apariencia, de igual forma las ventanas y las llantas que aparecen en todos los lados, lográndolo robusto a la detección en cualquier rotación que sufra.

También es adecuado para realizar las pruebas de predicción y adaptabilidad, debido a su fácil manipulación.

Por las características de movimiento que posee el objeto, se realizara el experimento en un entorno terrestre y con pocas texturas presentes.

El experimento se desarrollara para demostrar la fiabilidad de los algoritmos propuestos, el cual constará de 5 etapas

1. El objeto atravesará el campo de visión y no será reconocido.
2. Pasará de nuevo el objeto y ahora será reconocido por su movimiento, al mismo tiempo que se crean los bancos de imágenes. Se verificara el aprendizaje a la variación de los valores en los píxeles.
3. Se entrena para ser reconocido como patrón a detectar, comprobando la eficacia en cada modo de entrenamiento.
4. Cuando vuelve a pasar ya es reconocido como patrón y se inicia su seguimiento en función de las características que tenga, proporcionando la ubicación futura, así como su distancia.

Aquí se comprobará la validez de la predicción y su efecto a cambios de dirección inesperados.

5. Se comprobará la adaptabilidad en la velocidad, el tiempo de asentamiento, así como la respuesta a entradas distintas.

5.4 RESULTADOS

5.4.1. Escaneo

Se constató que efectivamente el mapo produce una escena esférica del entorno, el tiempo de escaneo por etapa varía dependiendo de la exposición a la luz que hay, con una buena iluminación el tiempo promedio es de 5 segundos, por lo que un escaneo total utilizando el mismo ángulo de apertura en la cámara, tardó 160 segundos, más el tiempo de posicionamiento inicial y final, se concluye en 3 minutos.

El tiempo es posible reducir, si se conoce que partes de la escena no estará el objeto a detectar, en esta situación la acción sería reducir las secciones de escaneo.

5.4.2. Detección de movimiento

El algoritmo funciona como se pensaba, detecta los objetos en constante movimiento y extrae la ROI del mismo. Sin embargo se detectaron los problemas propios de este algoritmo.

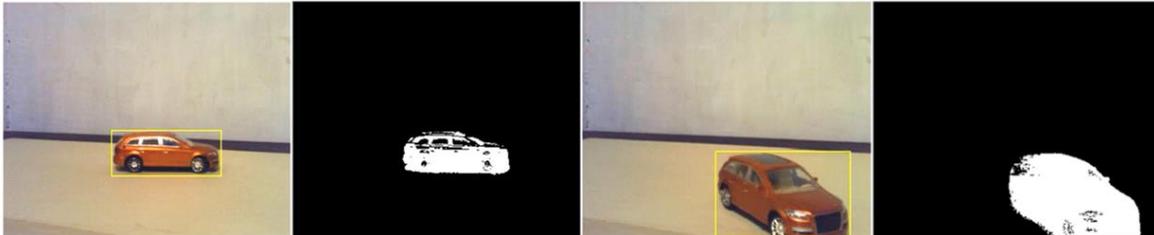


Figura 5.36: Detección de movimiento y objeto dentro del cuadro delimitador (imagen positiva).

5.4.2.1. Tiempo de evolución

El detector de movimiento se configuro a 0.007 en su tasa de aprendizaje, dando como resultado que en 20 frames empezara a considerar pixeles del primer plano como modelo del fondo. Lo anterior a consecuencia de la ubicación estática del objeto, a pesar de esto la dilatación en su estructura binaria permitió que la mayoría de las veces se le considerará como un solo objeto.

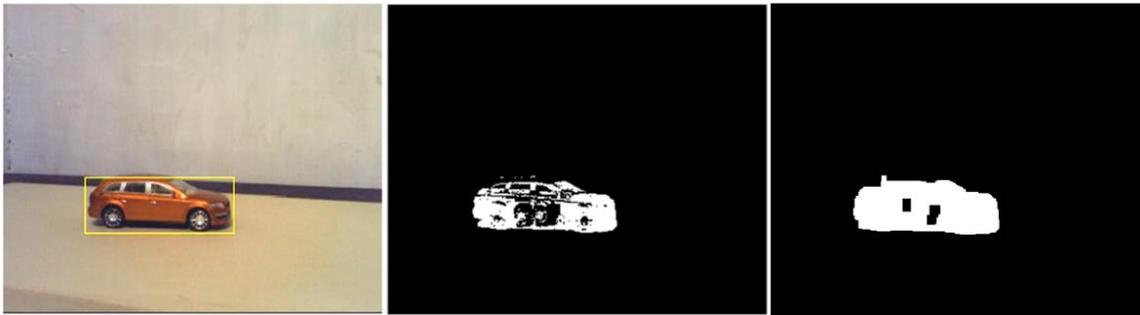


Figura 5.37: Objeto estático con huecos en la estructura binaria y estructura binaria dilatada

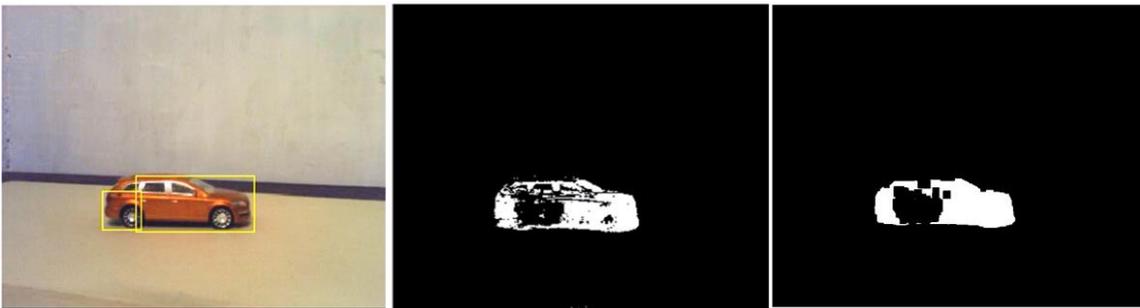


Figura 5.38: Objeto con más tiempo estático y en consecuencia con más huecos produciendo la división de la estructura binaria

5.4.2.2. Falsas detecciones

Estas se crearon utilizando sombras y cambios de iluminación con el propósito de observarlas plenamente, ya que algunas de estas aparecen como reflejos del objeto a consecuencia del suelo liso, estas falsas detecciones aparecieron solo en pocos frames, pues sus distribuciones gaussianas son muy similares a sus anteriores valores, por lo que se adaptaron rápidamente.

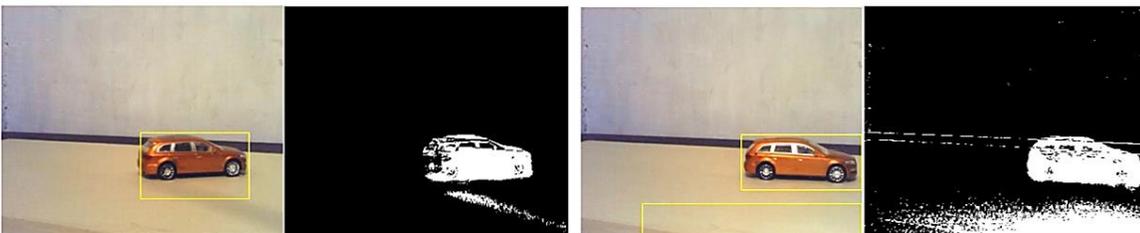


Figura 5.39: Efectos en la adición de sombras y fuentes de luz.

En este experimento se encontró una propiedad que no se había considerado, resulta que cuando en el modelo de fondo, existen distribuciones similares a las que tiene el objeto, tiende a calificarlas como fondo aunque el objeto se encuentre en movimiento, ahora esto se considerará para próximos experimentos.

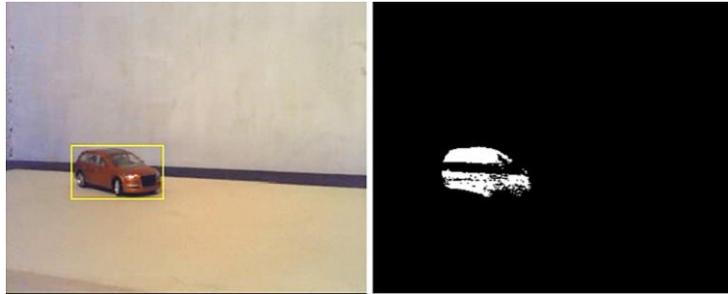


Figura 5.40: Efecto con estructuras de iguales distribuciones gaussianas.

5.4.3. Entrenamiento

Si bien, se ha mencionado que el tipo de entrenamiento depende de las características del objeto, en este experimento el objeto posee cualidades que lo hacen propenso a destacar en ambas. Por tal situación se entrenara con ambas.

5.4.3.1. Elección de características

El objeto tiene suficientes cambios de intensidad, produciendo vectores orientados a esos cambios, se encuentran más acentuados en las llantas, ventanas y el cofre, siendo las ventanas y las llantas las más importantes pues se encuentran en ambos costados del vehículo.

