



Diseño y construcción de un prototipo para localización y seguimiento de personas basado en el sensor de detección electroóptico D1(SEO D2) equipado en estructura estabilizadora Gimbal

Moyano Cabezas, Luis Fernando y Viracucha Mosquera, Dennis Alexander

Departamento de Ciencia de la Energía y Mecánica

Carrera de Ingeniería en Mecatrónica

Trabajo de titulación previo a la obtención del título de Ingeniero en Mecatrónica

Dr. Aguilar Castillo, Wilbert Geovanny

15 de agosto de 2022



TesisMoyanoViracucha.pdf

Scanned on: 23:56 August 11, 2022 UTC



Overall Similarity Score



Results Found



Total Words in Text

Identical Words	691
Words with Minor Changes	0
Paraphrased Words	166
Omitted Words	1027

.....
Dr. Aguilar Castillo, Wilbert Geovanny.
DIRECTOR



Departamento de Ciencias de la Energía y Mecánica

Carrera de Ingeniería en Mecatrónica

Certificación

Certifico que el trabajo de titulación: **“Diseño y construcción de un prototipo para localización y seguimiento de personas basado en el sensor de detección electroóptico D1(SEO D2) equipado en estructura estabilizadora Gimbal”** fue realizado por los señores Moyano Cabezas Luis Fernando y Viracucha Mosquera Dennis Alexander; el mismo que cumple con los requisitos legales, teóricos, científicos, técnicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, además fue revisado y analizado en su totalidad por la herramienta de prevención y/o verificación de similitud de contenidos; razón por la cual me permito acreditar y autorizar para que se lo sustente públicamente.

Sangolquí 15 de agosto del 2022.

Firma:

Dr. Aguilar Castillo, Wilbert Geovanny

C. C 0703844696



Departamento de Ciencias de la Energía y Mecánica

Carrera de Ingeniería en Mecatrónica

Responsabilidad de Autoría

Nosotros, **Moyano Cabezas Luis Fernando** y **Viracucha Mosquera Dennis Alexander**, con cédulas de ciudadanía n° 1726016718 y n° 1724015431, declaramos que el contenido, ideas y criterios del trabajo de titulación: **"Diseño y construcción de un prototipo para localización y seguimiento de personas basado en el sensor de detección electroóptico D1(SEO D2) equipado en estructura estabilizadora Gimbal"** es de nuestra autoría y responsabilidad, cumpliendo con los requisitos legales, teóricos, científicos, técnicos, y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

Sangolquí 15 de agosto del 2022.

Firmas

Moyano Cabezas Luis Fernando
CC. 1726016718

Viracucha Mosquera Dennis Alexander
CC. 1724015431



Departamento de Ciencias de la Energía y Mecánica

Carrera de Ingeniería en Mecatrónica

Autorización de Publicación

Nosotros **Moyano Cabezas Luis Fernando y Viracucha Mosquera Dennis Alexander**, con cédulas de ciudadanía n° 1726016718, y n° 1724015431 autorizamos a la Universidad de las Fuerzas Armadas ESPE publicar el trabajo de titulación: **Título: "Diseño y construcción de un prototipo para localización y seguimiento de personas basado en el sensor de detección electroóptico D1(SEO D2) equipado en estructura estabilizadora Gimbal"** en el Repositorio Institucional, cuyo contenido, ideas y criterios son de nuestra responsabilidad.

Sangolquí 15 de agosto del 2022.

Firmas

.....
Moyano Cabezas Luis Fernando
CC. 1726016718

.....
Viracucha Mosquera Dennis Alexander
CC. 1724015431

Dedicatoria

Dedico este trabajo a mi familia, mi mami Mónica, mi papi Ramiro y a mi hermano Mateo, que han sido el pilar más importante y apoyo incondicional a través de este viaje universitario que, a pesar de tantas caídas, me han sabido levantar de tal manera que me entregan más fuerzas de avanzar. A mi abuelita Lucrecia, por haberme criado desde niño y estar siempre al pendiente de los logros y avances que he tenido en mi vida, y haberme dado fuerzas de seguir cumpliendo objetivos y que pueda estar presente en ellos. Por ese apoyo incondicional, me encuentro donde estoy ahora, y tratar de hacerlos sentir orgullosos en la persona que me he convertido.

Viracucha Mosquera Dennis Alexander

Dedico este trabajo a dios y a la madre dolorosa por siempre haberme dado las fuerzas y la sabiduría para continuar en los momentos difíciles, a mi padre Juan por todo el esfuerzo realizado para que pueda alcanzar mis metas, así como sus consejos y enseñanzas, a mi madre Mercedes por siempre estar pendiente, velar por nuestra seguridad y enseñarnos el amor más grande, a mi hermano Cristian por siempre ser una guía y un ejemplo en el camino, a mi hermano Juan José por siempre apoyarme incondicionalmente. Gracias a ustedes he logrado cumplir esta meta la cual requirió de mucho esfuerzo y sacrificio, por lo cual este y todos mis logros serán continuamente dedicado a ustedes

Moyano Cabezas Luis Fernando

Agradecimientos

Mi agradecimiento va dirigido totalmente hacia mi familia, por haberme dado la fuerza que necesitaba cuando sentía que ya no podía dar un paso más. A mis padres Ramiro y Mónica por enseñarme que teniendo constancia y una meta clara se pueden lograr grandes objetivos a largo y corto plazo, por estar ahí en las madrugadas para decirme que ya fue suficiente y que descanse, o que aún puedo dar un poco más de esfuerzo para presentar algún proyecto. Gracias por ser los mejores padres en este mundo. A mi ñaño Mateo, por demostrarme que cuando hay algo que le apasiona, no importa el día, hora ni distancia que le impida seguir su sueño, generando tal confianza que me hace sentir orgulloso de ser su hermano.

A mi grupo de amigos denominados “The Boys” por haber hecho de este viaje lo más ameno posible, por haberme incluido en el grupo y saber que en la Universidad sí se puede tener amigos de verdad que se cuidan entre ellos, sin envidia y cariño genuino, convirtiéndose, ya no solo en un grupo de amigos, sino también en una familia en quien confiar y con varias experiencias únicas que nunca se van a olvidar y a su vez, se convertirán en muchas más.

A mi amigo Luis “Moya” por haberme acompañado en este último paso, en este último proyecto, que, a pesar de todos los obstáculos que tuvimos, supimos llevar a cabo el proyecto final. Además, el agradecimiento por no dejarme botar la toalla cuando ya todo parecía perdido y a pesar de todas las noches sin dormir, todo el esfuerzo realizado juntos, valdrá la pena.

Agradezco a la Universidad de las Fuerzas Armadas-ESPE por acogerme en sus aulas y permitirme convertirme en Ingeniero Mecatrónico, un sueño que parecía tan lejano como el primer día, pero que ahora, se ve tan cerca como tener a la palma de la mano.

Agradezco al Dr. Wilbert Aguilar, por habernos confiado el proyecto presente para su realización, además por su ayuda y guía para poder conseguir este hito de manera satisfactoria en todos los aspectos.

Viracucha Mosquera, Dennis Alexander

Agradezco a dios y a la madre dolorosa por haberme permitido cumplir esta meta por la sabiduría y la fortaleza para no desmayar en los momentos difíciles cuando el camino se volvió difícil.

Agradezco a mi familia por confiar en mí y motivarme a seguir adelante, agradezco a mi padre Juan a mi madre Mercedes, a mis hermanos Cristian y Juan por siempre ser el soporte en mi vida y ayudarme con un consejo una risa los cuales siempre me motivaron a seguir adelante y cumplir todos mis objetivos. Gracias por cada esfuerzo y desvelo realizado por mi padre para poder estudiar y siempre tener todas las herramientas para seguir adelante, gracias por cada desayuno que con tanto amor realiza mi madre para tener energía todos los días, por su amor y cariño incondicional, gracias a mis hermanos por siempre ayudarme en cualquier situación con sus consejos con su compañía con sus risas y peleas.

Agradezco a Karla por siempre brindarme su apoyo para cada día ser mejor y crecer juntos, por todo su amor y cariño incondicional, por siempre apoyarme y motivarme a cumplir mis metas, por todas las risas, anécdotas, desvelos que hicieron de esta etapa de mucha felicidad

Agradezco a mi amigo Alex por culminar juntos la última etapa de esta meta la cual ha sido una de las más difíciles, gracias por el apoyo consejo risas, por motivarnos mutuamente para no desmayar y llegar a nuestro objetivo.

Agradezco a todos mis amigos de la universidad con los cuales siempre supieron ayudarme en los momentos difíciles, en las pruebas en los proyectos, en los viajes en cada partido de futbol, demostrando siempre que existen grandes amistades que estarán ahí para apoyarte cuando lo necesites en cualquier momento.

Agradezco a la Universidad de las Fuerzas Armadas ESPE y a todos sus docentes los cuales lograron transmitir su conocimiento a mi persona para poder

convertirme en un Ingeniero Mecatrónico con un objetivo muy firme ser más para servir mejor.

Agradezco al Dr. Wilbert Aguilar por brindando todos sus conocimientos y consejos en el desarrollo del trabajo de titulación, siendo una fuente de motivación para poder alcanzar esta meta tan anhelada.

Luis Fernando Moyano Cabezas

Índice de contenidos

Contenido

Portada.....	1
Dedicatoria	6
Agradecimientos.....	7
Índice de contenidos.....	11
Índice de Tablas	15
Índice de Figuras	17
Resumen	23
Abstract	24
Capítulo I.....	25
Antecedentes	25
Justificación e importancia	30
Objetivos.....	31
Objetivo general	31
Objetivos específicos.....	31
Alcance del proyecto.....	32
Sistema Mecánico.....	32
Sistema electrónico.....	33
Sistema de control	34
Sistema de tecnologías de la información	34
Estructura del proyecto	35
Capítulo II.....	37

	12
Sensor electro-óptico	37
Gimbal	38
Disposición RPY (Roll-Pitch-Yaw)	38
Disposición PTZ (Pan, Tilt, Zoom)	39
Tipos de gimbal	40
Motores	42
Diseño generativo	44
Centro de masas.....	48
Tipos de control.....	49
Controlador PID.....	49
Estructura del controlador PID.....	49
Controlador lógica difusa	51
Machine Learning.....	54
Redes Neuronales Convolucionales	55
Aprendizaje supervisado.....	58
TensorFlow.....	60
TensorFlow Mobile	60
TensorFlow Lite	61
Android Studio.....	61
Comunicación	62
Comunicación I2C	62
Comunicación radiofrecuencia.....	63
Tecnologías inalámbricas	63

	13
Telemetría.....	64
Capítulo III	66
Diseño mecatrónico	66
Subsistema mecánico	67
Selección de componentes.....	68
Selección de configuración de gimbal.....	68
Modelamiento matemático.....	70
Diseño CAD.....	74
Análisis diseño generativo	81
Análisis del centro de masa	89
Subsistema electrónico	90
Sistema de adquisición de imagen	90
Tarjeta controladora.....	93
Alimentación.....	97
Subsistema Control.....	100
Algoritmo de visión artificial	100
Diseño de la aplicación.....	113
Comunicación.....	118
Diseño del controlador.....	125
Diseño del HMI de estación en tierra	132
Capitulo IV.....	137
Pruebas y resultados.....	137
Sistema mecánico	137

	14
Pruebas de comunicación.....	141
Prueba de Red Neuronal	143
Pruebas de seguimiento	164
Capítulo V.....	167
Conclusiones y recomendaciones.....	167
Conclusiones	167
Recomendaciones	169
Bibliografía.....	171
Apéndices.....	182

Índice de Tablas

Tabla 1 Requerimientos mecánicos.....	67
Tabla 2 <i>Conceptos disposición gimbal</i>	68
Tabla 3 <i>Matriz de selección disposición</i>	70
Tabla 4 Tabla de parámetros Denavit-Hartenberg	72
Tabla 5 <i>Propiedades mecánicas de Acrílico Estándar</i>	82
Tabla 6 <i>Resumen-Análisis estático</i>	84
Tabla 7 <i>Resumen del diseño generativo por pieza</i>	88
Tabla 8 <i>Requerimientos sistema de adquisición de imagen</i>	90
Tabla 9 <i>Conceptos adquisición de imagen</i>	91
Tabla 10 <i>Matriz de selección sistema de adquisición de imágenes</i>	92
Tabla 11 <i>Requerimientos tarjeta controladora</i>	93
Tabla 12 <i>Conceptos tarjetas controladoras</i>	94
Tabla 13 <i>Matriz de selección tarjeta controladora</i>	95
Tabla 14 <i>Mediciones experimentales corriente servomotores</i>	97
Tabla 15 <i>Dimensionamiento de la corriente</i>	99
Tabla 16 <i>Modelos Pre-entrenados</i>	100
Tabla 17 Comparación de prototipo con gimbal comercial.....	139
Tabla 18 <i>Pruebas de funcionamiento</i>	141
Tabla 19 <i>Distancia de transmisión sin repetidor</i>	142
Tabla 20 <i>Distancia de transmisión con repetidor</i>	143
Tabla 21 Diseño experimental de Taguchi	145
Tabla 22 <i>Valores de certeza variando distancia</i>	148
Tabla 23 <i>Valor de certeza variando la luminosidad</i>	150
Tabla 24 <i>Valor de certeza variando la altura</i>	151
Tabla 25: <i>Valor de certeza variando la cantidad de personas</i>	153
Tabla 26 Diseño de Taguchi 3 factores con 3 niveles	154
Tabla 27 Ejemplo de los diferentes casos matriz de confusión	158

Tabla 28 Diseño de Taguchi con Luminosidad Alta	159
Tabla 29 Diseño de Taguchi con Luminosidad Media	159
Tabla 30 Diseño de Taguchi con Luminosidad Baja	160

Índice de Figuras

Figura 1 <i>Gimbal comercial de 3 ejes</i>	25
Figura 2 <i>Sistema de monitoreo múltiple</i>	27
Figura 3 <i>Uso de Visión Artificial para detectar EPP.</i>	28
Figura 4 <i>Detección de perfil de rostros</i>	29
Figura 5 <i>Detección de Sonrisas</i>	30
Figura 6 <i>Diseño de conexión de elementos electrónicos</i>	33
Figura 7 <i>Diagrama sistema de control del Gimbal</i>	34
Figura 8 <i>Detección de objetivos</i>	35
Figura 9 <i>Modelos de sensores electro ópticos de la familia STAMP</i>	37
Figura 10 <i>Gimbal 3 ejes</i>	39
Figura 11 <i>Funcionamiento de Cámara PTZ</i>	40
Figura 12 <i>Gimbal de 2 ejes</i>	41
Figura 13 <i>Gimbal de 3 ejes</i>	41
Figura 14 <i>Gimbal 4 y 5 ejes</i>	42
Figura 15 <i>Funcionamiento motor DC</i>	43
Figura 16 <i>Diseño servomotor</i>	43
Figura 17 <i>Estructura del motor Brushless</i>	44
Figura 18 <i>Diseño generativo aplicado a la arquitectura</i>	45
Figura 19 <i>Diseño generativo con SolidWorks</i>	46
Figura 20 <i>Diseño Generativo por Altair</i>	47
Figura 21 <i>Centro de masa de diferentes figuras geométricas</i>	48
Figura 22 <i>Centro de masa gimbal</i>	49
Figura 23 <i>Estructura de PID</i>	51
Figura 24 <i>Lazo de control difuso</i>	52
Figura 25 <i>Diagrama de bloques controlador fuzzy</i>	53
Figura 26 <i>Estructura básica de control Fuzzy</i>	54
Figura 27 <i>Arquitectura de una CNN</i>	55