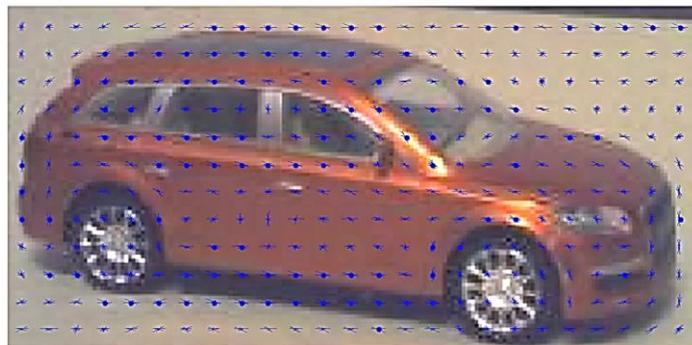


Figura 5.41: Orientaciones de los vectores encontrados en el objeto

De igual manera el objeto cuenta con altos cambios de contraste, lo que favorece la mezcla de características rectángulo, estas al igual encuentran sus mejores correspondencias en las llantas y ventanas.



Figura 5.42: algunas distribuciones de las características rectángulo

-
- Numero de etapas

Se lograron obtener alrededor de 500 imágenes positivas y debido al escaneo del entorno se consiguió alrededor 1000 imágenes negativas, al tener imágenes en la orden de cientos, se convino entrenarlo con 8 etapas y una tasa de falsos baja de 0.2 para cada tipo de entrenamiento.

5.4.3.2. Tiempo de entrenamiento

En ambos entrenamientos se realizó con 450 positivas y negativas por etapa, alcanzando diferentes tiempos en su entrenamiento:

Característica	Tiempo de entrenamiento
Hog	444 segundos
Haar	1138 segundos

Estos tiempos difieren debido a que es computacionalmente más difícil extraer las características rectángulo, sin embargo son más fructuosas cuando el objeto es apropiado, por el contrario las características de vectores orientados se extraen con rapidez consiguiendo un entrenamiento relativamente rápido.

5.4.4. Detección de patrón

Este tipo de detección se realiza utilizando un patrón a seguir, este patrón es resultado de los anteriores entrenamientos. El detector cascada se implementó con un umbral de 25 en la detección para cada entrenamiento, obteniendo resultados diferentes.

5.4.4.1. Veracidad en la detección (Tipo Haar)

Se comprobó que el entrenamiento tipo Haar para este objeto en particular no es apropiado, si bien tiene características favorables, estas no son suficientes para la minuciosa detección que necesita la aplicación.

5.4.4.2. Detección en diferentes zonas de mapeo

Se comprobó que el sistema logro detectar el objeto en todas las zonas entrenadas, sin embargo se encontraron inconvenientes.

(A-B-D), muestran que efectivamente detectan el objeto, no obstante muestran algunas falsas detecciones alrededor del objeto, que pueden eliminarse con el aumento del umbral de detección. (C), se puede apreciar que siendo una escena sin objeto, el detector concibe que existe un objeto, (Falso Positivo), el cual es perjudicial para el sistema, al sistema no se le puede permitir perseguir un objeto que no es el de interés. (E), es una correcta detección (Verdadero Positivo). (F), se puede apreciar que existen dos errores, el vehículo esta y no lo detecta (Falso Negativo) y también ocurre

una detección errónea sobre una sombra con la que no fue entrenada (Falso Positivo). (G-H), en estos frames si detecta el objeto junto con falsas detecciones, además detectan un objeto erróneo.

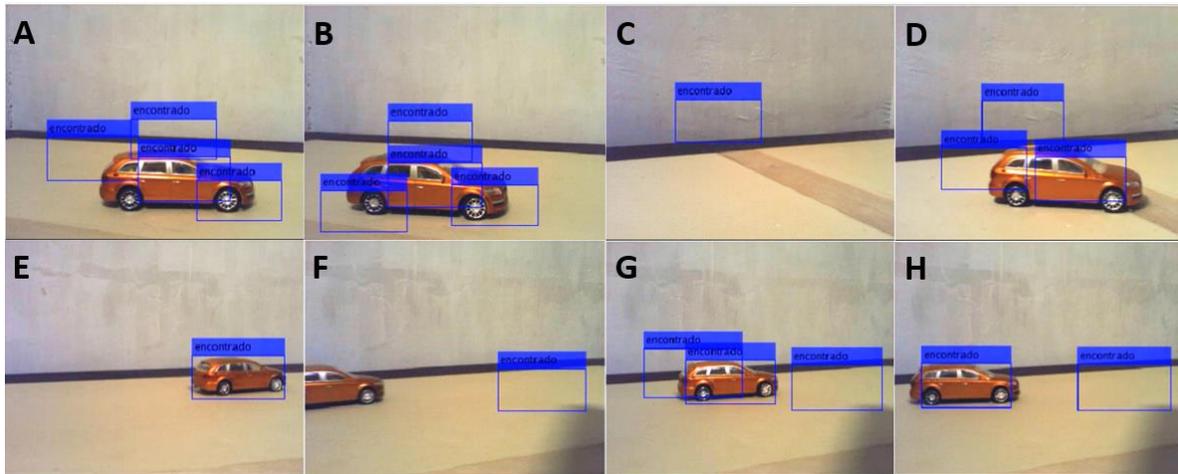


Figura 5.43: Detección de objeto en diferentes zonas con entrenamiento tipo Haar.

Se comprobó que la distancia mínima y máxima, efectivamente dependen de la configuración en el área mínima y máxima del detector, estas se configuraron como:

Min Área = [50, 50] alcanzando 300 cm en la máxima distancia de detección (A).

Max Área = [200, 200] alcanzando 15cm en la mínima distancia de detección (B).

Sin embargo al igual se encontraron falsas detecciones.

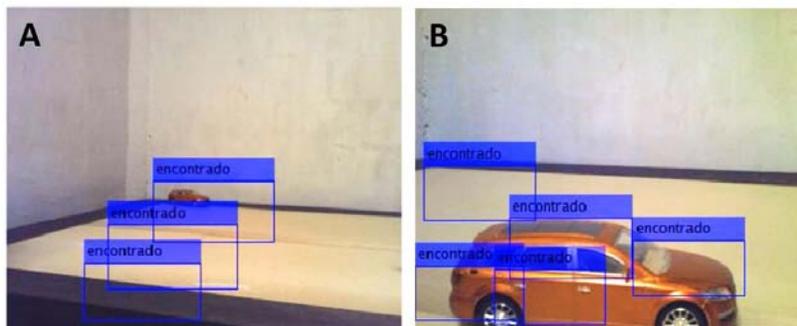


Figura 5.44: Máxima y mínima distancia de detección (Haar).

5.4.4.3. Rotación del objeto

Conociendo que las muestras positivas constaron de solo laterales del objeto se puede intuir que el detector solo se encontrará esas laterales, sin embargo por lo explicado en un inicio el detector logro encontrar más lados no entrenados.

De igual forma se encontraron falsos positivos, (A) lado frontal, (B) frontal lateral derecho, (C) frontal lateral izquierdo, (D) lado trasero. (E) lateral derecho, (F) posterior lateral derecho, (G) parte superior, en apariencia en ella no existe relación con los laterales, sin embargo por la inclinación de la cámara y la del propio objeto es posible detectar partes de él. (H) parte inferior, al igual que el anterior no se entrenó con este lado, de igual manera detecta solo algunas partes del objeto. Esto es producto de la asignación “características rectángulo”.

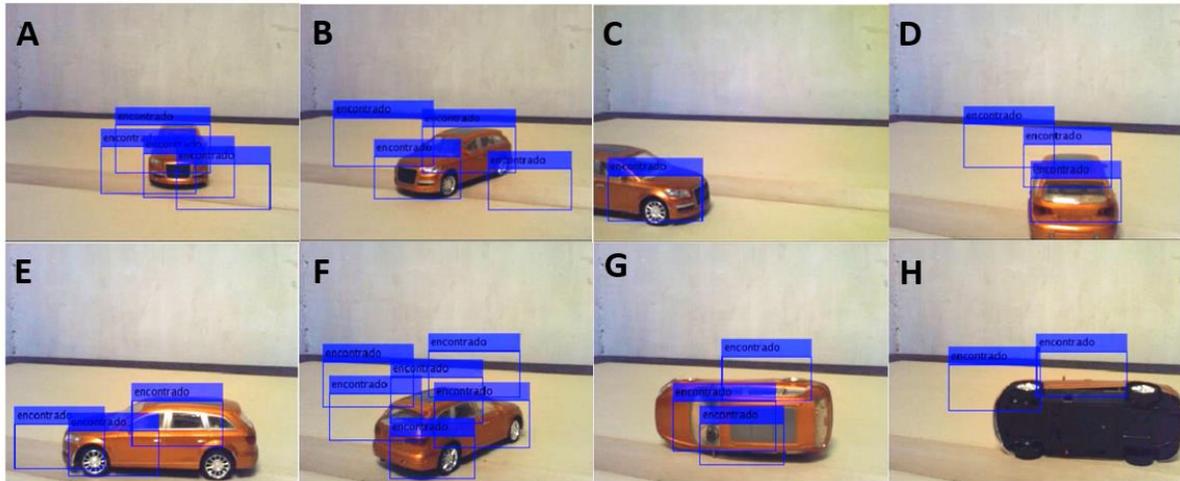


Figura 5.45: Detección en diferentes rotaciones del objeto (Haar).

Existen diversas maneras para perfeccionarlo en esta modalidad de detección:

- Modificar el entrenamiento: aumentar las muestras positivas al igual que las negativas, aumentar la tasa de verdos positivos en cada etapa etc.
- Modificar el detector: aumentando el umbral de detección

5.4.4.4. Veracidad en la detección (Tipo Hog)

Esta manera de detección comprobó ser la más adecuada, los vectores orientados son muy particulares para cada objeto, sin embargo objetos con vectores orientados similares tienden a confundir el detector, partiendo de esto se hizo patente que para este objeto en específico, en este tipo de escena la detección fue muy eficiente.

5.4.4.5. Detección en diferentes zonas de mapeo

El objeto fue encontrado en todas las zonas incluso en las no entrenadas.

(A-B-C-G) se detectó correctamente en estas zonas mapeadas. (D), es una zona que tiene partes no mapeadas, en ella se puede apreciar que confundió parte del cable y una sombra (Falso Positivo),

así mismo aparece un (Falso Negativo). (G-F) se trata de zonas con partes no mapeadas donde el detector no confundió los vectores orientados. (H) en apariencia ocurre un falso positivo, pero no se le puede considerar como tal ya que es consecuencia de la oclusión parcial en el campo de visión.

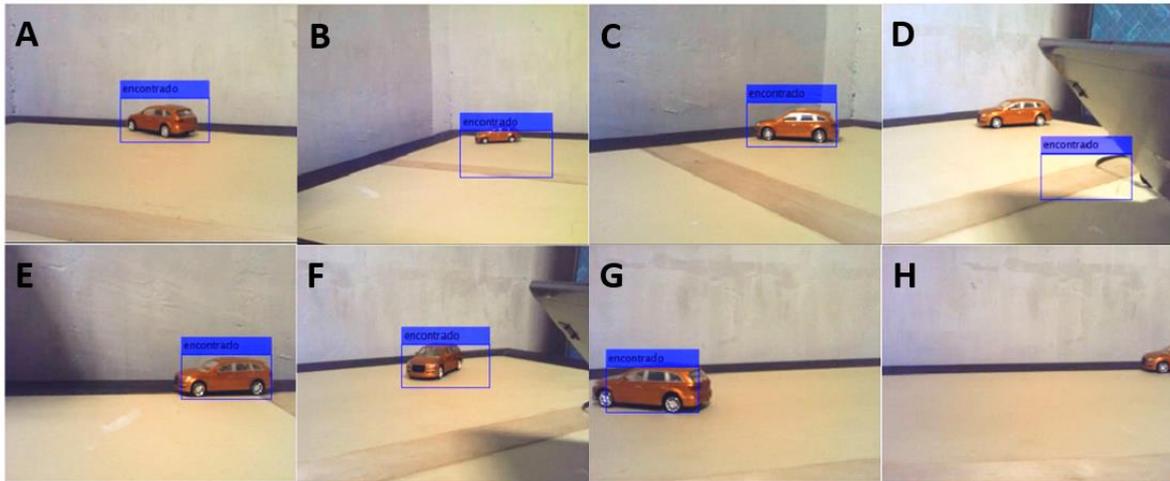


Figura 5.46: Detección de objeto en diferentes zonas con entrenamiento tipo Hog.

La mínima y máxima distancia de percepción que se logró, fue de igual forma que la detección con características tipo Haar, pero en este caso las detecciones fueron correctas (A-B).

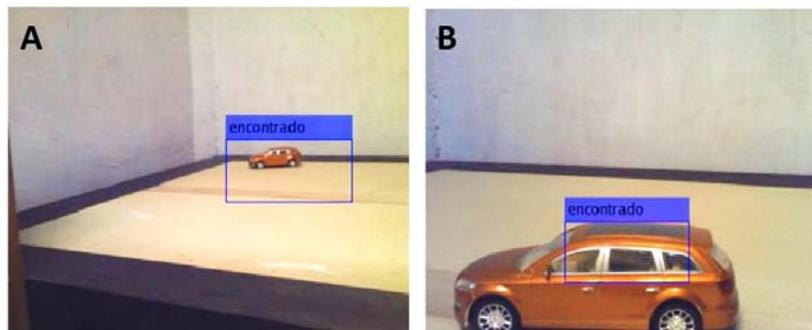


Figura 5.47: Máxima y mínima distancia de detección (Hog).

5.4.4.6. Rotación del objeto

Al igual que las características de tipo Haar, existen vectores orientados de algunas partes del objeto, que se repiten en todos los lados del objeto, ofreciendo robustez a las rotaciones experimentadas.

(A) Frontal lateral derecho, es detector encontró dos (Verdaderos Positivos). (B) lateral derecho. (C) lateral izquierdo. (D) parte inferior, este modo de detección no logró coincidir los vectores orientados con respecto a ese mínimo lateral visible, así mismo no detecto nada pues no se entrenó con esa parte. (E) frontal. (F) trasero. (G) frontal lateral izquierdo. (H) parte superior, aquí solo detecta un lateral que percibe.

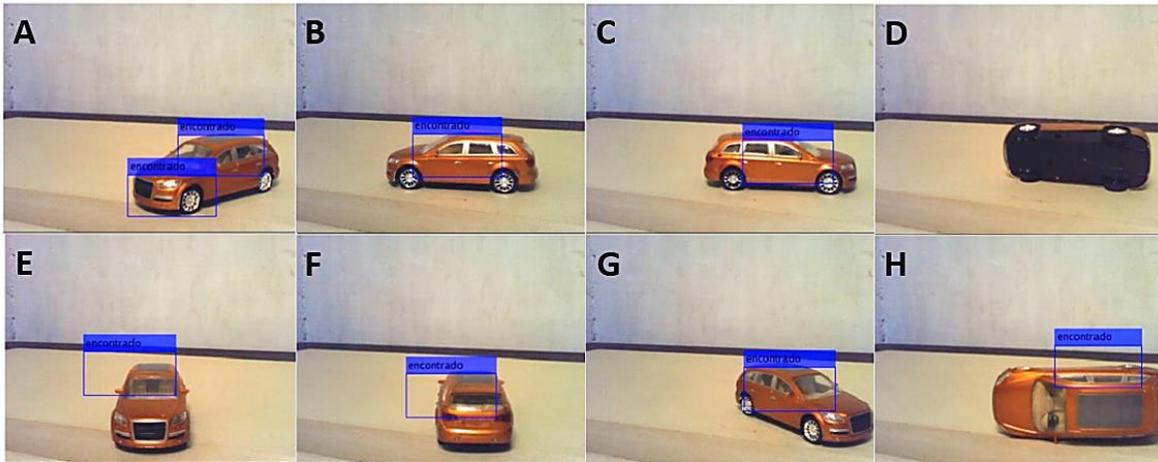


Figura 5.48: Detección en diferentes rotaciones del objeto (Hog).

5.4.5. Seguimiento

5.4.5.1. Seguimiento de esquinas

Como se ha indicado este tipo de seguidor, primero extrae las esquinas más prominentes y a continuación busca la correspondencia de las esquinas vecinas en las mismas ubicaciones del siguiente frame; en base a la eficacia en la detección por vectores orientados, se hace evidente que este tipo de seguimiento es el más apropiado.

Esta secuencia de imágenes demuestra el correcto seguimiento de las esquinas, el número de frame está indicado en la parte superior.

Inicialmente extrae las esquinas del objeto detectado. (10) en este frame se observa que las esquinas más débiles se perdieron, las fuertes permanecen en las ventanas, parte trasera y llanta. (20) el objeto continuo en movimiento y las esquinas continuaron correspondiendo con las originales. (30) el objeto alcanza una zona que no fue entrenada, sin embargo por seguir las esquinas este algoritmo no está interesado en la escena en la que se encuentre. (40) si bien el objeto sufrió una traslación, la cámara percibe una rotación respecto a ella, por tal motivo ahora visualiza una parte posterior, sin embargo aún encuentra la coincidencia. (50-70) en esta nueva rotación percibida por la cámara se observa que algunas esquinas fueron perdidas por efectos de oclusión de la parte trasera. (80-90) en estos frames continua siguiendo las esquinas que aún le quedan en la llanta. (100) aquí ha perdido

todas las esquinas, dando inicio al pronóstico del movimiento, al mismo tiempo que inicia la nueva detección del objeto, la última imagen muestra que ya lo detecto y extrajo nuevas esquinas a seguir.

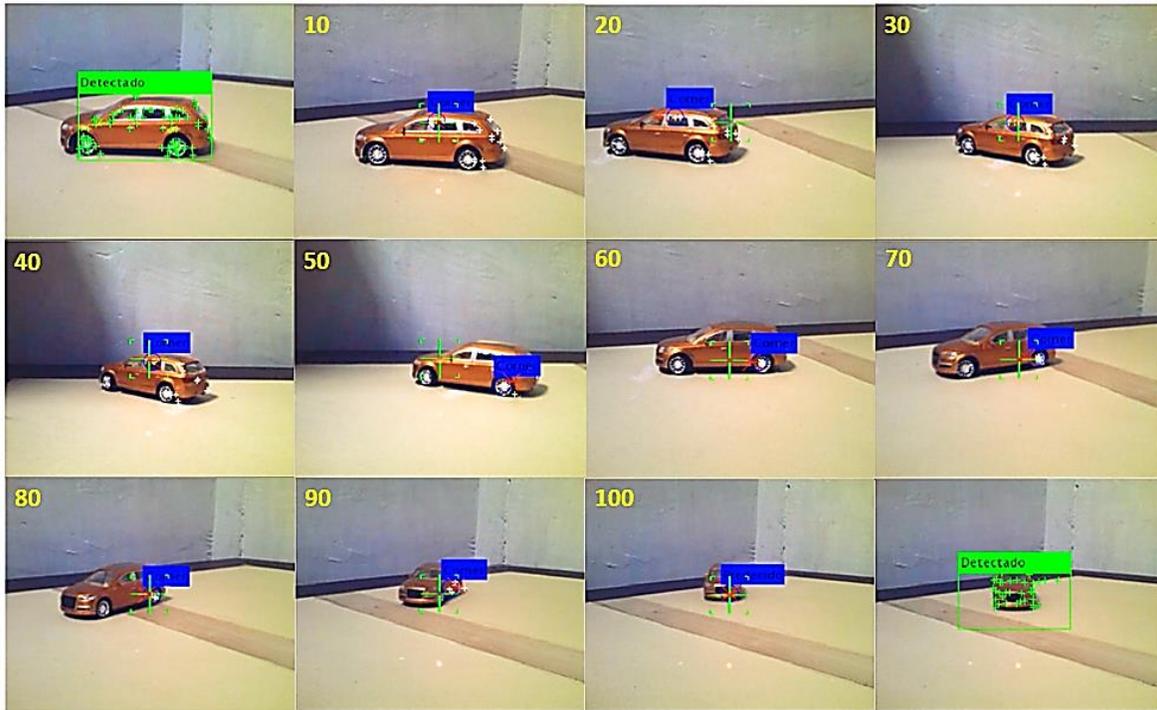


Figura 5.49: Seguimiento de las esquinas del objeto.