Figura 28 <i>Flujo de trabajo de aprendizaje supervisado</i>	59
Figura 29 <i>Estructura del proyecto de Android Studio</i>	62
Figura 30 <i>Comunicación I2C</i>	63
Figura 31 <i>Redes inalámbricas</i>	64
Figura 32 <i>Diagrama telemetría</i>	65
Figura 33 <i>Norma VDI2206</i>	66
Figura 34 <i>Configuración-Gimbal de 2 ejes</i>	71
Figura 35 <i>Configuración de juntas</i>	72
Figura 36 <i>Sistema de coordenadas esféricas</i>	73
Figura 37 <i>Versión en MDF</i>	74
Figura 38 <i>Versión MDF montado</i>	75
Figura 39 <i>Modelo V2</i>	76
Figura 40 <i>Modelo V2 montado</i>	77
Figura 41 <i>Modelo V3</i>	78
Figura 42 <i>Acople de motor V3</i>	79
Figura 43 <i>Modelo PTZ</i>	79
Figura 44 <i>Modelo V4</i>	80
Figura 45 <i>Modelo Final</i>	81
Figura 46 <i>Análisis de Tensiones</i>	82
Figura 47 <i>Análisis de Desplazamiento</i>	83
Figura 48 <i>Análisis de deformación unitaria</i>	83
Figura 49 <i>Definición de Geometría fija</i>	85
Figura 50 <i>Definición de carga externa</i>	85
Figura 51 <i>Objetivo de análisis</i>	86
Figura 52 <i>Definición de malla</i>	86
Figura 53 <i>Diseño Generativo final</i>	87
Figura 54 <i>Análisis gráfico de centro de masa</i>	89
Figura 55 <i>Sistema de control de prototipo</i>	96

Figura 56 <i>Medición de corriente servomotores</i>	98
Figura 57 <i>Batería de 7.4V y 1500mA</i>	99
Figura 58 <i>Ejemplo modelo pre-entrenado</i>	102
Figura 59 <i>Estructura de una red neuronal</i>	103
Figura 60 <i>Ejemplo de red con varias capas convolucionales</i>	104
Figura 61 <i>Modelo SSD</i>	105
Figura 62 <i>Imagen de dataset de validación COCO 2017</i>	106
Figura 63 <i>Base de datos de entrenamiento</i>	107
Figura 64 <i>Etiquetado de imágenes</i>	108
Figura 65 <i>Archivo de etiquetado</i>	108
Figura 66 <i>Parámetros de etiquetado</i>	111
Figura 67 <i>Graficas de pérdidas</i>	112
Figura 68 <i>Entrenamiento vs evaluación</i>	112
Figura 69 <i>Grafica de precisión del modelo</i>	113
Figura 70 <i>Diseño gráfico de la aplicación</i>	114
Figura 71 <i>Clases de la aplicación</i>	115
Figura 72 <i>Pruebas zoom</i>	118
Figura 73 <i>Diagrama de comunicación</i>	118
Figura 74: <i>Arquitectura cliente-servidor</i>	119
Figura 75 <i>Código implementación servidor raspberry</i>	120
Figura 76 <i>Implementación cliente AndroidStudio</i>	122
Figura 77 <i>Conversión de imagen a base 64</i>	123
Figura 78 <i>Formato JSON</i>	124
Figura 79 <i>Métodos HTTP</i>	125
Figura 80 <i>Toma de datos de comportamiento de planta</i>	126
Figura 81 <i>Movimiento de servomotores</i>	126
Figura 82 <i>Comportamiento de píxeles</i>	127
Figura 83 <i>Estimación numérica de la planta</i>	128

Figura 84 <i>Diseño de controlador</i>	129
Figura 85 <i>Estructura del filtro de Kalman</i>	131
Figura 86 <i>Comportamiento de Filtro de Kalman</i>	131
Figura 87 <i>Diseño HMI</i>	132
Figura 88 <i>Diseño HTML</i>	133
Figura 89 <i>Programación página web</i>	133
Figura 90 <i>HMI implementada en computadora</i>	134
Figura 91 <i>HMI implementado en teléfono móvil</i>	135
Figura 92: <i>Comportamiento de flujo de información</i>	136
Figura 93 <i>Diseño CAD en SolidWorks</i>	137
Figura 94 <i>Prototipo final gimbal</i>	138
Figura 95 <i>Comparación del diseño CAD y el prototipo físico.</i>	139
Figura 96 <i>Reconocimiento-pruebas de funcionamiento</i>	144
Figura 97 <i>Diseño experimental de Taguchi minitab</i>	146
Figura 98 <i>Criterio de calidad según Taguchi</i>	147
Figura 99 <i>Perdida por unidad diseño de Taguchi</i>	147
Figura 100 <i>Pruebas de distancia</i>	149
Figura 101 <i>Pruebas de luminosidad</i>	151
Figura 102 <i>Pruebas de altura</i>	152
Figura 103 <i>Pruebas realizadas variando cantidad de personas</i>	154
Figura 104 <i>Diseño experimental en minitab</i>	155
Figura 105 <i>Modelo de matriz de confusión</i>	156
Figura 106 <i>Gráfico de efectos principales para medias</i>	160
Figura 107 <i>Gráficos de residuos para medias</i>	160
Figura 108 <i>Ejemplos de la realización del diseño experimental</i>	161
Figura 109 <i>Respuesta para medias</i>	162
Figura 110 <i>Análisis de Sensibilidad realizado en minitab</i>	163
Figura 111 <i>Gráficas residuales realizadas en minitab</i>	164

Figura 112 Cálculo de error en imagen.....	165
Figura 113 Seguimiento de objetivo militar	166

Glosario

FPS: Frames por Segundo

SSD : Single Shot multiBox Detector

CNN: Convolutional Neural Network

PTZ: Pan-Tilt-Zoom

CAD: Dibujo Asistido por Computador

HMI: Interfaz Humano Máquina

I2C: Inter-Integrated Circuits

YOLO: You Only Look Once

IMU: Unidad de Medida Inercial

UAV: Unmanned Aerial Vehicle

PID: Proportional Integrative Derivative

Resumen

En el presente Proyecto de investigación, se presenta el desarrollo de un sistema para la detección y seguimiento de personas con vestimenta militar, el mismo que se encuentra montado en una estructura gimbal. Se desarrolló un prototipo de gimbal mediante un proceso de mejoramiento continuo hasta llegar al prototipo final pasando por diferentes diseños, así como de procesos de manufactura.

La detección de personas se realizó mediante redes neuronales convolucionales, usando un modelo pre-entrenado de TensorFlow el cual sea compatible con arquitectura Android. Se realizó una base de datos de imágenes de militares en diferentes situaciones de iluminación, distancia, altura y número de personas. Con el modelo entrenado se desarrolló la aplicación en AndroidStudio que permitió el envío de imágenes, puntos medios de la imagen, puntos medios del objetivo, nivel de certeza de detección y coordenadas del GPS, hacia la placa controladora.

Para el desarrollo del controlador se desarrolló una identificación de la planta para determinar el tipo de control más óptimo. Para la comunicación con la estación en tierra se realizó un servidor dentro de la tarjeta controladora Raspberry Pi 3, en la cual se implementó un servidor web para acceder a los datos e imágenes del gimbal, usando las mismas características de la tarjeta.

Se ejecutó la integración de los componentes mencionados anteriormente dando como resultado un dispositivo embebido capaz de realizar la detección y seguimiento de personas con vestimenta militar, los datos obtenidos se pueden en una estación en tierra conectada mediante un web server y una página web.

Palabras clave: gimbal, machine learning, tensorflow, detección, seguimiento

Abstract

In this research project, the development of a system for the detection and monitoring of people in military clothing is presented, which is mounted on a gimbal structure. A gimbal prototype was developed through a process of continuous improvement until reaching the final prototype, going through different designs, as well as manufacturing processes.

For the detection of people, it was done through convolutional neural networks, using a pre-trained model of TensorFlow which is compatible with Android architecture. A database of military images was made in different circumstances of lighting, distance, height and number of people. With the trained model, the application was developed in AndroidStudio, which allowed us to send images, midpoints of the image, midpoints of the objective, detection certainty level and GPS coordinates, to the controller board.

For the development of the controller, an identification of the plant was carried out to determine the most optimal type of control. For communication with the ground station, a server was developed within the Raspberry Pi 3 controller card, in which a web server was implemented through which we could access the data and images of the gimbal, using the same characteristics of the card.

The integration of the aforementioned components was carried out, giving us as a result an embedded device which is capable of detecting and monitoring people in military clothing, the data obtained can be seen in a ground station connected through a web server and a page Web

Keywords: Gimbal, machine learning, tensorflow, detection, tracking

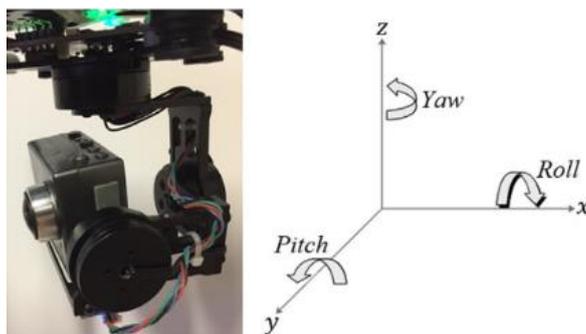
Capítulo I

Antecedentes

Un gimbal (Figura 1) es un dispositivo que suaviza los movimientos angulares de una cámara para adquirir imágenes más estables (Gašparović & Jurjevic, 2017). Es significativamente beneficioso para aplicaciones de estabilización de imagen en tiempo real, ya que reduce las vibraciones y mantiene la cámara en una posición predefinida (Aguilar & Angulo, Real-time video stabilization without phantom movements for micro aerial vehicles, 2014).

Figura 1

Gimbal comercial de 3 ejes



Nota: Gimbal de 3 ejes el cual nos permite movimiento en los ejes Roll, Pitch y Yaw. Tomado de *Robust control and modeling a 2-DOF Inertial Stabilized Platform* (p.223), por H. Khodadadi, M. R. J. Motlagh and M. Gorji, 2011

Estudios previos de diferentes universidades del país y del mundo muestran estrategias sobre estabilización de gimbal, procesamiento de imágenes (Aguilar & Angulo, Estabilización robusta de vídeo basada en diferencia de nivel de gris, 2013), detección de objetos, georreferenciación, entre otros (Galarza, Pérez, Serrano, Tapia, & Aguilar, 2018). Varios de ellos, se evidencian en el presente capítulo.

El trabajo de titulación “Sistema de procesamiento digital de imágenes para realizar tracking automático de objetos durante misiones de vigilancia y reconocimiento utilizando sistemas electroópticos del CIDFAE” desarrollado en la Universidad Técnica de Ambato (Araujo, 2018); tuvo como objetivo convertir una señal analógica proveniente de un sensor electro-óptico (cámaras), en una señal digital para el procesamiento, aplicando algoritmos y filtros respectivos. Esta investigación podría ser el primer paso de un tracking automático para identificar objetivos durante las misiones aéreas (Tenezaca, Canchignia, Aguilar, & Mendoza, 2019).

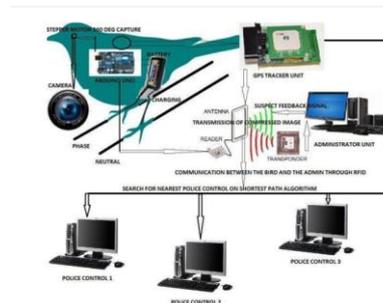
De la misma forma, el estudio “Control de sensores electro-ópticos en aviones no tripulados y tratamiento de imágenes utilizando máquinas de vectores soporte”, de la Universidad Carlos III de Madrid (Escobar, 2010), consistió en la implementación de algoritmos de geo-apuntamiento para calcular los ángulos del sensor electro-óptico hacia una latitud, longitud y altura determinada. Además, añadieron parámetros de geolocalización para calcular los mismos parámetros de la posición geográfica del enfoque de la cámara.

En una investigación desarrollada en la Universidad de las Fuerzas Armadas ESPE con el tema “Diseño de un sistema de geolocalización de objetivos en tierra basado en el sensor electro-óptico SEO D1(SEO D2)” (Zurita, Aguilar, & Enríquez, Toward the Development of Surveillance and Reconnaissance Capacity in Ecuador: Geolocation System for Ground Targets Based on an Electro-Optical Sensor, 2019) (Zurita, DISEÑO DE UN SISTEMA DE GEOLOCALIZACIÓN DE OBJETIVOS EN TIERRA BASADO EN EL SENSOR ELECTROOPTICO SEO D1(SEO D2), 2018), se logró determinar las coordenadas geográficas de un objetivo en el terreno a través de la metodología de Denavit Hartenberg. Las aplicaciones radican en un sistema de vigilancia y reconocimiento desarrollado por el CIDFAE, basado en el sensor electro-óptico SEO-D1.

Así mismo, uno de los estudios más destacados sobre inteligencia artificial es Artificial Intelligence to Impart Surveillance, Tracking & Actuation on Suspicious Activities (Bhattacharjee & Somashekhar, 2017) sobre la aplicación de detección de amenazas sospechosas en la India por medio de un sistema de reconocimiento adaptado en un dispositivo similar a un dron o pájaro volador. El dispositivo puede dirigirse y monitorear zonas riesgosas, analizar el comportamiento de gente sospechosa y realizar notificaciones a tiempo de ejecución (Figura 2).

Figura 2

Sistema de monitoreo múltiple

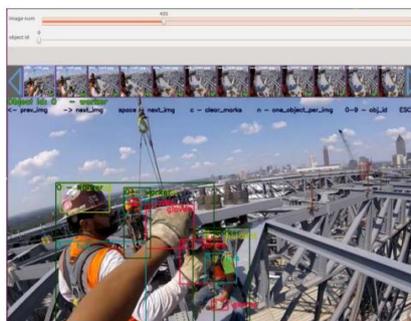


Nota: Estructura de un sistema de monitoreo. Tomado de *Artificial Intelligence to Impart Surveillance, Tracking, & Actuation on Suspicious Activities* (p.1), por Bhattacharjee, Surajit & Somashekhar G.C. 2017

Detección de equipos de protección personal mediante red neuronal convolucional YOLO (Massiris, Delrieux, & Fernández, 2020). Esta investigación se centró en el uso de las *Redes neuronales Convolucionales (CNN)* como método para detección de equipos de protección personal, que es última barrera para detener situaciones potenciales de riesgo físico del trabajador. El estudio propone utilizar visión artificial como alternativa cuantitativa para monitorear el uso de EPP's con la intención de detectar guantes, cascos, ropa de alta visibilidad y a los trabajadores. Utiliza una base de datos creada a partir de videos generados usando cámaras deportivas. Esto se presenta en la (Figura 3)

Figura 3

Uso de Visión Artificial para detectar EPP.