Durante el seguimiento es claro que algunas de estas esquinas se pierdan por la rotación del objeto, más aún si existen perturbaciones en el ambiente o la velocidad de captura es menor a la del movimiento.

Con el fin de verificar la confiabilidad en su seguimiento este seguidor fue sometió a diversas situaciones.

5.4.5.2. Cambios bruscos de iluminación

Se comprobó que en cambios de iluminación indirectos, el sistema pierde las esquinas localizadas en las texturas reflejantes, sin embargo en las texturas opacas los gradientes tiene similares valores consiguiendo que el sistema aún encuentre coincidencia en las esquinas (C). Cuando la iluminación es directa el valor en los pixeles de intensidad son alterados, concluyendo en la pérdida de las esquinas (D). (A-B) detección y seguimiento con iluminación controlada.

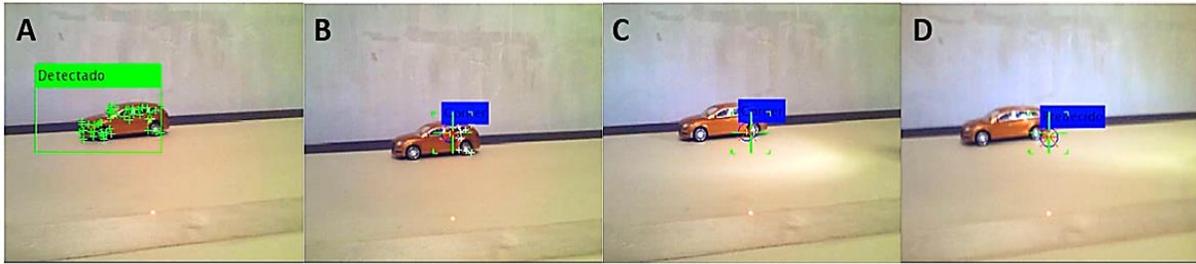


Figura 5.50: Efectos del seguimiento de esquinas a diferentes cambios de iluminación.

5.4.5.3. Velocidad de seguimiento

Como se mencionó esta velocidad es relativa a la distancia del objeto respecto a la cámara, lo que mide el sistema es la velocidad instantánea en píxeles por frame.

A si mismo esta velocidad es relativa pues el sistema compensa en todo momento, forzando a mantener el objeto en el centro de la imagen.

El experimento demostró que el desplazamiento máximo que puede percibir la cámara en píxeles sin perder las esquinas del objeto es de 119.8, a 150cm de él, lo que significa que el objeto estuvo a 50 píxeles de salir del campo de visión.

La secuencia de imágenes muestra el desplazamiento horizontal del objeto:

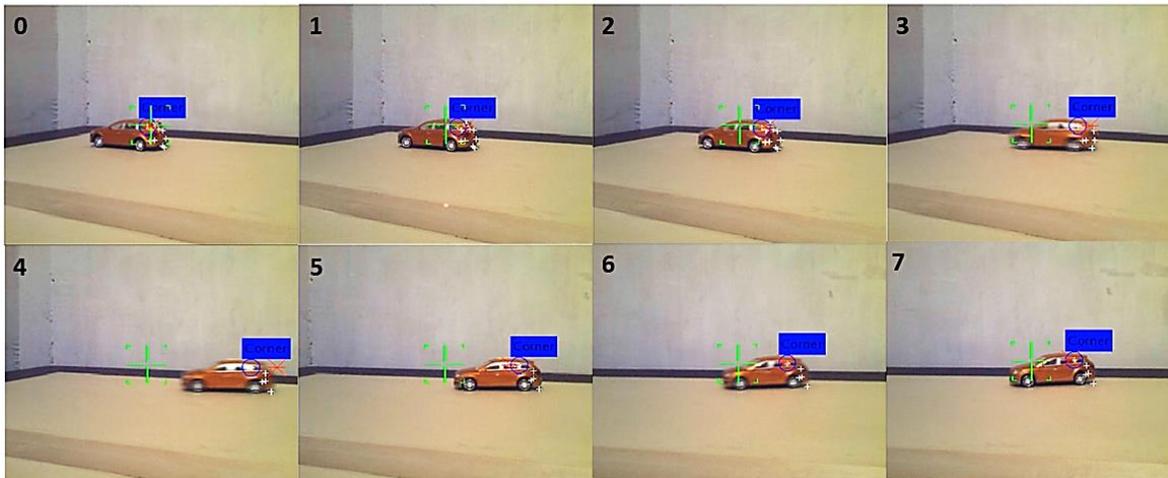
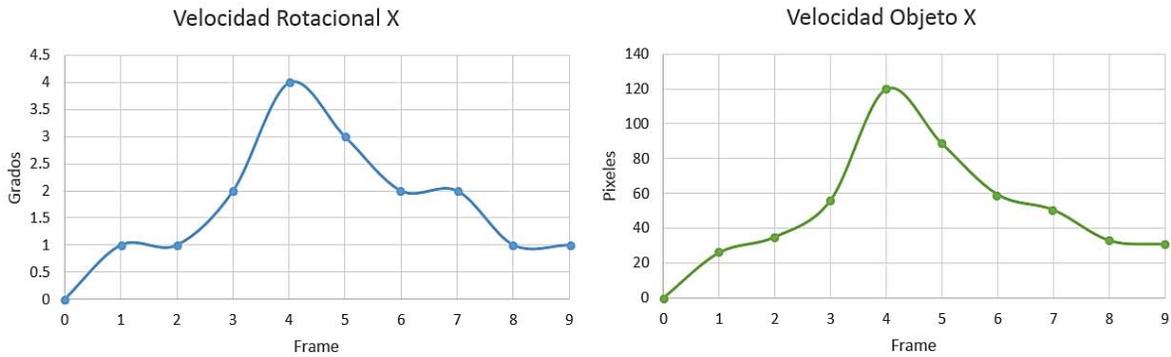


Figura 5.51: Frames consecutivos (0-9) de un movimiento fortuito.

Estas dos graficas muestran las velocidades que alcanzan, el cuarto frame alcanza la máxima velocidad tanto el objeto como el actuador.



Grafica 5.1: Comparativa cambio brusco de velocidad.

5.4.5.4. Seguimiento de color

Este método de seguimiento demostró no ser tan eficiente para este objeto en esta escena en particular.

Esta técnica como se ha explicado sigue el espectro que rechazan los pigmentos de la textura del objeto (el color del objeto), por esta situación cuando cambia la fuente de iluminación tiende a variar el color que percibe la cámara, este algoritmo hace frente a esta situación adaptándose a los cambios de incidencia que sufre el objeto.

Partiendo de lo anterior se comprobó en el experimento que esta adaptabilidad tiene consecuencias adversas sobre todo cuando la cámara utilizada ajusta el lente a la exposición de la iluminación, pues esta particularidad concentra la iluminación en los píxeles centrales.

Como se puede observar en (A) el objeto es fácilmente diferenciable del entorno así mismo se aprecia una penumbra que es consecuencia de la iluminación; en (B-C) el objeto se mueve hacia esta penumbra y se puede observar que la sombra que proyecta el vehículo es similar a esta penumbra, en (D) el objeto entra a esta penumbra y no es fácilmente identificable, en esta circunstancia el CAMShift comienza a adaptar el histograma que inicialmente se le asignó, al mismo tiempo reduce el porcentaje de confiabilidad, concluyendo en dos posibles situaciones: la primera es que este porcentaje sea tan bajo que lo de por perdido, la segunda es que no se a tan bajo y comience a seguir este nuevo histograma; (E-F) el seguimiento continua y la cámara reacciona ajustando la exposición a la iluminación.