Nota: Detección en tiempo real de los dispositivos EPP. Tomado de *Detección de equipos de protección personal mediante red neuronal convolucional YOLO* (p.1022), por Massiris, Manilo; Delrieux, Claudio; Fernández, Álvaro, 2020

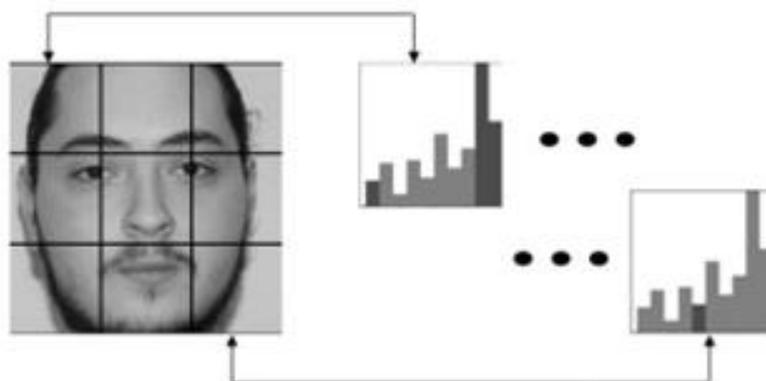
La investigación Artificial Intelligence Techniques Used to Detect Object and Face in an Image (Deepika, Chauhan, & Narayan, 2017) evidencia la necesidad de capacitar el desarrollo de un sistema y la inteligencia artificial para lograr realizar tareas de reconocimiento. El estudio sugiere algunos algoritmos de detección adaptativos y ágiles como “*Edge detection*” para manejar procesos robustos y de gran eficiencia en el reconocimiento; el presente trabajo de titulación utilizará varios de los parámetros manejados en el estudio evidenciado.

Igualmente, en la investigación Profile Face Recognition using Local Binary Patterns with Artificial Neural Network (Salcedo, Aguilar, Cobeña, Pardo, & Zahira, 2018), se muestran que los sistemas de reconocimiento robustos que usan inteligencia artificial (Figura 4), tienen una capacidad limitada, y que, para aumentar los índices de eficacia y rendimiento, es necesario un estudio por patrones de reconocimiento mediante herramientas capacitadas. Los investigadores evidenciaron que una imagen captada se puede desglosar por subregiones de reconocimiento caracterizándola por

partes, de esta forma se puede reconocer una mayor cantidad de patrones, y favorecer al funcionamiento de la red neuronal.

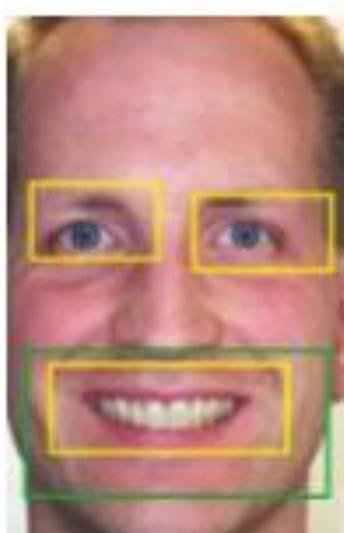
Figura 4

Detección de perfil de rostros



Nota: La figura muestra la segmentación de colores en el rostro, lo cual sirve para detección. Tomado de *Profile Face Recognition using Local Binary Patterns with Artificial Neural Network* (p.4), por Salyut, Jandaulyet, 2018

Finalmente, el estudio *Mejoras en el Entrenamiento de Esquemas de Detección de Sonrisas Basados en AdaBoost* (Merchán Fernando, 2014) abordó aspectos del entrenamiento de la máquina de aprendizaje AdaBoost con modelos de reconocimiento de objetos con características como: Patrones Binarios Locales (LBP), Histograma de Gradientes Orientados (HOG) (Aguilar W. G., y otros, *Pedestrian Detection for UAVs Using Cascade Classifiers and Saliency Maps*, 2017) y características tipo Haar para la detección de sonrisas (Figura 5).

Figura 5*Detección de Sonrisas*

Nota: La figura muestra la detección de las sonrisas mediante cuadros delimitadores. Tomado de *Mejoras en el entrenamiento de esquemas de detección de sonrisas basados en AdaBoost* (p.30) por Merchán, Fernando & Galeano, Sebastián, 2014

Justificación e importancia

El CIDFAE es un organismo que forma parte de la Fuerza Aérea Ecuatoriana con el objetivo desarrollar investigación científica y tecnológica aeroespacial para mejorar la capacidad operativa de la Fuerza Aérea y contribuir a la producción científica tecnológica y al desarrollo nacional.

Uno de los grandes problemas del Ecuador es el aumento de las actividades ilícitas como narcotráfico y muertes violentas, que generalmente se presentan en zonas de difícil acceso o peligrosas para el personal militar o civil; por consiguiente, las misiones de vigilancia y reconocimiento implican poner en riesgo a los pilotos y exponerlos a ser derribados, capturados o heridos.

Frente a esta problemática, el presente proyecto de titulación desarrollará un prototipo para la detección y seguimiento de personas dentro de espacios cerrados, además, el ejemplar podrá ser escalable para su aplicación en zonas abiertas, de difícil acceso e inseguras.

Cabe mencionar, que el algoritmo de visión artificial determinará un rápido procesamiento de las imágenes, envío de señal a los actuadores y selección de objetivos; de esta forma, se podrá realizar el seguimiento sin afectar en la ruta del UAV, un problema común al utilizar una cámara fija.

Objetivos

Objetivo general

- Diseñar y construir un prototipo tipo gimbal, basado en el sensor de detección electro-óptico D1 (SEO D2) con algoritmo de visión artificial y machine learning para detección y seguimiento de objetivos

Objetivos específicos

- Realizar el diseño y construcción mecánico de la estructura estabilizadora tipo gimbal
- Diseñar el sistema electrónico y comunicación para la transmisión y procesamiento de información.
- Desarrollar un sistema de visión artificial para el control del prototipo que permita la detección y seguimiento de personas.
- Realizar pruebas de funcionamiento del prototipo empleando un modelo experimental y análisis estadístico para validación del mismo

Alcance del proyecto

Dentro del alcance del proyecto se considera los aspectos de diseño mecánico, electrónico, diseño de control y tecnologías de información. Todos permitirán realizar el algoritmo de detección de personas con un control para el seguimiento automático montado en una estructura gimbal. El diseño incluye cálculos de ingeniería, diseño de prototipos, desarrollo de algoritmos y documentación de ingeniería tales como: diagramas de conexión, planos mecánicos individuales y de ensamblaje, además de documentación de explicación de algoritmos usados en el proyecto.

Sistema Mecánico

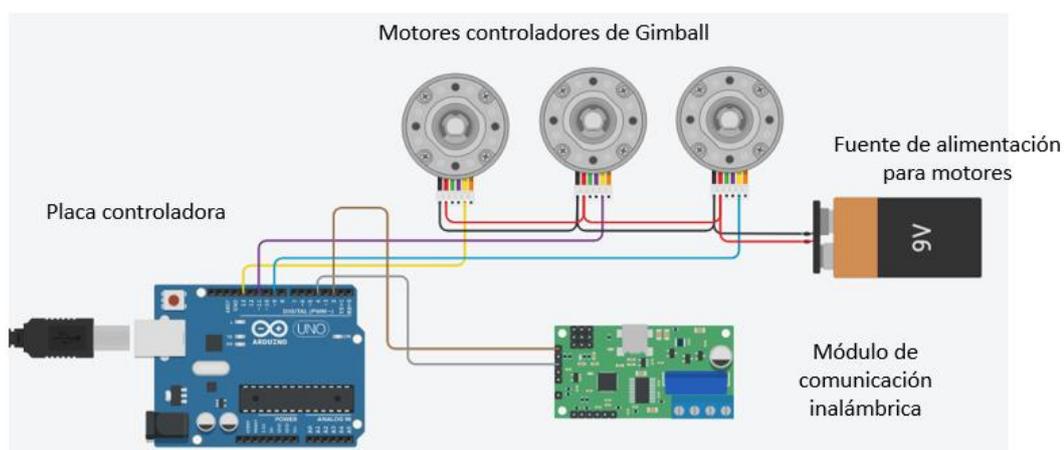
- **Diseño estructural:** Se establecerán los parámetros de construcción del prototipo: cargas a soportar y torques generados por los actuadores seleccionados. Se realizará un diseño generativo para reducir el peso del prototipo con las características de resistencia mecánica disminuyendo el peso por ser estructuras porosas. Esto se efectuará mediante tecnologías CAD/CAE con simulaciones, comprobación y comparación de resultados, con el objetivo de que la estructura sea resistente y de bajo peso.
- **Selección de actuadores (motores):** Se optará por actuadores mecánicos que soporten los torques generados por el sostén de los diferentes elementos, esto en base a catálogos y hojas de datos de fabricantes. Se analizará el uso de servomotores de precisión y caja mecánica para facilitar el soporte de la estructura.
- **Planos mecánicos:** Se elaborarán los planos de piezas y de conjunto necesarios para la manufactura de cada elemento, estos planos se encontrarán bajo la norma ecuatoriana INEN 003 para dibujo mecánico.

Sistema electrónico

- **Selección de controlador:** Se seleccionará una tarjeta microcontroladora adecuada, en base a su arquitectura, entradas y salidas analógicas/digitales disponibles, así como su tipo de comunicación apta para la aplicación requerida.
- **Selección de sensores:** Se requerirá sensores de cámara, giroscopios para la correcta estabilización del sistema gimbal. Estos sensores serán aplicados en base a catálogos y hojas técnicas de los fabricantes.
- **Planos eléctricos:** Se realizarán diagramas esquemáticos y planos eléctricos donde se detallen interconexiones, alimentación y funcionamiento del sistema de acuerdo a la norma IEC 1082-1 (Figura 6)

Figura 6

Diseño de conexión de elementos electrónicos



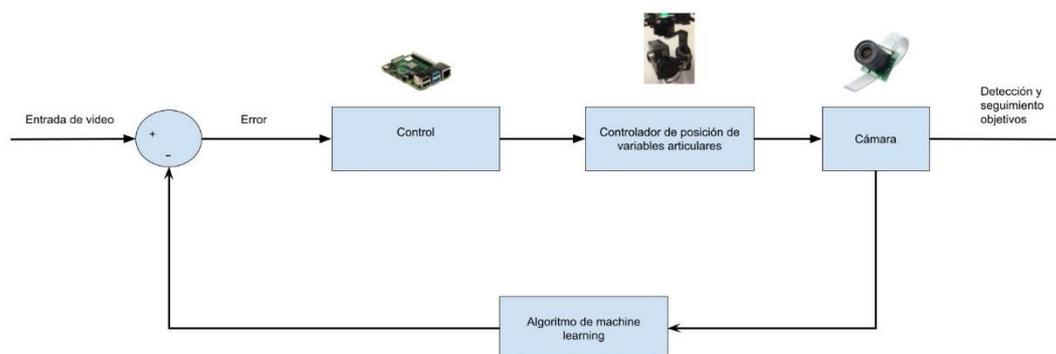
Nota: Se describe de manera esquemática las conexiones entre los componentes electrónicos

Sistema de control

- **Diseño de lazos de control:** El lazo de control tendrá como entrada la señal de video captada por la cámara, la misma será filtrada, analizada y procesada, mediante algoritmos de visión artificial y machine learning como máquinas de soporte vectorial o redes convolucionales. El control enviará la señal a los actuadores de cada articulación para realizar el seguimiento del objetivo (persona).
- **Control del gimbal:** Se realizará una interfaz HMI en una PC que permitirá monitorear y controlar en tiempo real la imagen captada por el gimbal, así como la seleccionar los objetivos para el seguimiento. Esta estructura se presenta en la (Figura 7)

Figura 7

Diagrama sistema de control del Gimbal



Nota: Se describe de manera esquemática, el control que tendrá el sistema

Sistema de tecnologías de la información

- **Implementación de algoritmo de detección de objetivos:** El algoritmo de detección utilizará visión artificial, su función será detectar a una persona y realizar su seguimiento. Se tendrá como base lenguajes de alto nivel que

admitan el uso de herramientas de visión artificial y Machine Learning. La representación del sistema se expone en la (Figura 8).

Figura 8

Detección de objetivos



Nota: La detección de objetivos se puede determinar mediante cuadros delimitadores en la misma imagen. Tomado de *Desarrollo de un Sensor Virtual basado en Modelo NARMAX y Máquina de Vectores de Soporte para Molienda Semiautógena* (p.109), por Acuña, Gonzalo; Curilem, Millaray & Cubillos, Francisco, 2014

Estructura del proyecto

El presente documento está conformado por 5 capítulos. En el Capítulo I se explica los estudios realizados sobre Machine Learning, Algoritmos de visión Artificial, trabajos anteriores realizados sobre seguimiento de objetivos, la importancia del proyecto para el CIDFAE y el alcance del mismo.

En el Capítulo II se detalla el estado del arte. Se exponen los tipos de gimbal, elementos que lo conforman como el sensor IMU, motores, paca controladora, entre otros. Además, se explica los entornos necesarios para el desarrollo de aplicaciones móviles Android, así como la comunicación serial con la placa controladora. Finalmente,

se presenta la comunicación de radio frecuencia para el control y monitoreo del gimbal desde la estación en tierra.

En el capítulo III se exhibe el desarrollo de los diferentes sistemas mecatrónicos. En el subsistema mecánico se realizará el diseño y análisis de la estructura, así como la selección de actuadores; por otro lado, en el subsistema electrónico se seleccionará los componentes de alimentación, tarjetas controladoras y sistema de adquisición de imagen. Por último, en el subsistema de control se realizará el desarrollo del algoritmo de para la detección de objetos al igual que el control y comunicación a tierra.

En el capítulo IV se efectúan las pruebas de funcionamiento del gimbal, tiempo de duración, pruebas de alcance con la estación en tierra y pruebas del algoritmo sometiéndolo a diferentes parámetros de luminosidad, distancia, altura y número de personas.

Por último, en el capítulo V se describen las conclusiones en base al diseño de ingeniería y la eficiencia del seguimiento de objetivos. De la misma forma, se presentan recomendaciones para mejorar el sistema.

Capítulo II

Sensor electro-óptico

Un sensor electro-óptico es un dispositivo que convierte la información contenida de la luz visible o no visible, en una señal eléctrica o viceversa. Una de las ventajas que posee son bajos voltajes y mayor tiempo de velocidad de operación (Martinez, 2018).

CONTROLOP es una empresa privada que se especializa en el desarrollo y producción de sistemas de control de movimiento electro-óptico y de precisión para aplicaciones de vigilancia, defensa y seguridad nacional (CONTROLOP, 2021). La compañía posee la familia STAMP, especializada en sistemas de carga útil reducida principalmente para UAVs con los siguientes modelos: T-STAMP-XD, M-STAMP NG, D-STAMP-HD y T-STAMP-XR. Estos modelos se pueden observar en la (Figura 9). Uno de los grandes inconvenientes de los modelos es el alto costo comercial, superando los miles de dólares dependiendo de las características de las cámaras.