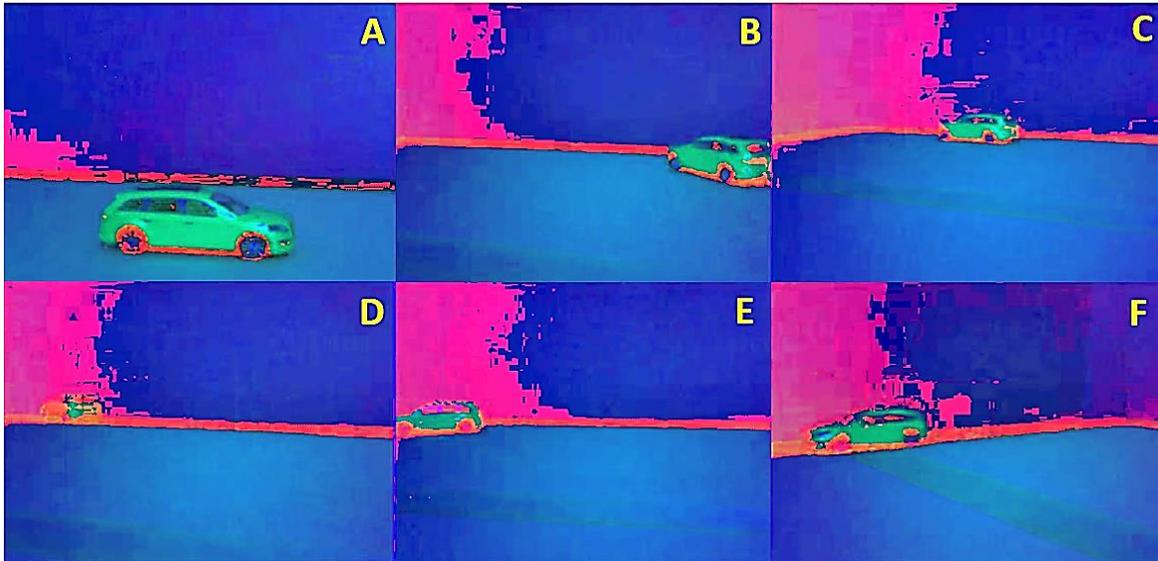


Figura 5.52: Secuencia de frames en HSV donde la cámara ajusta el lente a la exposición lumínica.

5.4.5.5. Cambios de iluminación

El cambio indirecto y directo de iluminación afecta el seguimiento del histograma, sobre todo cuando la superficie del objeto es lisa, como lo es el objeto en este experimento.

Esta secuencia de imágenes demuestra la pérdida del objeto al cambio brusco de iluminación.



Figura 5.53: Efecto del seguimiento de color a diferente cambio de exposición.

5.4.5.6. Velocidad de seguimiento

El experimento demostró que el desplazamiento máximo que puede percibir la cámara es de 120 píxeles sin perder el histograma del objeto, a 100 cm de él.

La secuencia de imágenes muestra el desplazamiento horizontal ocurrido.

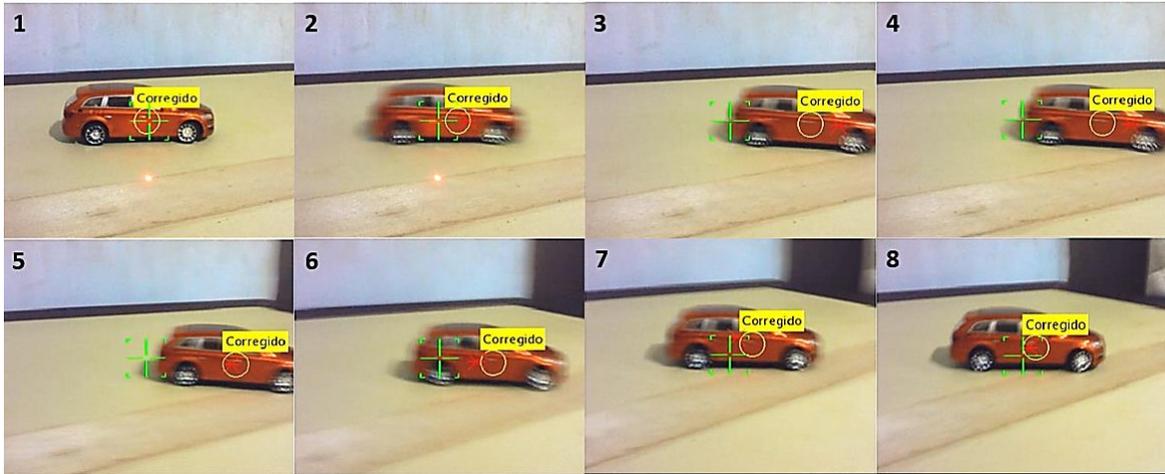
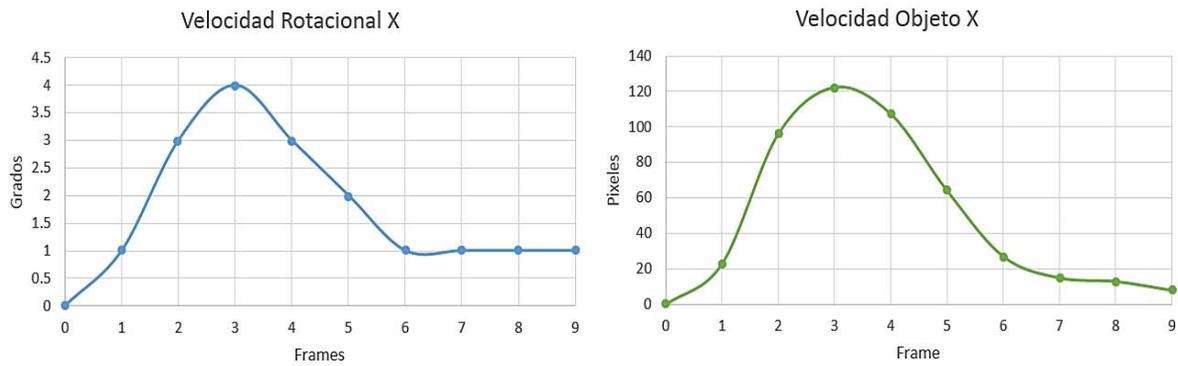


Figura 5.54: Frames consecutivos (0-9) de un movimiento fortuito.

Estas dos graficas muestran la velocidad maxima que alcanzan, ocurre en el tercer frame tanto del objeto como del actuador.



Grafica 5.2: Comparativa cambio brusco de velocidad.

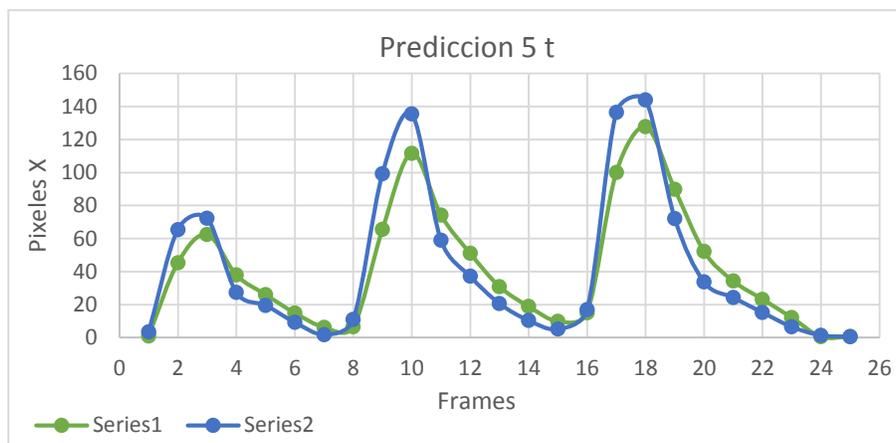
5.4.6. Predicción

Conociendo que el sistema predice el desplazamiento en pixeles sobre una imagen dinámica y que al mismo tiempo este desplazamiento se compensa con las ganancias del control adaptable, no es posible interpretar la predicción sin comprender la acción del control.

A pesar de ello, en este apartado solo se explicará la validez de la predicción futura y la estimación en oclusión.

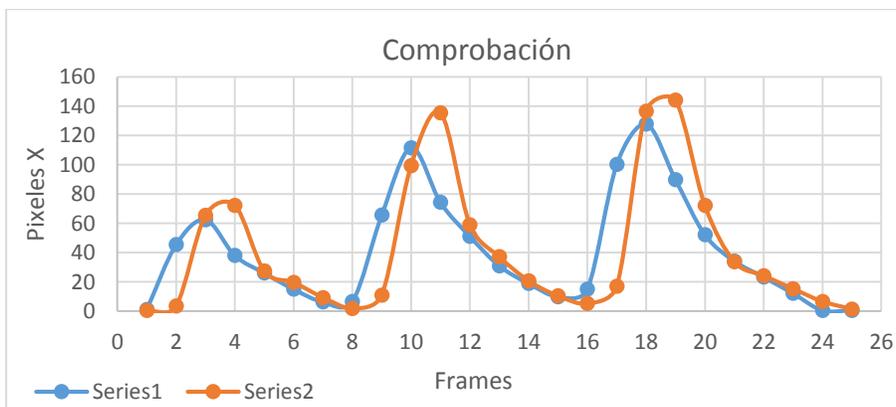
En base a experimentos se comprobó que la predicción más confiable, en la que converge en un error mínimo, es a 5 t, se hace de esta manera pues los intervalos de muestreo no son en tiempos regulares y además la variancia (error) de la posición no es posible identificarla debido a que las ubicaciones del seguimiento cambian por efectos de perspectiva de iluminación.

La grafica muestra 3 desplazamientos del objeto, cada uno con diferentes velocidades, cada línea de predicción termina al cambiar el sentido de desplazamiento, este sentido cambia a consecuencia de la acción de control, el sistema continuamente predice por tal razón al cambiar el sentido crea un error antes que se actualice la nueva posición.



Grafica 5.3: Desplazamiento real contra predicción del desplazamiento futuro.

En esta grafica se demuestra que efectivamente predice sobre las líneas de desplazamiento y los errores ocurren en el cambio de sentido.

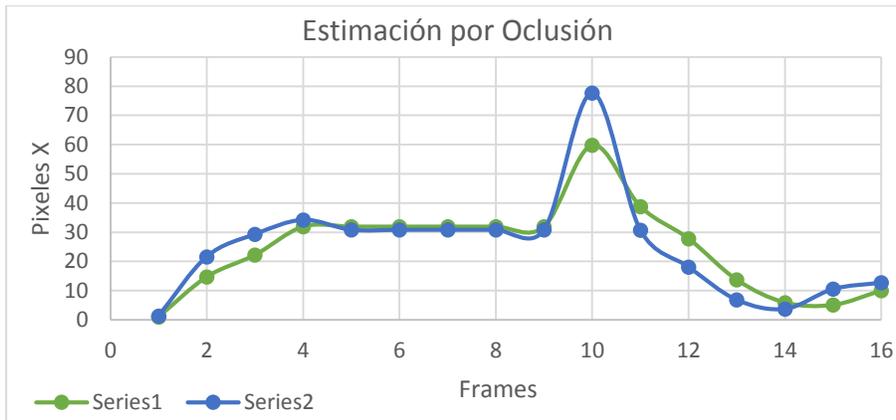


Grafica 5.4: Desplazamiento real contra predicción adelantada.

5.4.6.1. Estimación por pérdida

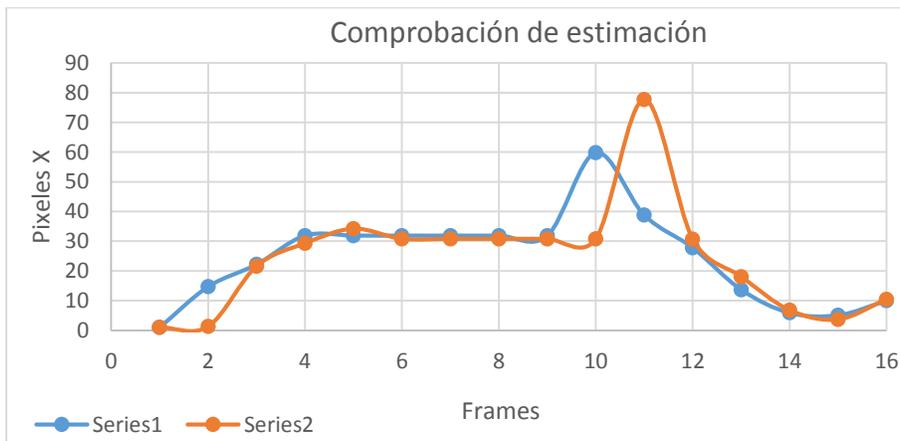
El algoritmo de Kalman como se ha estudiado, tiene habilidad de estimar la ubicación del objeto cuando sea perdido o por efectos de oclusión.

Para conocer su eficacia de estimación, se forzó la pérdida del objeto, la gráfica muestra este vector de estimación, el objeto se extravió en el cuarto frame y apareció en el décimo, se aprecia esta línea continua de estimación entre el (4-9) frame, así mismo se aprecia que en esta situación de estimación el sistema automáticamente finaliza la predicción futura; en el décimo frame se obtiene la nueva ubicación de objeto, corrigiendo la trayectoria.



Grafica 5.5: Desplazamiento real contra predicción del desplazamiento futuro.

En esta grafica se demuestra la valides de predicción durante el proceso de estimación.



Grafica 5.6: Desplazamiento real contra predicción adelantada.

5.4.7. Velocidad adaptable

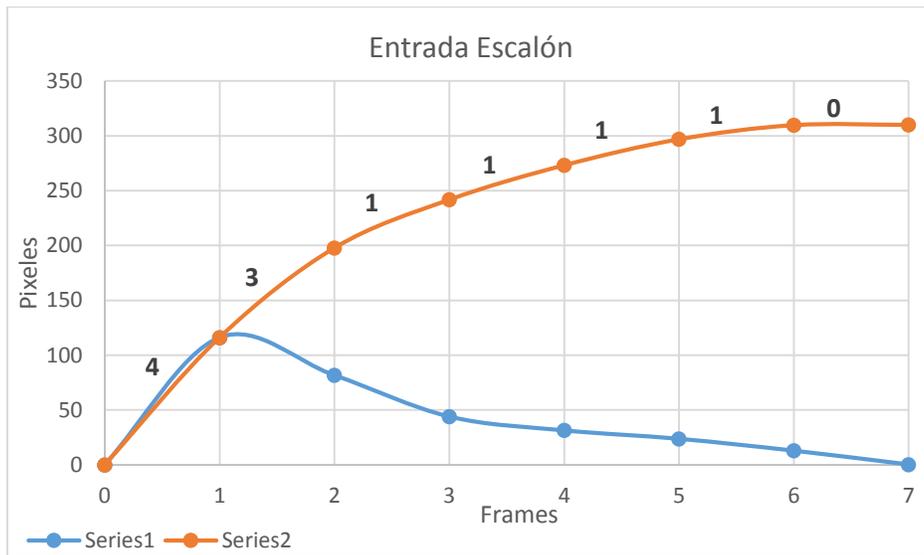
Como se precisó en la implementación este tipo de control requiere de una tabla de ganancias programadas y estas a su vez se especifican de acuerdo a la respuesta requerida y para esta aplicación la respuesta deseada es sobre-amortiguada.

Los resultados prueban que efectivamente la respuesta es sobre-amortiguada para todas las entradas de prueba. Estas respuestas se obtuvieron experimentalmente.

Todas las tablas muestran dos graficas: la primera indica la respuesta al desplazamiento total, y en la segunda se aprecia la respuesta percibida desde la cámara.

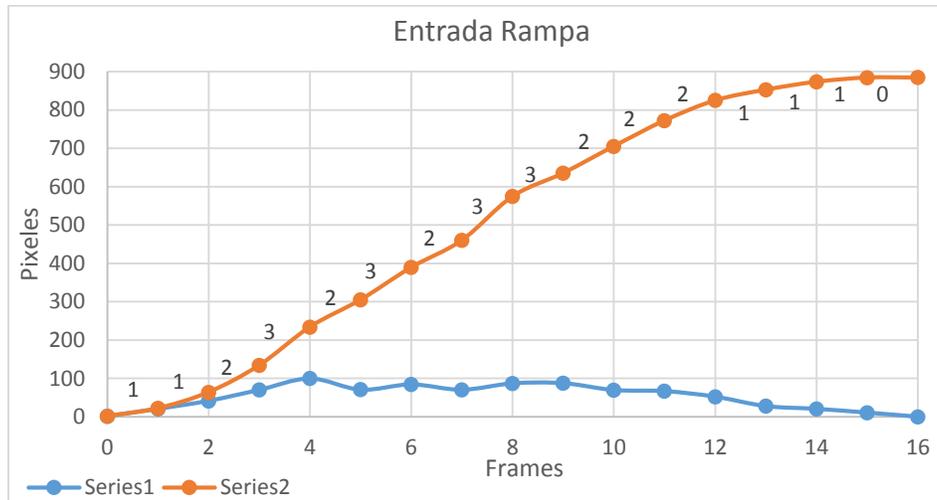
La entrada escalón tiene una entrada de 310 pixeles (desplazamiento total), el sistema responde con distintos desplazamientos de acuerdo a la posición del objeto respecto al centro de la cámara, a consecuencia de este movimiento “sutil”, no ocurre ningún sobre-paso en la entrada.

El tiempo de asentamiento es de 6 frames, conociendo que a velocidad de captura esta entre (5-10) frames por segundo, se puede considerar el tiempo de asentamiento menor a 1 segundo.



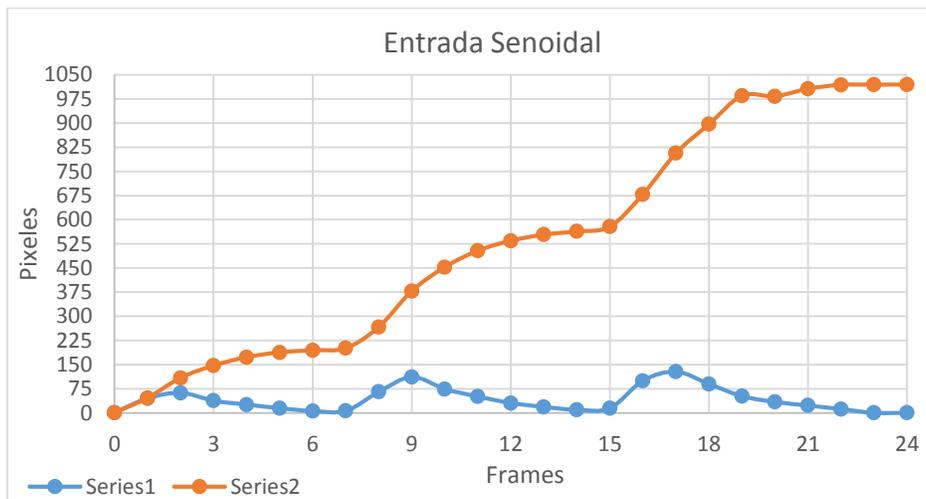
Grafica 5.7: Desplazamiento total contra desplazamiento percibido.

La entrada rampa es creciente y el sistema responde manteniendo los desplazamientos constantes. El desplazamiento total es de 885 pixeles y se realizó en 15 frames.



Grafica 5.8: Desplazamiento total contra desplazamiento percibido.

La entrada senoidal consta de tres desplazamientos fortuitos cada uno con diferente desplazamiento máximo, el primero ocurre en 194.4 pixeles y su tiempo de asentamiento es de 6 frames, el segundo alcanza 369.1 pixeles y lo alcanza en 8 frames, el tercero es de 455.8 pixeles y llega a este valor en 9 frames. Cada desplazamiento termina e inicia en estado estático (centrado en la cámara).