Figura 9

Modelos de sensores electro ópticos de la familia STAMP



Nota: Los diferentes modelos de sensores electro-ópticos pueden tener varias configuraciones para realizar el mismo trabajo. Tomado de *Visual SLAM with a RGB-D Camera on a Quadrotor UAV Using on-Board Processing* (p.596) por Aguilar Wilbert; Rodríguez, Guillermo & Álvarez, Leandro, 2020

El CIDFAE es un organismo altamente técnico con una gran visión del futuro. Su misión está orientada a resolver problemas de orden tecnológico para mejorar la capacidad operativa de la institución (Cedeño, 2019). Uno de los proyectos desarrollados en el área de sensores electro-ópticos es el prototipo SEO-D2 (Jara-Olmedo A. , y otros, 2018), que permite incrementar la capacidad de: observación (Aguilar, y otros, 2017), vigilancia (Aguilar W. G., y otros, 2017) y reconocimiento (Calderón, Aguilar, & Merizalde, 2020) a través del procesamiento de imágenes y explotación de las aplicaciones del sistema que incluyen: visión infrarroja (Andrea, Byron, Jorge, Inti, & Aguilar, 2018), rastreo (Aguilar, Cobeña, Rodríguez, Salcedo, & Collaguazo, 2018) y geo-referencia de blancos (Aguilar, Manosalvas, Guillén, & Collaguazo, 2018), reconocimiento de coordenadas (Aguilar & Morales, 3D Environment Mapping Using the Kinect V2 and Path Planning Based on RRT Algorithms, 2016), toma de videos (Basantes, y otros, 2018) e imágenes aéreas (Pardo, Aguilar, & Toulkeridis, 2017).

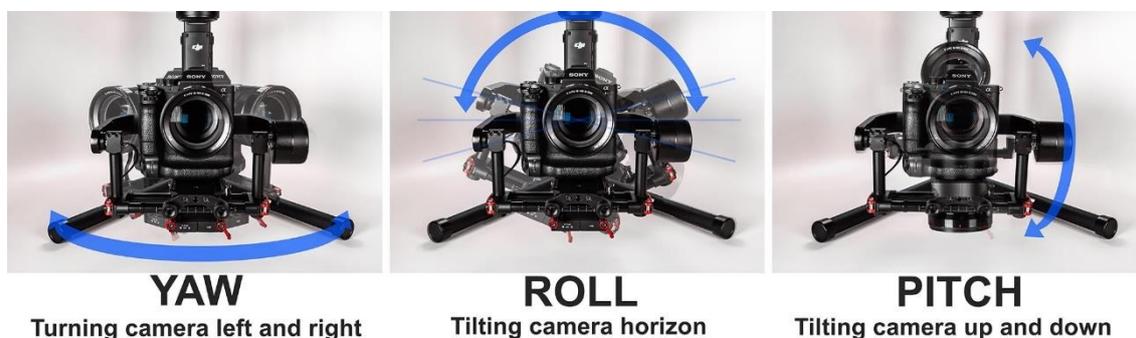
Gimbal

Disposición RPY (Roll-Pitch-Yaw)

A pesar de que los requerimientos para los gimbal son amplios dependiendo de la aplicación, se busca mantener o controlar la Línea de Visión de un objeto acorde con otro objetivo o el espacio inercial (Hilkert 2008). La línea de visión es determinada por el desplazamiento angular del ensamblaje del gimbal. La plataforma de tres ejes del gimbal se conforma de un motor para dirigir cada eje, sensores de posición para la medición del ángulo y un giroscopio (Segarra, Caballeros, Aguilar, Samà, & Rodríguez-Martín, 2019) que mide la velocidad del gimbal. La estabilización del sistema de un gimbal de 3 ejes posee 3 juntas de revolución y ha sido representado como un eje yaw-roll-pitch. Esta configuración se presenta en la (Figura 10).

Figura 10

Disposición RPY gimbal 3 ejes.



Nota: Se explica de mejor manera los ejes del sistema de ejes en un dispositivo Gimbal. Tomado de *Visual and Inertial Data-Based Virtual Localization for Urban Combat* (p.65), por Aguilar, Wilbert G; Calderón, Marco; Merizalde, Darwin; Amaguaña, Fabricio; Tituaña, Jonathan, 2021

Disposición PTZ (Pan, Tilt, Zoom)

El seguimiento de objetos con una cámara PTZ es diferente de un seguimiento de objetos con una cámara estacionaria. Este último pretende mantener el seguimiento de objetivo dentro del campo de visión (Aguilar, Calderón, Merizalde, Amaguaña, & Tituaña, 2020), mientras que el seguimiento por PTZ engloba la dirección de la cámara para que el objeto se encuentre dentro del campo de visión. Los métodos de detección y seguimientos de objetos que hacen uso de diferenciación temporal experimentarán dificultades con un sistema de seguimiento PTZ (Figura 11) al movimiento de la cámara, lo cual causará movimiento aparente para todos los píxeles (Telechanda, Villacís, & Rosales, 2019).

Figura 11

Funcionamiento de Cámara PTZ



Nota: La cámara PTZ tiene como tercer eje el zoom que genera la misma cámara.

Tomado de Control PTZ de un Sistema Electro-Óptico utilizando una Plataforma Robótica (p.12), por Telechanda, B.; Villacís, M.; Rosales, A., 2019

Las cámaras PTZ pueden obtener información de visión de ángulo múltiple con información de resolución múltiple. Estas cámaras son más económicas y se pueden implementar en aplicaciones reales.

Tipos de gimbal

Gimbal de 2 ejes. Este tipo de gimbal (Figura 12) posee dos motores que permiten corregir y estabilizar dos ejes:

- Eje de "Roll" es el movimiento hacia izquierda o derecha: Funciona para estabilizar los movimientos y las fuerzas recibidas en el eje longitudinal del UAV.
- Eje de "Pitch" refiere a la dinámica del movimiento en ir en dirección vertical hacia arriba y hacia abajo. Con esto se puede lograr una estabilización de los movimientos detectados en el eje Transversal

Figura 12

Gimbal de 2 ejes



Nota: A pesar de tener 2 ejes, este gimbal puede generar una estabilidad considerable para la obtención de imágenes claras. Tomado de *Control PTZ de un Sistema Electro-Óptico utilizando una Plataforma Robótica. Telechanda*(p.13), por B.; Villacís, M.; Rosales, A., 2019

Gimbal de 3 ejes. El gimbal de 3 ejes (Figura 13) se conforma por tres motores que estabilizan los ejes de Roll y Pitch, el tercer eje es conocido como Yaw encargado de corregir y estabilizar las fuerzas recibidas sobre el eje vertical del UVA y el eje transversal.

Figura 13

Gimbal comercial de 3 ejes



Nota: Este modelo Gimbal tiene una estabilización más robusta por lo que puede obtener imágenes más limpias. Tomado de *Control PTZ de un Sistema Electro-Óptico utilizando una Plataforma Robótica* (p.14), por Telechanda, B.; Villacís, M.; Rosales, A., 2014

Gimbal de 4 o 5 grados de libertad. Los sistemas gimbal de 3 grados de libertad (Figura 14) alcanzan un volumen dentro del espacio de trabajo en los ángulos Roll, Pitch y Yaw. Al momento que aumentan más grados de libertad se convierte en un robot redundante para conseguir un volumen de trabajo más preciso, esto también aumenta la dificultad del modelamiento matemático y los recursos para el procesamiento del mismo.

Figura 14

Gimbal 4 y 5 ejes



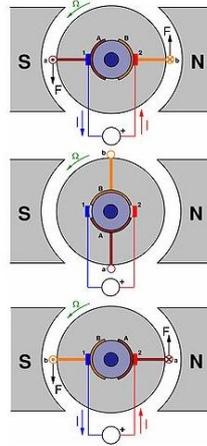
Nota: Al tener más de 3 ejes, estos modelos pueden lograr más puntos en el espacio con más movimientos de sus grados de libertad. *Tomado de Reviews on Various Inertial Measurement Unit (IMU) Sensor Application* (p.35), por Ahmad, Norhafizan, 2013.

Motores

Motor DC. Los motores DC convierten la energía eléctrica en energía mecánica. Está compuesto por el estator (estructura y parte fija) y por el rotor, parte móvil que generará el movimiento. El funcionamiento se basa en bobinas que al energizarse generan un campo magnético permitiendo el movimiento del rotor. El diagrama de funcionamiento se presenta en la (Figura 15)

Figura 15

Funcionamiento motor DC



Nota: Los motores DC tienen como funcionamiento a dos imanes con polos opuestos que generan el movimiento. Tomado de *Development of a Simple and Low-cost Smartphone Gimbal with MPU-6050 Sensor* (p.52), por Rafiq Arif Ainur, 2020

Servomotores. Los servomotores (Figura 16) poseen un circuito para controlar con gran precisión la posición angular, aceleración y velocidad. Poseen un motor DC y un sensor para realizar retroalimentación de la posición.

Figura 16

Diseño servomotor

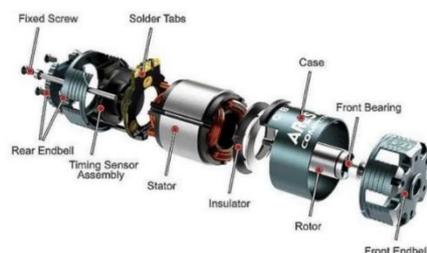


Nota: Los servomotores pueden generar un torque considerable para poder realizar trabajos de alto peso. Tomado de *Development of a Simple and Low-cost Smartphone Gimbal with MPU-6050 Sensor* (p.53), por Rafiq Arif Ainur. 2020

Motores Brushless DC. Los motores Brushless DC son muy utilizados tanto para aplicaciones de consumo como para operaciones industriales gracias a su tamaño compacto, controlabilidad y alta eficiencia. La continua reducción en los costes de materiales magnéticos y dispositivos electrónicos, ha contribuido para aumentar la popularidad de estos, así como el número de aplicaciones de alto nivel y potencia (Delgado, 2013). Los motores Brushless DC son muy utilizados tanto para aplicaciones de consumo como para operaciones industriales gracias a su tamaño compacto, controlabilidad y alta eficiencia. La continua reducción en los costes de materiales magnéticos y dispositivos electrónicos, ha contribuido para aumentar la popularidad de estos, así como el número de aplicaciones de alto nivel y potencia (Delgado, 2013). Un motor Brushless (Figura 17) tiene la característica de no emplear escobillas para el cambio de polaridad, por consiguiente, se reduce la fricción provocando menor temperatura, menor desgaste y mayor rendimiento.

Figura 17

Estructura del motor Brushless



Nota: Los motores Brushless ofrecen un mayor rendimiento debido a su estructura.

Tomado de *CONTROL DE VELOCIDAD PARA MOTOR DC BRUSHLESS SIN SENSORES*, por Delgado, José, 2013

Diseño generativo

El diseño generativo posibilita elaborar, total o parcialmente, diseños mediante el uso de algoritmos predefinidos, es decir, mediante un conjunto ordenado y finito de operaciones para hallar la solución de un problema (Velasco 2014). Varios autores

difieren en la definición de diseño generativo. Celestin Soddu menciona que “El enfoque del diseño generativo trabaja en la imitación de la naturaleza, llevando a cabo ideas como códigos, capaces de generar infinitas variaciones”.

Por otro lado, según Paola Fontana “El proceso de diseño generativo trata sobre el modelado de las condiciones iniciales de un objeto (su genética) en lugar del modelado de la forma final”. Es evidente que los autores basaron el diseño generativo en la arquitectura al arte y sus aplicaciones (Figura 18).

Figura 18

Diseño generativo aplicado a la arquitectura



Nota: El diseño generativo se puede aplicar a varios campos de la ingeniería como la arquitectura, diseño de pieza por citar algunos ejemplos. Tomado de *IndustryWeek*, 2018

El diseño generativo ha sido aplicado en áreas de la mecánica y el diseño de elementos, ya que es de vital importancia la reducción de características como el peso sin comprometer otros factores vitales como la resistencia. Uno de los programas para realizarlo es mediante SolidWorks (Figura 19) mediante un estudio topológico para obtener el resultado final.

Figura 19

Diseño generativo con SolidWorks



Nota: La evolución del diseño generativo se puede visualizar mediante software y escoger la mejor opción, tanto como la construcción como para la visualización.

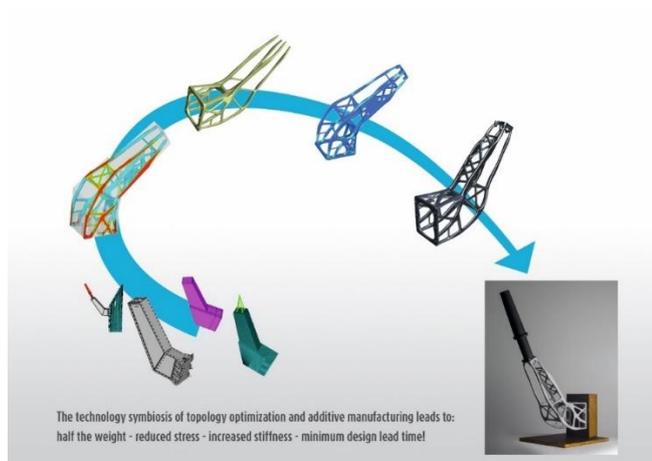
Tomado de *IndustryWeek*, 2018.

Tanto el diseño generativo, como la optimización topológica comparten algunas características en común, y esto ha sido objeto de confusiones entre los investigadores. Estas ambigüedades surgen a raíz de que las “inputs” del proceso generativo son muy parecidas a la de una optimización topológica. (Akella, 2018)

En la década de 1990 es cuando se empiezan a describir los primeros escritos la optimización topológica, sin embargo, este término es menos novedoso si lo comparamos con el diseño generativo (Altair, 2010). En la (Figura 20), se puede observar el proceso de la creación de una pieza mecánica aplicando diseño generativo por parte de la empresa Altair.

Figura 20

Diseño Generativo por Altair



Nota: Mediante visualización de los componentes se puede observar los cambios que éstos generan en el diseño generativo. Tomado de Altair.
<https://web.altair.com/generative-design-report>, 2010

Para poder reducir el material existente en una pieza mecánica, es posible usar la herramienta de optimización topológica, esto claro, manteniendo la función para la que fue creada dicha pieza, sin embargo, tenemos diferencias muy marcadas entre diseño generativo y optimización topológica

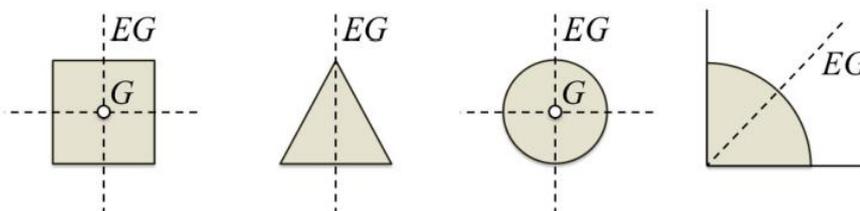
- La optimización topológica tiene como objetivo la mejora de un diseño previamente existente, rediseñar un componente. Por otro lado, el diseño generativo tiene la libertad de crear posibilidades nuevas de diseño.
- La diferencia marcada es que, el diseño generativo “juega” de cierta manera con el proceso de fabricación. Con esto podemos tener una reducción de la tarea de la prueba de productos y tener en cuenta la fase de diseño. (Menéndez, 2019).

Centro de masas

El centro de masa es el punto de un sistema de partículas o de un cuerpo físico en donde se concentra toda la masa, de tal manera que el momento de la masa concentrada respecto a un eje o plano cualquiera fuese igual al momento respecto a dicho eje o plano de la masa distribuida (Vásquez, 2017). Un aspecto importante es que cuando el centro de masas se encuentra en el centro de una figura geométrica, el mismo se anula, esto se evidencia en la (Figura 21). En los casos 1 y 3, cuando el centro de masa no posee centro la desviación, se produce un momento estático generando la inclinación del sólido hacia ese lado.