Grafica 5.9: Desplazamiento total contra desplazamiento percibido.

Si bien, la transición entre ganancias es el problema común en este tipo de control, no se vio afectada la respuesta debido a la relativa “rapidez” de muestro.

5.4.8. Triangulación láser

Se realizaron mediciones comprobando la exactitud del instrumento, estas mediciones son a intervalos regulares.

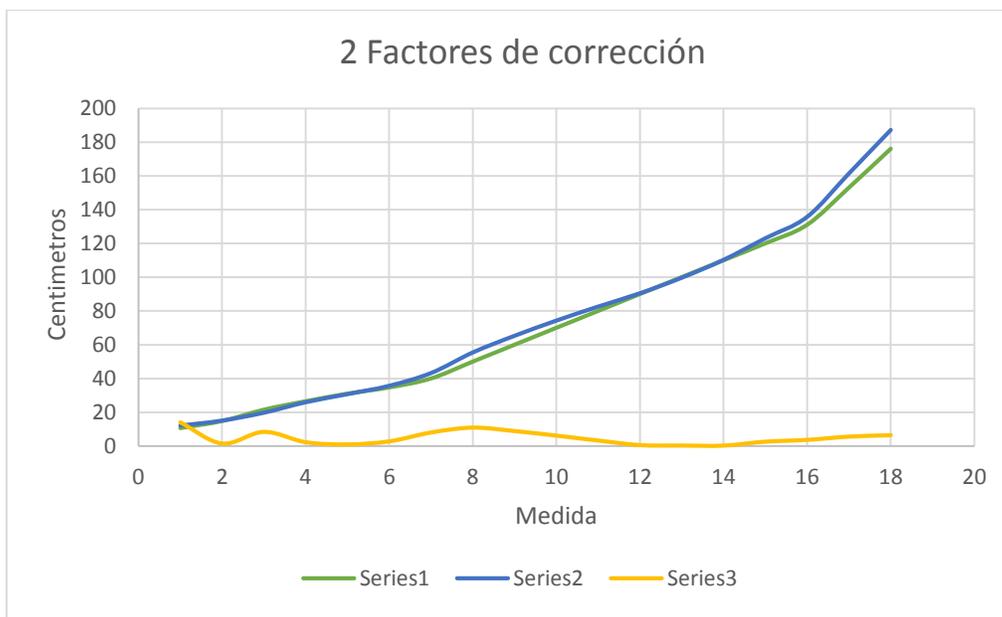
Los errores asociados a las medidas se caracterizan observando su exactitud y precisión.

La exactitud se refiere a que tan cercano está el valor calculado con respecto al valor real y la precisión se refiere a que tan cercano esta un valor individual calculado respecto a otros.

El error comprueba la desviación que se produjo en el proceso de linealización, este dato se obtuvo calculando el error relativo porcentual en cada medición, la operación consiste en normaliza el error respecto al valor real, se consigue con:

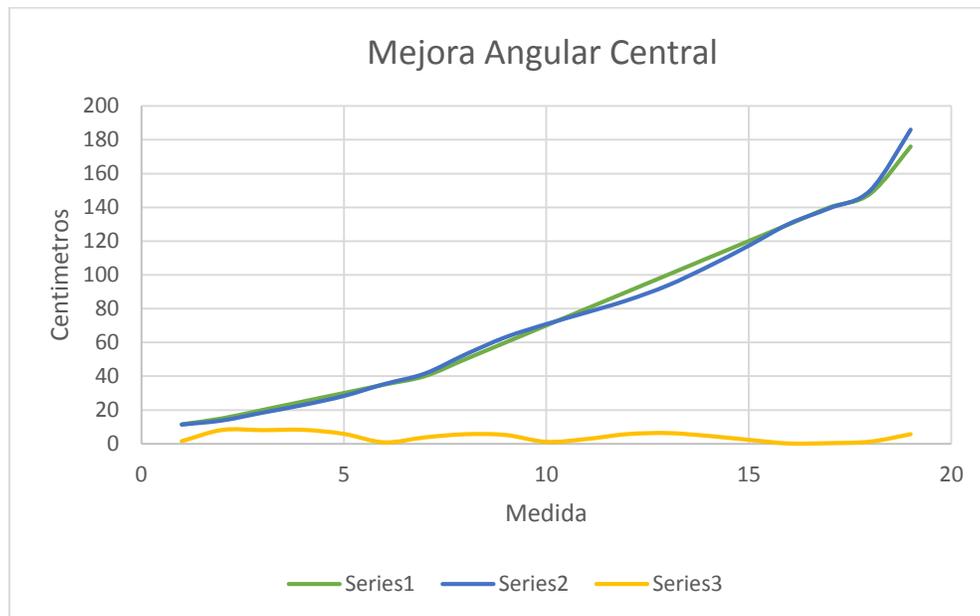
$$\varepsilon_t = \left(\frac{\text{Medida Real} - \text{Medida calculada}}{\text{Medida Real}} \right) 100\% \quad (5.28)$$

Esta grafica muestra la relación entre el la medida real y la calculada cuando se utilizan dos factores, se aprecia la creciente desviación alrededor de los 100cm, así mismo presenta poca exactitud ya que solo la mediciones cercanas a inicio y al centro son realmente fiables, el promedio del porcentaje de error relativo es de 4.788%, su precisión es de 44% respecto a todas las mediciones partiendo del promedio.



Gráfica 5.10: Medida real contra media calculada y error relativo porcentual.

Al utilizar la mejora en la doble linealización se puede apreciar el “serpenteo” a rededor de las mediciones reales, el máximo error se encuentra en los 100cm producto de la reducción del ángulo de apertura a esa distancia, se sacrificó esa exactitud que existía a esa distancia con el fin de obtener más exactitudes fiables a diferentes distancias, el promedio del porcentaje de error relativo es de 4.077%, su precisión es de 52% respecto a todas las mediciones partiendo del promedio.



Grafica 5.11: Medida real contra media calculada y error relativo porcentual.

CONCLUSIONES

Los objetivos planteados en un inicio se llevaron a cabo obteniéndose resultados muy convincentes.

Los resultados demuestran que el seguimiento y predicción de cualquier objeto móvil es posible, así como la adquisición de su ubicación espacial en movimiento.

La detección del objeto en movimiento resultó ser la mejor opción para suplir de imágenes positivas y negativas, no obstante encuentra inconvenientes en un entorno dinámico, sin embargo es posible disminuir la tasa de aprendizaje con el fin de obtener una adaptabilidad más rápida eliminando casi al instante el movimiento involuntario del entorno.

El entrenamiento para la detección de patrones mostró ser bastante confiable, lo único que necesita es un analizador de imágenes que decida qué modo de entrenamiento conviene más.

Este analizador de imágenes de la misma manera puede ser utilizado para decidir el tipo de seguimiento, con esta mejora, los seguidores aumentarían su eficacia y entregarían un vector de desplazamiento más consistente.

Se ha mencionado que la predicción con cámara dinámica depende en gran medida del desplazamiento mecánico, que es producto de la compensación, a pesar de ello la predicción futura mostró ser bastante confiable.

Este desplazamiento mecánico adaptable, aunque fue programado con pocas ganancias demostró seguir a la par la velocidad del objeto.

La triangulación láser demostró ser lo suficientemente correcta para obtener la lejanía del objeto, aunque la configuración implementada no fue a ideal cumple con los requerimientos planteados en un inicio.

Con lo que se concluye que el sistema de derribo puede ser implementado con algunos perfeccionamientos en el sistema de seguimiento.

MEJORAS FUTURAS

Mejorar el filtro de Kalman, para que pueda predecir la ubicación futura en el espacio.

Cambiar el sistema de control a un control adaptativo autoajustable

Sustituir la cámara por otra que ofrezca una adquisición de imágenes más eficiente en cuanto a los cambios bruscos de iluminación, así mismo que no esté limitada a la comunicación USB

Aumentar la velocidad de procesamiento migrando a procesadores en Paralelo (GPU) utilizando el lenguaje CUDA y OPENCV

Utilizar una tarjeta embebida que ofrezca más recursos, así como mayor velocidad de comunicación