Figura 21

Centro de masa de diferentes figuras geométricas



Nota: Se puede ver el centro de gravedad de algunas figuras básicas. Tomado de *GEOMETRÍA DE MASAS*(p.35), por Vásquez, Augusto, 2017

Dentro de un dron el centro de masa es indispensable ya que permitirá que el peso se distribuya de manera uniforme hacia los motores (Figura 22), por consiguiente, el gimbal debe mantener su centro de masa en la mitad de tal manera que se anule los momentos estáticos y no afecte al funcionamiento del dron.

Figura 22

Centro de masa gimbal



Nota: Para la mayoría de dispositivos UAV su centro de masas coincide con el centro de sus ejes. Tomado de *Obstacle Avoidance Based-Visual Navigation for Micro Aerial Vehicles*(p.121), por Electronics, 2017

Tipos de control

Controlador PID

El control PID es la estrategia más utilizada actualmente tanto para aplicaciones comerciales como de sistemas de navegación autónoma (Aguilar, Casaliglla, & Pólit, *Obstacle Avoidance Based-Visual Navigation for Micro Aerial Vehicles*, 2017). Durante el último siglo ha sido fundamental en industrias y proyectos académicos, sobre todo en las áreas de reglas de sintonización, identificación de esquemas y técnicas de adaptación (Aguilar, Salcedo, Sandoval, & Cobeña, 2017). Los tres términos de un controlador PID cumplen con los tres problemas más comunes de control (Kanospe, 2006).

Estructura del controlador PID

Comúnmente las estructuras de un controlador PID son: paralelo y en serie:

- **Paralelo:** En esta forma, las acciones Proporcional P, Integral I, o derivativo D, ocurren en términos de ecuación separados con sus efectos combinados y en la suma que es producida (Merizalde, Aguilar, & Calderón, 2020). En

este tipo, cada parámetro es independiente de otros y la ley correspondiente a esta estructura es representada como:

$$u(t) = K_c \left(e(t) + \frac{1}{T_i} \int e(t) dt + T_d \frac{de(t)}{dt} \right)$$

Donde, la ganancia proporcional es $K_p = K_c$, el tiempo integral es T_i y $k_i = K_c/T_i$, el tiempo derivativo es T_d y $k_d = K_c T_d$. Esto también es conocido como una representación ideal (Borase et al. 2021).

- **Serie:** Las series o ecuación interactiva se deriva principalmente desde las características de un circuito electrónico analógico y un circuito neumático. Solo como un PID ideal, K_c impacta en las tres acciones de control, pero las constantes derivativa e integral tienen influencia en la acción proporcional. Este tipo tiene la siguiente ley de control:

$$e_1(t) = e(t) + T_d \frac{de(t)}{dt}$$

$$u(t) = K_c \left(e(t) + \frac{1}{T_i} \int e(t) dt \right)$$

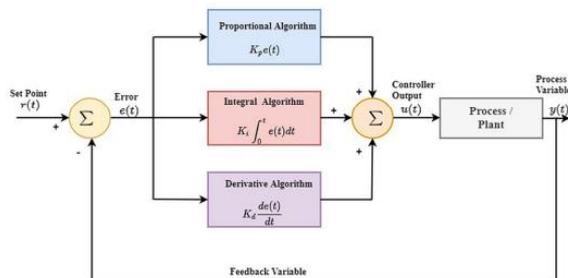
Aquí, la ganancia K_c afecta a las tres porciones de la estructura PID. Por otro lado, los valores de sintonización de integración y derivación T_d y T_i también afectan al término proporcional. Entonces, modificando T_d tiende a afectar las acciones D y P , modificando T_i , impacta en las acciones de I y P , y modificando K_c impacta en todas las tres acciones.

La mayoría de controladores usan solo un algoritmo en serie, principalmente con la forma interactiva. Usualmente es difícil encontrar un modelo perfecto en el mundo práctico (Aguilar, Angulo, & Costa-Castello, Autonomous Navigation Control for Quadrotors in Trajectories Tracking, 2017).

La “prueba y error” son muy usados en procedimientos de sintonización (Borase, Rakesh, Maghade, Sondkar, & Pawar, 2021). Una de las estructuras básicas del controlador PID se visualiza en la (Figura 23).

Figura 23

Estructura de PID



Nota: Esta estructura, es la más usada por las máquinas para el control de sus procesos. Tomado de *A Review of PID Control, Tuning Methods and Applications*(p. 18), por International Journal of Dynamics and Control, 2021

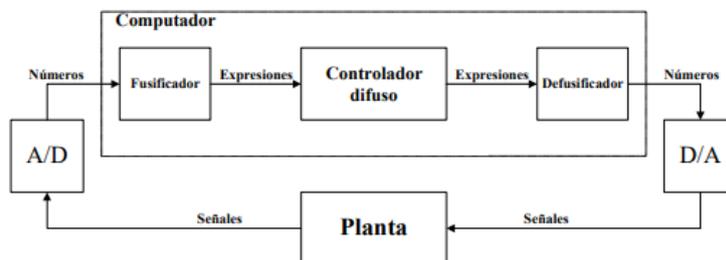
Controlador lógica difusa

La logica difusa es un sistema interpretativo donde objetos o elementos son relacionados con conjuntos de fronteras no nitidamente definidas. A diferencia de la lógica tradicional, en la lógica difusa se otorga un grado de pertenencia relativo graduado según las funciones de membresía (Musalem, 2019).

Para poder realizar el control fuzzy las mediciones se realizan con datos numéricos (excluyendo variables de ceros) por lo que los valores de entrada tienen que pasar por un proceso de fusificación, procesamiento por el controlador y finalmente pasa por el proceso de defusificación. Este proceso se visualiza en la (Figura 24).

Figura 24

Lazo de control difuso



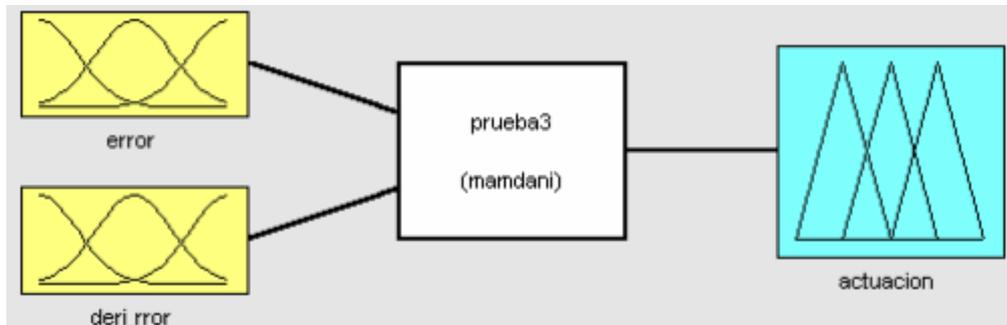
Nota: El control difuso puede reemplazar o complementar a un controlador tradicional. Tomado de *Control Mediante Logica difusa*. <http://www2.elo.utfsm.cl/~elo377/documentos/Fuzzy.pdf>, 2019

El proceso de fusificación a diferencia de una clasificación normal, otorga a los datos de entrada un grado de membresía dentro de los rangos de las funciones, buscando la correspondencia entre las entradas y la función correspondiente. Entre las ventajas del controlador fuzzy, se encuentra que se debe realizar las diferentes reglas para definir el comportamiento de la planta, y para ello es necesario un amplio conocimiento del comportamiento de la planta en lugar de conocer el modelamiento matemático.

Como último paso se realiza la defusificación, que consiste traducir un valor numérico según el rango de salida asignado. Cabe mencionar que se puede tener más de una función de entrada relacionada con una función de salida (Figura 25).

Figura 25

Diagrama de bloques controlador fuzzy



Nota: Gracias a software matemáticos se puede ver la salida o entradas de un controlador fuzzy mediante su implementación. Tomado de *Control Mediante Logica difusa*. <http://www2.elo.utfsm.cl/~elo377/documentos/Fuzzy.pdf>, 2019

La aproximación del control basado en la experiencia humana actúa sobre el control Fuzzy expresando los requerimientos y elaborando la señal de control en términos del lenguaje natural, a través de las reglas *IF-THEN* que pertenecen al grupo de reglas:

...

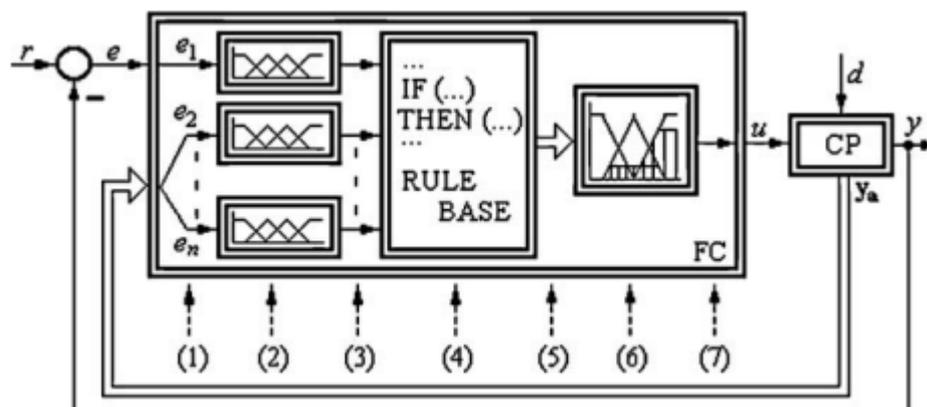
IF(Antecedente) THEN (consecuencia)

...

Donde, el antecedente (premisa) se refiere a encontrar la situación concerniente a la dinámica de control (comparado usualmente con la dinámica deseada/impuesta) y la consecuencia (conclusión) refiere a las medidas que deben ser tomadas. (Precup, 2011). La estructura de una planta con control difuso se puede observar en la (Figura 26).

Figura 26

Estructura básica de control Fuzzy



Nota: La estructura de un controlador fuzzy permite añadir tantas reglas de entrada como queramos para obtener una sola salida. Tomado de *A survey on industrial applications of fuzzy control* (p.213), por Precup, Radu-Emil, 2011

Machine Learning

El “Machine Learning” es una rama de algoritmos computacionales diseñados para emular la inteligencia humana mediante aprendizaje de su entorno. Son considerados, “el caballo de batalla” en la nueva era del Big Data. Las técnicas basadas en el machine learning han sido aplicadas exitosamente en diferentes campos desde patrones de reconocimiento (Galindo, Aguilar, & Reyes Ch, 2019), visión por computador (Aguilar, y otros, 2018), ingeniería espacial, finanzas, entretenimiento, y biología computacional hasta aplicaciones médicas y biomédicas (El Naqa & Murphy, 2015).

La capacidad de resolver avanzados problemas en sistemas computacionales mediante el machine learning, es gracias a modelos analíticos que generan predicciones, reglas, respuestas, recomendaciones, entre otros (Janiesch, Zschech, & Heinrich, 2021).

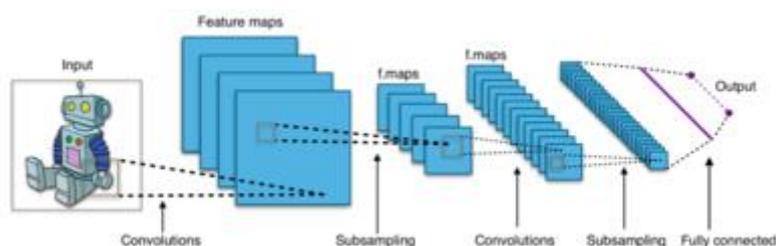
Redes Neuronales Convolucionales

Una red Neuronal Convolutiva o CNN (Convolutional Neural Network) es un tipo de Red Neuronal Artificial que procesa sus capas imitando al cortex visual del cerebro humano para identificar distintas características en las entradas (Ayora, 2012). Las CNN es una red neuronal que ha sido usada para producir un desempeño preciso en tareas de visión computacional, tales como clasificación y detección de imágenes (Aguilar, Quisaguano, Alvarez, Pardo, & Zahira, 2018). Son como las redes neuronales tradicionales, pero con capas más profundas, y tienen sesgos, pesos, y salidas a través de una activación no lineal.

La arquitectura de una Red Neuronal Convolutiva (Figura 27) está compuesta por una capa de convolución, una capa de pooling y una capa de conexión completa. Las capas de convolución y de pooling son alternadas típicamente y la profundidad de cada filtro incrementa de izquierda a derecha mientras que la medida de salida (alto y ancho) van decreciendo. La capa de conexión completa es la última etapa que es similar a la última capa de una red neuronal convencional (Galvez et al, 2018).

Figura 27

Arquitectura de una CNN



Nota: Las diferentes capas de las redes neuronales convolucionales se encuentran ocultas a simple vista para la obtención de un resultado. Tomado de *Object Detection Using Convolutional Neural Networks*(p.2023), por Galvez, Reagan, 2018

La entrada es una imagen que guardará los valores de los píxeles. Posee 3 dimensiones: alto, ancho y profundidad (canales RGB) por ejemplo [50 x 50 x3]. La capa convolucional computará la salida de las neuronas que están conectadas a regiones locales en la entrada. Los parámetros de las capas están compuestos por un grupo de filtros de aprendizaje (o *kernels*) que convolucionan a través del alto y ancho del volumen de entrada extendiendo a través de su profundidad, computando el producto punto entre las entradas de la entrada y el filtro. Esto produce un mapa de activación de 2 dimensiones y como resultado, la red aprende filtros que se “disparan” cuando se detecta alguna característica particular en alguna posición espacial en la entrada. La función llamada Unidad Lineal Rectificada (Rectified Linear Unit, ReLU) está definida como:

$$f(x) = \max(0, x)$$

Esta función es cero para valores negativos y crece linealmente para valores positivos. Cabe mencionar que no afectará el tamaño del volumen. La capa de pooling extrae la activación máxima en una región. Esto muestra las dimensiones espaciales como alto y ancho. La capa de salida es la capa totalmente conectada que es similar a la capa final de una red neuronal, esta capa usa comúnmente la activación softmax para extraer la probabilidad de distribución encima del número de clases de salida.

La función *softmax*, también conocida como la función exponencial normalizada es una generalización multiclase de la función logística. Esta toma un vector z de K dimensiones como entrada y retorna un vector $\sigma(z)$ de dimensión K donde $\sum \sigma(z) = 1$ y donde el valor de cada elemento en $\sigma(z)$ está entre 0 y 1. (Christopher, 2006). La función softmax es comúnmente usada como la última capa en una red neuronal para clasificación multiclase como el vector $\sigma(z)$ representa la probabilidad para las K diferentes clases (Oscar, 2018) , y la ecuación viene descrita de la siguiente manera:

$$\sigma(z)_j = \frac{e^{z_k}}{\sum_{k=1}^K e^{z_k}} \text{ para } j = 1, \dots, K$$

Transferencia de aprendizaje. Es raro entrenar una Red Neuronal Convolutiva desde un script inicial, esto requiere una amplia cantidad de datos para que funcione bien, y entrenar una red neuronal con una cantidad de datos bastante considerable podría tomar semanas a grupos de GPU trabajando a su máxima capacidad.