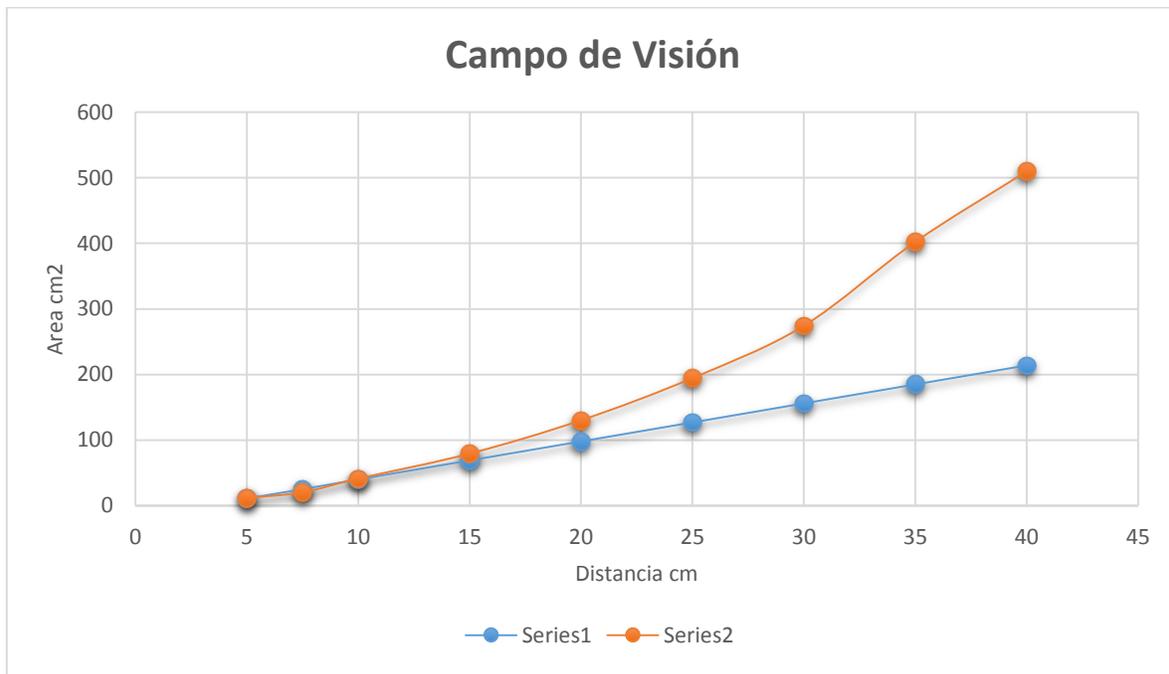
Aumentar la resolución por revolución del sistema de actuación sustituyendo los servomotores por motores de CD retroalimentados digitalmente por encoders absolutos, esta sustitución ayudara a alcanzar mayores velocidades angulares

BIBLIOGRAFÍA

- [1] R. L. Finn and D.Wright; "Unmanned aircraft systems: Surveillance, ethics and privacy in civil applications" Computer Law & Security Review, 2012.
- [2] D. Rainer, Rules; " regulations and codes for drones, unmanned aerial vehicle, extgen air transportation, unmanned air systems," Journal of Chemical Health and Safety;2014.
- [3] Ben Richmond; "Colorado votara para derribar Drones"; VICE Media; 2013
- [4] Northrop Grumman; "Skyguard Laser Defence System" Space war; 2006
- [5] Loreto G.; "Visual Servoing con compensación utilizando redes neuronales"; Tesis CINVESTAV-IPN; 2002
- [6] D. Marr, "Visión". San Francisco, 1982
- [7] Gonzalo Pajares & De la cruz, "Visión por computador: Imágenes digitales y aplicaciones". Alfaomega, México, 2008
- [8] B. A. Fernández; "El láser, la luz de nuestro tiempo"; Isabel Arias Tobalina; 2010
- [9] Thomas L. Floyd, "Dispositivos Electronicos", Pearson Education, 2008
- [10] García Eduardo, "Compilador C CCS para Microcontroladores PIC", Alfaomega, 2008
- [11] Paul Viola & Michael J. Jones; "Robust Real-Time Object Detection"; Cambridge Research Laboratory; 2001
- [12] Paul Viola & Michael J. Jones; "Rapid Object Detection using a Boosted Cascade of Simple Features"; Conference on Computer Vision and Pattern Recognition"; 2001
- [13] IEEE Standar Glossary for Image Processing and Pattern recognition Terminology. Published by the Institute of Electrical and Electronics Engineers, New York, USA, 1990.
- [14] P. KaewTraKulPong & R. Bowden, "An improved Background Mixture Model for Real-time Tracking with Shadow Detection"; Brunel University, 2001
- [15] Stauffer &Grimson , " Adaptive background mixture models for real-time tracking", The Artificial Intelligence Laboratory Massachusetts Institute of Technology, 1999,
- [16] Gary R. Bradski; "Computer visión face tracking for use in a perceptual user interface"; Intel Technology Journal; 1998; vol.2, pp. 13-27
- [17] D. & Meer P.; "Mean Shift: A robust approach toward feature space analisis" IEEE transactions on pattern analysis and machine intelligence; 2002

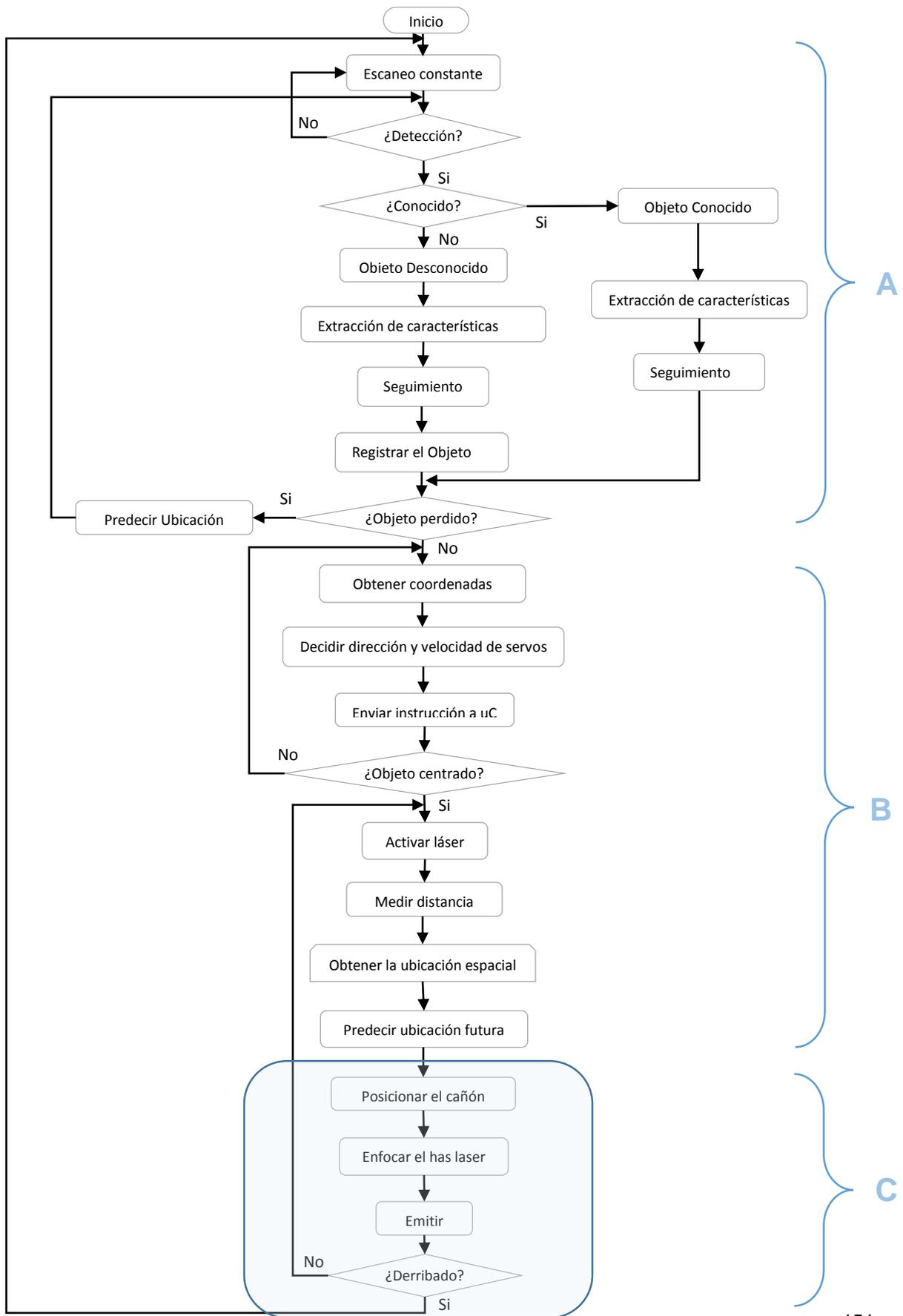
-
- [18] Shi, J., and C. Tomasi, "Good Features to Track," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, June 1994, pp. 593–600.
- [19] Tomasi, Carlo and Takeo Kanade. Detection and Tracking of Point Features, Computer Science Department, Carnegie Mellon University, April, 1991.
- [20] F. Rodríguez Rubio; & López M. "Control Adaptativo y Robusto"; Universidad de Sevilla; España; 1996
- [21] Greg Welch & Gary Bishop; "An Introduction to the Kalman Filter"; Department of Computer Science University of North Carolina at Chapel Hill, 2006
- [22] Gonzalo Pajares & De la cruz, "Visión por computador: Imágenes digitales y aplicaciones". Alfaomega, México, 2008
- [23] Ahmad A. Hammed K. Muhammad I.; "Automatic Visual Tracking and Firing System for Anti Aircraft Machine Gun"; National Engineering & Scientific Commission; Pakistan; 2009
- [24] Van Vankenburgh; "Fire Control Fundamentals"; Department of the Navy USA; 1953
- [25] T. Flores; "La ablación láser"; Tesis; CICATA-IPN; 2009
- [26] M. Inguscio; "Solid State Laser", New Developments and Applications; 1993
- [27] Jürgen Pöppelmann Tte. Cnel. (RE) FAA, "Usos militares del láser", Revista de publicaciones navales, 2009
- [28] Dr. Oded Amichai; "Defense Systems Against Rockets & Mortars"; 2008

ANEXO



Grafica que muestra la tendencia no lineal en la apertura de la cámara comparada con la apertura lineal esperada.

Los valores se obtuvieron de forma experimental aumentando al mismo tiempo que midiendo la distancia de la cámara respecto a la zona de medición. En esta zona de medición se consiguió la distancia máxima perceptible tanto en X como en Y, con estos valores se calculó el área del campo de visión.



A

B

C