La transferencia de aprendizaje es una técnica de aprendizaje profundo donde modelos pre-entrenados pueden ser usados para la extracción de características y la sintonía fina (Falconi, Perez, Aguilar, & Conci, Transfer learning and fine tuning in mammogram bi-rads classification, 2020). Esta técnica puede ser aplicada en la clasificación de imágenes como categorización de vehículos, detección de objetos y segmentación. La ventaja de usar este sistema es el corto tiempo de entrenamiento de la red desde el inicio y se necesitan menos datos para obtener buenos resultados. El entrenamiento puede iniciarse usando una unidad de procesamiento central (CPU) incluso sin el poder computacional de una unidad de procesamiento gráfico (GPU), además, en lugar de crear un nuevo modelo desde un borrador, los modelos pre-entrenados pueden ser usados. Estos modelos son entrenados desde una amplia base de datos como Imagenet y COCO (Falconi, Perez, Aguilar, & Conci, Transfer learning and fine tuning in breast mammogram abnormalities classification on CBIS-DDSM database, 2020).

Modelo SSD con MobileNet. Un Disparo multi-caja simple (Single Shot Multi-Box Detector, SSD) es una aproximación a la detección de objetos en imágenes usando una red neuronal profunda. Produce un cuadro delimitador y el porcentaje del objeto detectado a una velocidad más alta que la aproximación previa como el modelo YOLO (You Only Look Once). Por otro lado, MobileNet es un modelo de aprendizaje profundo que puede ser aplicado a varias tareas de reconocimiento como detección de objetos, reconocimiento de puntos de referencia (Aguilar, Álvarez, Grijalva, & Rojas, 2019) y

atributos de rostros. Las ventajas de MobileNet son precisión, rapidez, pequeño y fácil de sintonizar. Está basado en una arquitectura simplificada que usa convoluciones separables en profundidad para construir pesos ligeros para una red neuronal profunda. Los modelos SSD que usan MobileNet en detección de objetos son ligeros por lo que se utilizan en dispositivos móviles y pueden ser ejecutados en tiempo real (Galvez et al, 2018).

SSD usa un set fijo de cuadros delimitadores por defecto usando filtros convolucionales aplicados a mapas de características de diferentes escalas con el objetivo de obtener una alta precisión en las predicciones. En el momento de la predicción, esa red crea ajustes de recuadros para maximizar la forma del objeto produce la probabilidad para la existencia de un etiquetado de clasificación en el recuadro.

MobileNets. MobileNets están basados en una arquitectura alineada que usa convoluciones altamente separables para construir la Red Neuronal Convolucional (Falconí, Pérez, & Aguilar, 2019). Esto resulta en un peso ligero de Red neuronal no solo restringiendo el tamaño del modelo sino también, primariamente la latencia del modelo (tiempo de inferencia). Estos grupos de datos permiten la optimización de hiperparámetros para ajustar la latencia/efectividad dependiendo del problema a la mano. MobileNets han sido bastante efectivas en un amplio rango de aplicaciones, incluyendo la detección de objetos, clasificación, atributos faciales, reconocimiento y geolocalización. (Howard, Zhu, & Kalenichenko, 2017)

Aprendizaje supervisado

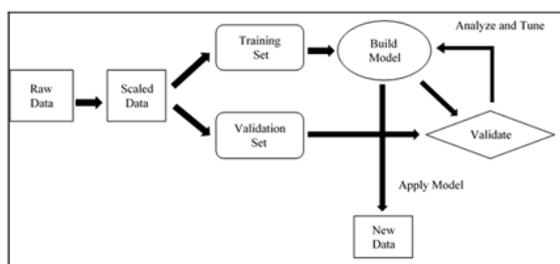
En el aprendizaje supervisado se enseña al algoritmo cómo realizar su trabajo (Aguilar W. G., y otros, Pedestrian detection for UAVs using cascade classifiers with meanshift, 2017), con un conjunto de datos clasificados bajo una cierta apreciación o idea para encontrar patrones que puedan aplicarse en un análisis y producir una salida que ya se conoce. (Manrique, 2019).

Por ejemplo (Figura 28), se considera un problema de clasificación, donde el sistema debe atribuir algunas observaciones experimentales en uno de las N clases diferentes ya conocidas. En este problema, el set de entrenamiento es presentado como una secuencia de pares del tipo $\{(X_1, Y_1), \dots, (X_n, Y_n)\}$, donde X_i son los vectores de entrada y Y_i representa la clase deseada para el correspondiente vector de entrada. La mayoría de los algoritmos de aprendizaje supervisado comparte una característica: se realiza el mantenimiento mediante la minimización de una función de pérdida (particular loss function), que representa el *error de salida* con respecto al sistema de salida deseado.

La mayoría de funciones de salida usadas para este tipo de entrenamiento calcula la *desviación estándar* entre la salida deseada y otra otorgada por el sistema. Después del entrenamiento, la precisión del modelo es medido en un set de ejemplos aparte del set de entrenamiento, también llamado *set de validación*. En esta fase la capacidad de generalización del modelo es verificada: se comprobará *si la salida es correcta* para una entrada sin uso en la fase de entrenamiento. (Pang, Nijkamp, & Wu, 2020).

Figura 28

Flujo de trabajo de aprendizaje supervisado



Nota: El aprendizaje supervisado tiene la ventaja de saber las entradas del sistema las que se están aplicando para poder analizarlas. Tomado de *Deep Learning With TensorFlow: A Review*(p.224), por Pang, Bo; Nijkamp, Erik; Wu, Ying Nian, 2020

TensorFlow

TensorFlow es una librería de software libre, desarrollado por Google Brain Team con Google Machine Learning Intelligence Research Organization. TensorFlow combina el álgebra computacional de las técnicas de optimización, convirtiendo el cálculo de expresiones matemáticas más sencilla.

Las principales características incluyen:

- Definir, optimizar, y eficientemente calcular expresiones matemáticas incluyendo arreglos multi-dimensionales (tensores)
- Soporte en las técnicas de machine learning de redes neuronales profundas.
- Uso transparente del cómputo con GPU, manejo automatizado y optimización de la misma memoria y de los datos usados. Se puede escribir el mismo código y correrlo en la CPU o en la GPU. Más específicamente, TensorFlow se dará cuenta qué partes del cómputo deberían ser movidos a la GPU.
- Alta escalabilidad de computación a través de máquinas y grandes data sets

La computación detrás de Tensorflow es descrita por una *gráfica dirigida*, la cual está compuesta de un grupo de nodos. La gráfica representa un flujo de información de cómputo, con extensiones para permitir, a algunos nodos, mantener y actualizar estados persistentes para un lazo de control dentro del gráfico. (Michael, 2013).

TensorFlow Mobile

Durante el diseño, Google desarrolló TensorFlow para que puedan ser ejecutados en sistemas heterogéneos, incluyendo dispositivos móviles. Esto fue debido a los problemas de envío de datos de vuelta y progresivamente entre dispositivos y centros de datos cuando los cálculos pueden ser ejecutados en dispositivos.

TensorFlow Lite permite a los desarrolladores crear aplicaciones interactivas sin la necesidad de enlazar ni retrasar cálculos de Machine Learning.

TensorFlow Lite

TensorFlow Lite es la evolución de TensorFlow Mobile, el cual ya ayuda con la ejecución de modelos en dispositivos embebidos. Existe una tendencia de incorporar Machine Learning en aplicaciones móviles y como los usuarios tienen altas expectativas en sus aplicaciones móviles en términos de cámara y voz, se ha incrementado la optimización de TensorFlow Mobile a un uso móvil más ligero.

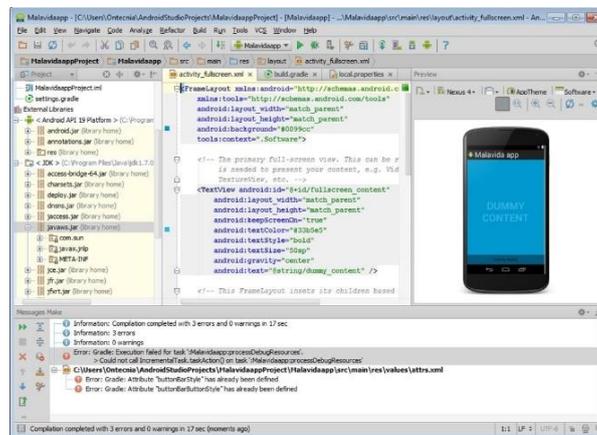
Algunas de las optimizaciones incluidas en TensorFlow Lite son la aceleración de hardware a través de la capa de silicón, frameworks como Android Neural Network API y las redes Neuronales optimizadas como MobileNet (Howard, Zhu, & Kalenichenko, 2017)

Android Studio

Android Studio es un programa que permite desarrollo de app para Android, posee características que lo hacen una herramienta de gran potencia como: sistema de compilación basado en Gradle, emulador de teléfonos, entorno adaptado para todos los tipos de Android. La estructura se evidencia en la (Figura 29).

Figura 29

Estructura del proyecto de Android Studio



Nota: Actualmente los IDE de AndroidStudio se encuentran de manera muy amigable al usuario para lograr un mejor desarrollo de los mismos

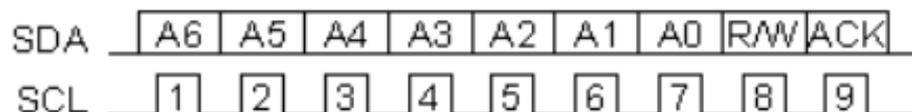
Comunicación

Comunicación I2C

Inter Integrate Circuit (I2C) indica que se realizará una comunicación entre circuitos integrados para facilitar la fusión entre microcontroladores, memorias y otros dispositivos; solo requiere de dos líneas de señal y una común o masa. (Carlletti, 2018). Inter Integrate Circuit (I2C) indica que se realizará una comunicación entre circuitos integrados para facilitar la fusión entre microcontroladores, memorias y otros dispositivos; solo requiere de dos líneas de señal y una común o masa. (Carlletti, 2018). La comunicación I2C se realiza en serie bit por bit y de manera síncrona, el bus consta de tres líneas, la línea SCL indica pulsos de reloj que permiten sincronizar con el sistema, mientras que la línea SDA permite la transmisión de datos entre los dispositivos y GND o masa, que es el común entre los dos dispositivos. En la (Figura 30) se muestra que la señal SDA realiza la transmisión de datos bit a bit mientras que la señal de reloj SCL indica cuando comienza cada bit.

Figura 30

Comunicación I2C.



Nota: Descripción de la comunicación I2C. Tomado de *Comunicacion BusI2Cc*(p.35), por Carletti, Eduardo, 2018

Comunicación radiofrecuencia

La radiación electromagnética (Jara-Olmedo A. , Medina-Pazmiño, Tozer, Aguilar, & Pardo, 2018) es una combinación de campos eléctricos y magnéticos oscilantes que se propagan a través del espacio y transportan energía de un lugar a otro (Acevedo, 2013). El espectro electromagnético es un conjunto de ondas que poseen diferente longitud de onda, entre las que está la radiofrecuencia.

La radiofrecuencia establece un enlace sin cables entre dos puntos para la transmisión de información (Juan, 2019). Para poder comprender de forma clara la radio frecuencia, es necesario entender la radiación electromagnética, pues las ondas de radiofrecuencia abarcan el espectro de longitud de onda desde 3kHz (Murillo, 2012).

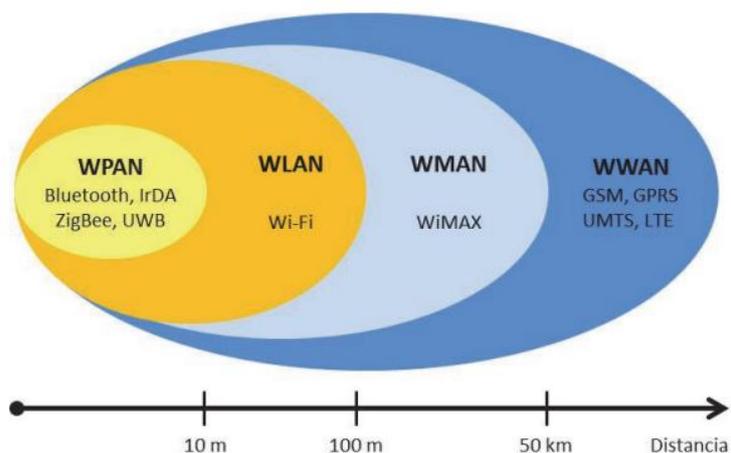
Tecnologías inalámbricas

Las redes inalámbricas es una tecnología que permite comunicar de manera inalámbrica dispositivos diferentes, se pueden clasificar en cuatro grupos específicos (Figura 31) según el área de aplicación y el alcance de la señal: i) redes inalámbricas de área personal (Wireless Personal-Área Networks - WPAN), ii) redes inalámbricas de área local (Wireless Local-Área Networks - WLAN), iii) redes inalámbricas de área

metropolitana (Wireless Metropolitan- Área Networks - WMAN), y iv) redes inalámbricas de área amplia (Wireless Wide-Área Networks - WWAN) (Salazar, 2005).

Figura 31

Redes inalámbricas



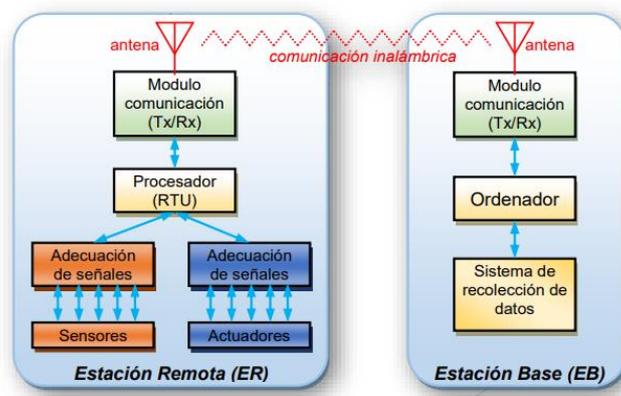
Nota: Alcance de las redes inalámbricas. Tomado de *Redes inalámbricas*(p.152), por Salazar, Jordi, 2005

Telemetría

El término telemetría viene “tele” que significa a distancia y “metron” que significa medida, por lo que es considerada como la ciencia de la recolección de datos mediante una forma remota hacia un lugar (Capraro, 2018). Esta estructura de funcionamiento se puede observar en la (Figura32). La telemetría permite la monitorización y control de datos, y generalmente se la utiliza en zonas de difícil acceso.

Figura 32

Diagrama telemetría



Nota: Tomado de *Comportamiento de una Estación Remota y Estación base Telemetría agrícola*.(p.165), por Capraro, Flavio, 2018

En la actualidad existen diferentes sistemas que requieren de un monitoreo continuo, en varias ocasiones estos sistemas se encuentran separados del cuarto de control ya que la zona a la que están expuestos es peligrosa. Se los puede encontrar en los siguientes casos:

- Redes de suministro: agua, gas, electricidad
- Industria automotriz: carreras automovilistas
- Plantas químicas: procesos peligrosos o tóxicos
- Industria Aeronáutica: aviones de reconocimiento no tripulado

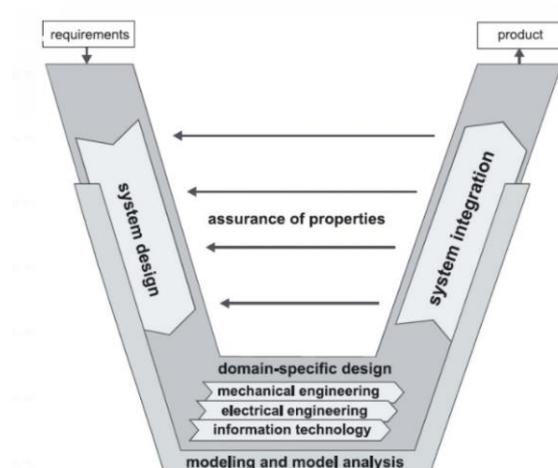
Capítulo III

Diseño mecatrónico

Para el desarrollo de sistemas mecatrónicos se implementó la norma VDI2206 (Figura 33), que genera la sinergia entre el desarrollo mecánico, electrónico y tecnología de la información. Para poder alcanzar el objetivo de la norma es importante seguir los siguientes pasos: i) Modelado, ii) Simulación, iii) Prototipado, iv) Despliegue.

Figura 33

Norma VDI2206



Nota: El desarrollo de un producto mecatrónico mediante la norma VDI resulta de gran ayuda para saber las necesidades del cliente. Tomado de *Diseño y desarrollo de productos* (p.65), por Ulrich, K, 2013

Al ser un sistema mecatrónico posee tres pilares fundamentales: el subsistema mecánico, subsistema electrónico y subsistema de control. Se realizó el diseño de cada sistema posterior, se integró para poder llegar al prototipo final, el desarrollo de cada sistema se describe a continuación.

Subsistema mecánico

El subsistema mecánico está conformado por una estructura tipo gimbal. Se realizó un análisis de modelos comerciales junto con las necesidades y requerimientos mecánicos que deben cumplir. Este análisis se detalla en la (Tabla 1).

Tabla 1

Requerimientos mecánicos

Ítem	Parámetro	Valor
1	Peso	Debe ser ligero y no superara más de 3Kg
2	Resistencia de materiales	Los materiales deben de presentar resistencia a condiciones ambientales
3	Torque	Torque debe de ser lo suficiente para mover los diferentes eslabones de la estructura
4	Distribución	Centro de masa en el centro para no generar momento estático
5	Ensamblaje	Debe de ser fácil de ensamblar y montar en el dron

Nota: Los requerimientos mecánicos del dispositivo se deben cumplir junto con los objetivos planteados

Para poder determinar si los parámetros establecidos en la (Tabla 1) se cumplen satisfactoriamente, es necesario establecer métricas. Las métricas más útiles son aquellas que reflejan, de forma directa, el grado al cual el producto satisface las necesidades del cliente (Ulrich & Eppinger., 2013). Acto seguido se asignó un grado de importancia donde 5 es máxima importancia y 1 menor importancia. Esta metodología se utilizará para realizar la selección de concepto de los diferentes elementos.

Selección de componentes

Al desarrollar el subsistema mecánico se deben de tener varias consideraciones, la primera se centrará en la disposición del gimbal, específicamente al número de ejes que poseerá, los mismo que deben ser impulsado mediante actuadores seleccionados según sus características. El último aspecto es el material para su fabricación.

Selección de configuración de gimbal

La configuración del gimbal delimitará el espacio de trabajo a alcanzar, por tanto, se analizaron 3 conceptos que contemplaron diferentes configuraciones para el gimbal como se muestra en la (Tabla 2).

Tabla 2

Conceptos disposición gimbal

	Concepto	Características	Descripción grafica
A	Disposición (RPY) Roll, Pitch and Yaw	<ul style="list-style-type: none"> • Posee 3 ejes mecánicos • Se necesitan 3 actuadores • El dispositivo aumenta el peso por el tercer eje • El segundo eje nos ayuda para estabilización mas no para el seguimiento • Mas utilizado para estabilización de imagen 	
B	Disposición (PTZ) Roll, Pitch and Zoom	<ul style="list-style-type: none"> • Posee dos ejes mecánicos • Se necesitan dos actuadores • Tercer eje implementado en software 	

	<ul style="list-style-type: none"> • El peso disminuye a comparación de la disposición RPY 	
	<ul style="list-style-type: none"> • Mediante los dos ejes se puede realizar el seguimiento 	
<p>C Disposición Roll and pitch</p>	<ul style="list-style-type: none"> • Posee dos ejes mecánicos • Se necesitan dos actuadores • Con los dos ejes se puede implementar el seguimiento • No implementa ningún eje mediante software 	

Nota: Para cada disposición se tiene unas ventajas y desventajas, entre las cuales aumentan o disminuyen el costo de creación

Se realizó una matriz de selección en la cual se pondera la importancia relativa de los criterios de selección y se enfoca en comparaciones más refinadas con respecto a cada criterio. Las evaluaciones del concepto están determinadas por la suma ponderada de las calificaciones (Ulrich & Eppinger., 2013).

Para poder realizar la matriz de selección, se determinaron los criterios de evaluación presentados en la (Tabla 3). A los mismos, se les asignó una ponderación dependiendo de la importancia. La calificación se lo realizó del 1 al 5 donde 5 corresponde al mejor puntaje.

Tabla 3*Matriz de selección disposición*

Concepto							
		A	B		C		
							
Criterio de selección	Peso	Calificación	Evaluación ponderada	Calificación	Evaluación ponderada	Calificación	Evaluación ponderada
Peso	25%	2	0.5	4	1	4	1
Ensamblaje	15%	3	0.45	4	0.6	4	0.6
Disposición	15%	3	0.45	4	0.6	2	0.3
Costo	25%	2	0.5	4	1	3	0.5
Tamaño	20%	3	0.4	4	0.8	4	0.8
Total	100%		2.3		4		3.2
Lugar			3		1		2

Nota: La puntuación permite la elección de manera objetiva de un modelo que cumpla con los objetivos planteados

Al realizar el análisis mencionado, se concluyó que el concepto B posee mayor puntuación. Este corresponde a la disposición PTZ, una de las grandes ventajas es su disposición ya que posee dos ejes físicos y un eje modificado por software. Se ve una reducción del tamaño respecto al concepto A. Por lo tanto, se logró realizar los siguientes pasos del diseño.

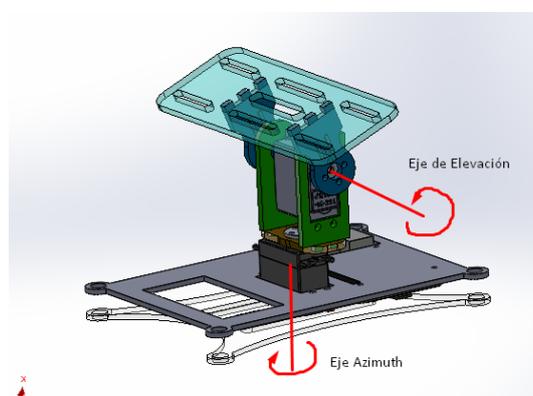
Modelamiento matemático

El Gimbal propuesto, consta de 2 juntas; una junta externa adjunta al eje *Azimuth* y una junta interna adjunta al eje de elevación como se presenta en la (Figura 34) (Wongkamchang & Sangveraphunsiri, 2013). Ambas juntas son controladas por

servomotores de corriente continua. La cámara está montada en el centro del eje de la junta interna. Es imprescindible que cada uno de los ejes sean controlados, la cámara está montada de tal manera que pueda moverse libremente en dos grados de libertad dentro de un ángulo pequeño limitado (Orbea, y otros, Math Model of UAV Multi Rotor Prototype with Fixed Wing Aerodynamic Structure for a Flight Simulator, 2017).

Figura 34

Configuración-Gimbal de 2 ejes

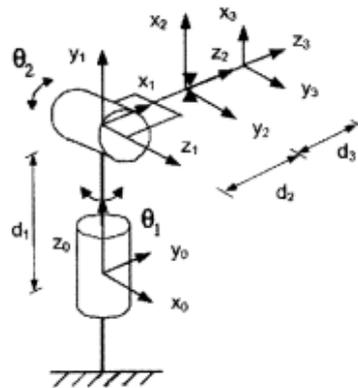


Nota: La elección de dos grados de libertad facilita los cálculos de la cinemática del prototipo

Cinemática directa de Gimbal. Para obtener las ecuaciones cinemáticas del gimbal, se utiliza la convención de Denavit-Hartenberg (Caiza, Aguilar, Albán, & Fernández, 2020), mediante la siguiente imagen demostrativa de las juntas de revolución.

Figura 35

Configuración de juntas



Nota: La simplificación del sistema permite una visualización más detallada de la configuración de juntas. Tomado de *Gimbal Influence on the Stability of Exterior Orientation Parameters of UAV Acquired Images*(p.401), por Gasparovic, Mateo, . 2017

Al realizar el análisis de la cinemática de la (Figura 5) se pudo obtener los resultados los cuales se muestran en la (Tabla 4).

Tabla 4

Tabla de parámetros Denavit-Hartenberg

Junta	α (rad)	a (m)	θ (rad)	d (m)
1	$-\pi/2$	0	θ_1	0
2	$\pi/2$	0	θ_2	0

Nota: Estos valores ayudan a entender de mejor manera los movimientos que cada eslabón puede generar al momento del funcionamiento, además de conocer sus alcances (Aguilar, Angulo, & Pardo, Motion intention optimization for multirotor robust video stabilization, 2017).

Partiendo de la Tabla 4 se realiza la transformación homogénea:

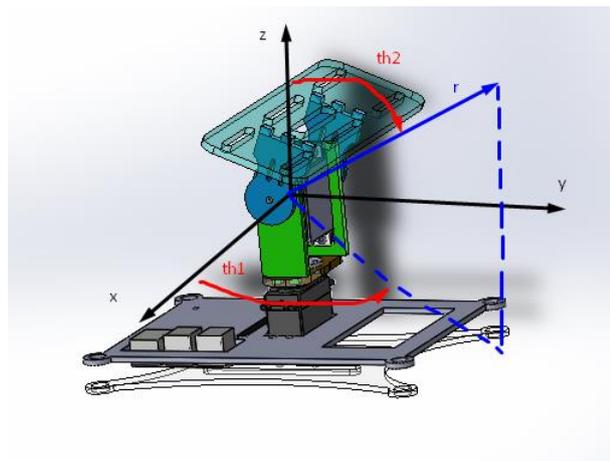
$$T_0^2 = \begin{vmatrix} \cos\theta_1 \cos\theta_2 & -\sin\theta_1 & \cos\theta_1 \sin\theta_2 & 0 \\ \sin\theta_1 \cos\theta_2 & \cos\theta_1 & \sin\theta_1 \sin\theta_2 & 0 \\ -\sin\theta_2 & 0 & \cos\theta_2 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

Donde θ_1 , θ_2 son los ángulos externos e internos, respectivamente

Cinemática Inversa. El problema de la cinemática inversa radica en encontrar las variables de la junta dando la posición y orientación del efector final. Debido a que el centro o el origen de los dos ejes están localizados en el mismo punto (en el centro de masa del gimbal), se puede aplicar un sistema de coordenadas esféricas (Figura 36) (Wongkamchang & Sangveraphunsiri, 2013).

Figura 36

Sistema de coordenadas esféricas



Nota: La cinemática inversa ayuda a la elaboración de parámetros de movimiento a partir de coordenadas básicas del mismo prototipo

Las coordenadas basadas en la estructura fija del gimbal se escriben como:

$$x = r \cos\theta_1 \sin\theta_2$$

$$y = r \sin\theta_1 \sin\theta_2$$

$$z = r \cos\theta_2$$

La cinemática inversa puede ser resuelta como:

$$\theta_1 = \tan^{-1}\left(\frac{y}{x}\right)$$

$$\theta_2 = \tan^{-1}\left(\frac{\sqrt{x^2 + y^2}}{z}\right)$$

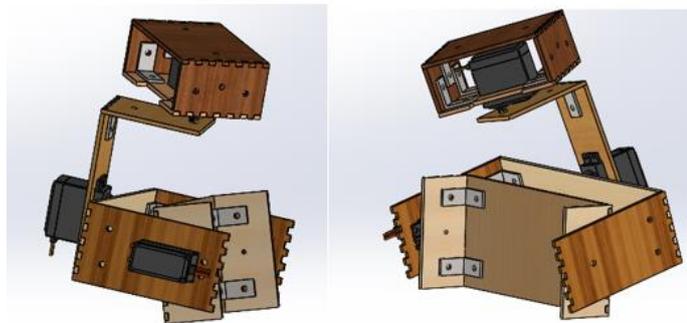
Diseño CAD

Para el prototipo, se llevaron a cabo varias versiones de diseño, los cuales se describen rápidamente hasta llegar al modelo final.

Modelo V1. Esta versión del modelo se desarrolló mediante la aplicación del material de Fibropanel de densidad media o MDF con la finalidad de reducir el peso del modelo final. A continuación, se presenta el ensamblaje (Figura 37).

Figura 37

Versión en MDF



Nota: El torque generado por el mismo peso provocó una sobrecarga de los servomotores que impide el movimiento

Este tipo destaca por ser de modelo de Gimbal de 3 grados de libertad, además de tener la facilidad de ser modular al juntar sus piezas con acoples de metal ajustables mediante tornillos de 4 mm (4M). Al ser un primer prototipo, se buscó economizar costos

y conocer el alcance de visión que tendría el dispositivo cuando se encuentre montado en el dron (UAV) (Orbea, y otros, Vertical take off and landing with fixed rotor, 2017).

Figura 38

Versión MDF montado



Nota: El control del sistema mostrado no era posible debido a las altas vibraciones que el mismo presentaba

EL modelo (Figura 38) tiene la desventaja del peso de cada eslabón de madera MDF, ya que aumenta más el torque que cada uno de los motores debe soportar, por lo tanto, al ser un prototipo demasiado pesado para cada uno de los motores que sirven de grados de libertad, no permite el libre movimiento o al menos alcanzar el rango de movimiento completo de cada uno de los servomotores. Es por esto que el diseño fue descartado, sin antes tener las consideraciones aprendidas en este modelo, tales como:

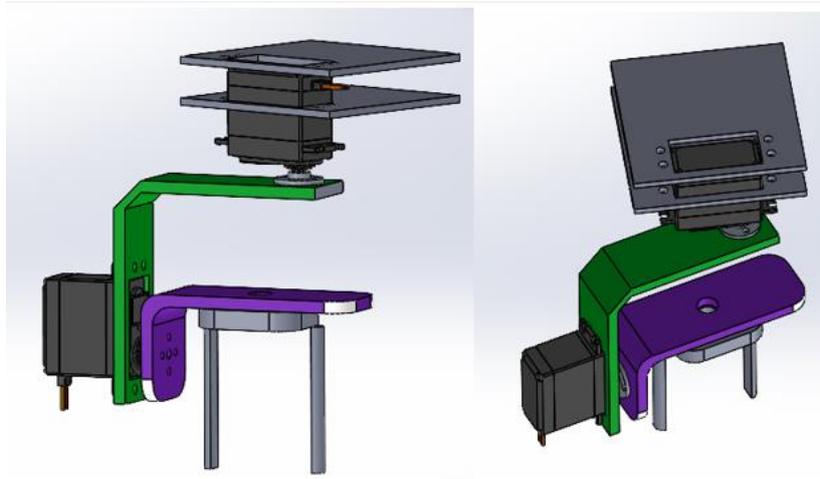
- Tornillos excesivamente a la vista
- Sujeción para celular
- Disminución de material innecesario

Modelo V2. En esta segunda versión se consideró las observaciones del modelo anterior, lo que permitió realizar un modelo más compacto; que no afecte al peso del dron en su mayoría y generar el rango de movimiento completo a todos los grados de libertad. También se cambió el material del prototipo con el fin de evitar el uso de tornillos de sujeción a la vista. El material que se usó es el de PLA, obtenido mediante

Manufactura Aditiva (Impresión 3D). Para esto, se consideró un relleno del 80% para la primera prueba del prototipo. Además de usar un soporte ajustable para celular, comercial, el que pueda ser adaptado mediante un tornillo M8 que pueda ser asegurado al modelo.

Figura 39

Modelo V2

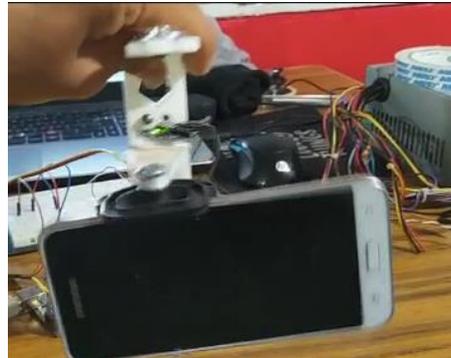


Nota: La sujeción del dispositivo móvil se veía comprometido por los movimientos bruscos que el prototipo pueda generar

Para este nuevo modelo (Figura 39) se redujo a dos grados de libertad, debido a que se decidió usar la cámara del dispositivo móvil (celular) como 3er eje, es decir, se seleccionó la disposición PTZ. Se consideró también que cada brazo se pueda juntar a cada servomotor, permitiendo así la reducción del uso de tornillos de sujeción.

Figura 40

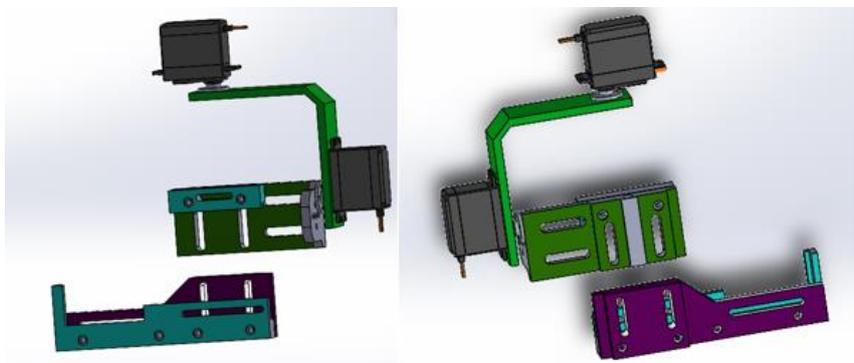
Modelo V2 montado



Nota: El control PID con un sistema de sujeción flojo, puede provocar un cambio en los parámetros de control

El diseño (Figura 40) generó un brazo de palanca elevado tanto en el primero, como en el segundo grado de libertad, lo que disminuyó el rango de movimiento de cada uno de los servomotores. Una de las desventajas de este sistema, es que, el sistema de sujeción no asegura más de un punto de apoyo. Además, el seguro ajustable puede interferir en los botones de encendido/apagado y volumen del celular lo que puede ser perjudicial. Otra desventaja del prototipado en Impresión 3D radica en el tiempo de ejecución, el cual resulta un inconveniente al tener que realizar pruebas.

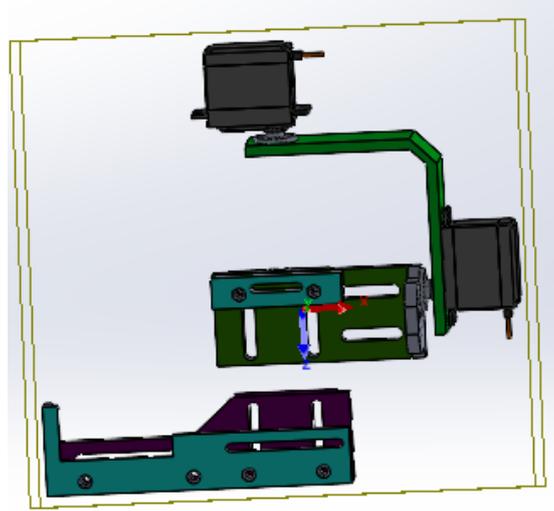
Modelo V3. En este tercer modelo (Figura 41) se consideró tener un seguro inferior, superior y laterales para el dispositivo celular, esto mantiene el diseño de brazos que se juntan mediante tornillos hacia el eje de cada servomotor.

Figura 41*Modelo V3*

Nota: La sujeción mediante tornillos o tuercas aumentaba el peso del diseño para controlar los grados de libertad

Se comprobó que la sujeción hacia el celular es óptima, sin embargo, en este modelo no es posible variar las medidas del celular modelo. Uno de los problemas encontrados, es que se debe utilizar tornillos M4 de una altura de casi 100 mm, con el fin de asegurar el modelo.

El brazo de palanca se mantiene fijo y al aumentar material en la sujeción, aumenta el mismo, reduciendo el movimiento de los servomotores. Otro de los problemas es que, la impresión por capas verticalmente provoca que, al colocar el peso deseado, se desprende de su estructura inicial, generando una posible caída del dispositivo. Sin embargo, con la ayuda del software SolidWorks se comprueba que el centro de masa se mantiene en el centro de todo el prototipo (Figura 42).

Figura 42*Acople de motor V3*

Nota: Cada modelo se encuentra pensado para que el centro de gravedad coincida con el eje vertical del primer motor

Para el siguiente prototipo de diseño, se consideró una disposición PTZ con el eje central como se muestra en la (Figura 43). Este modelo permitirá tener todos los elementos alineados a su eje central, así como el dispositivo a montar, reduciendo así cualquier brazo de palanca que se pueda producir.

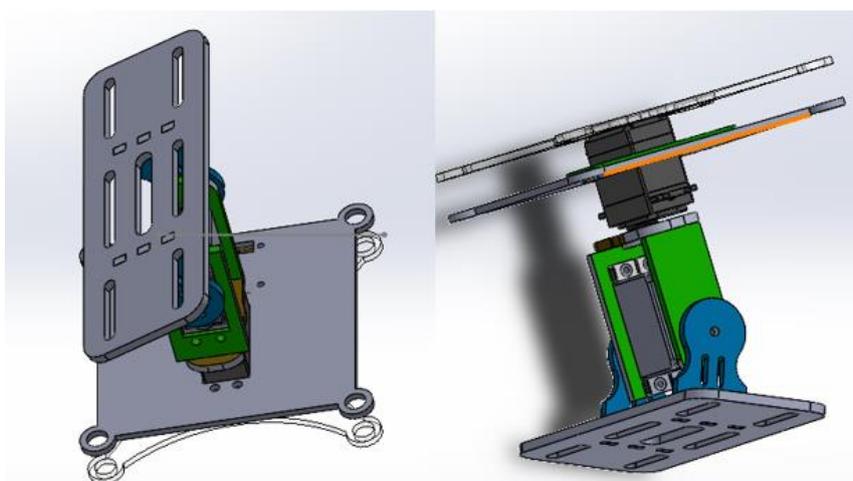
Figura 43*Modelo PTZ*

Nota: El modelo PTZ ayuda a reducir el diseño del mismo prototipo para lograr movimientos suaves

Modelo V4. Para este modelo (Figura 44) se consideró un corte en acrílico de 4 mm que permitirá una impresión rápida de cada uno de los elementos, además de ser en 2 dimensiones, y al realizar el ensamblado, se verificó su funcionamiento de cada uno de los ejes. Para esta versión se consideró tener todas las piezas planas con muescas, las cuales encajan en cada una de las piezas.

Figura 44

Modelo V4



Nota: Es el modelo final que se obtuvo con la experiencia de los anteriores

Este diseño también permitió reducir el peso de la estructura al tener piezas ligeras de material de acrílico, considerando en su mayoría solo el peso de los servomotores y del dispositivo móvil a montar.

Figura 45

Modelo Final



Nota: El movimiento generado en el modelo final puede ser considerado movimientos suaves para la detección de un objetivo

El diseño permite tener una sujeción estable entre el UAV y el prototipo, mediante unos seguros de goma, sirviendo de separadores entre placas. Cada uno de los servomotores, por el hecho de tener acoples junto a los eslabones, mejoran el rendimiento además del movimiento total de su rango (Figura 45).

Análisis diseño generativo

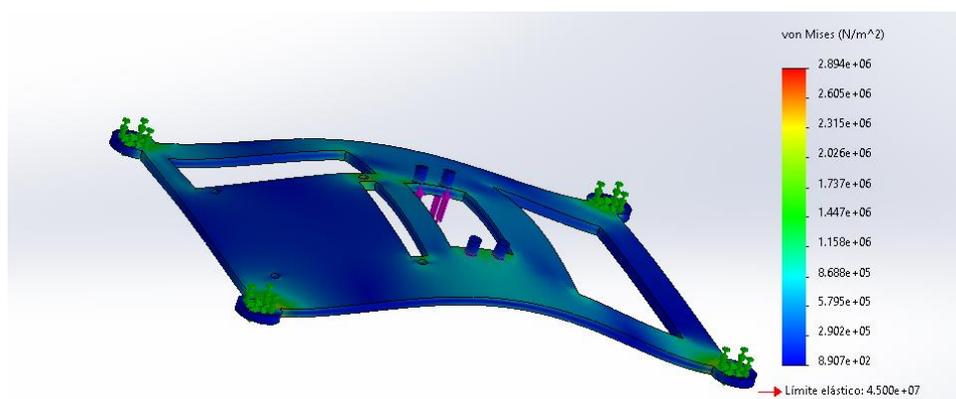
El propósito de realizar un diseño generativo es que, a partir de un análisis estático de fuerzas de cada pieza, reducir la masa de las piezas sin comprometer el comportamiento que éste mismo tendrá en su funcionamiento. Con estas consideraciones sobre la mesa y tomando en cuenta que se trata de un prototipo, es necesario aportar las características mecánicas del material el cual se lo va a fabricar, una opción es el acrílico, sus características lo podemos observar en la (Tabla 5)

Tabla 5*Propiedades mecánicas de Acrílico Estándar*

Material	Acrílico Estándar
Resistencia a la Tensión	72 MPa
Módulo de Tensión	2960 MPa
Resistencia a la Flexión	110 MPa
Módulo de Flexión	3170 MPa

Nota: Las propiedades del acrílico nos podrán ayudar para mediante el software determinar si va a resistir el material

Para verificar la resistencia de todas las piezas es necesario realizar un análisis de resistencia como se muestra en la (Figura 46), para este fin se realizó el análisis estático de cada pieza mostrándolo a continuación:

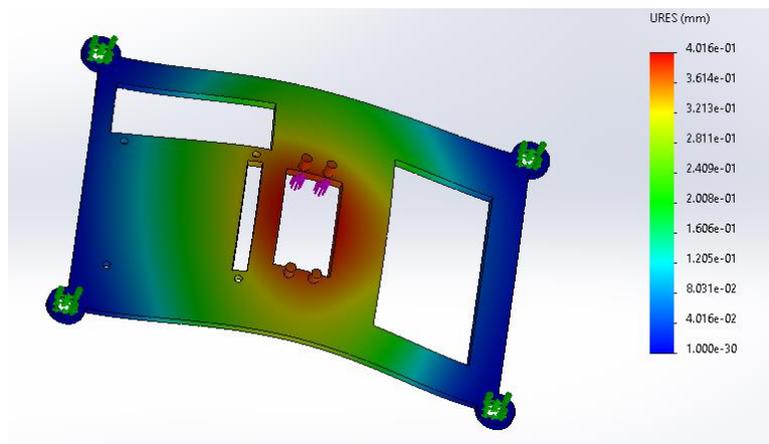
Figura 46*Análisis de Tensiones*

Nota: El análisis de tensiones nos genera la confiabilidad del uso de la pieza mecánica para el trabajo

Se observa en la (Figura 47), que no existe riesgo al usar la pieza manteniendo la geometría original, mientras que en la (Figura 48) de la deformación unitaria la cual nos indica el cambio con respecto a la longitud inicial, no existe variación por lo cual el elemento va a tener una alta resistencia.

Figura 47

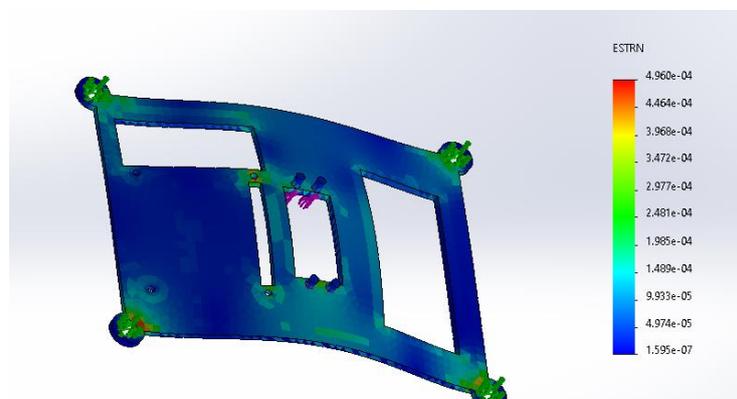
Análisis de Desplazamiento



Nota: El Análisis de desplazamiento ayuda a comprender los puntos críticos de la pieza mecánica al aplicar una carga puntual

Figura 48

Análisis de deformación unitaria



Nota: Esta gráfica nos da un entendimiento del valor de pandeo de la pieza mecánica

A continuación, (Tabla 6) se muestra un resumen del análisis estático de cada una de los elementos mecánicos diseñados, así como su cada resultado obtenido mediante el software de SolidWorks.

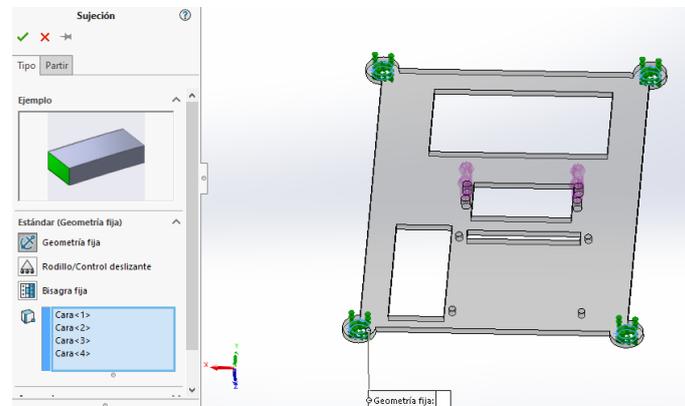
Tabla 6

Resumen-Análisis estático

Elemento	Esfuerzo máximo (Von Mises) [MPa]	Factor de seguridad mínimo	Desplazamiento máximo [mm]
Piso Servomotor	10.53	4.3	1.467
Acople de motor 2	3.104	4	$3.290 \cdot 10^{-2}$
Brazo 1	0.4433	1.2	$3.873 \cdot 10^{-3}$
Brazo 1 Espejo	0.05247	4.6	$8.858 \cdot 10^{-5}$
Acople Espejo Motor 1	0.4852	11	$2.07 \cdot 10^{-3}$
Acople Motor 1	0.4835	10	$2.068 \cdot 10^{-3}$
Base Celular	0.1112	4.6	$1.102 \cdot 10^{-3}$

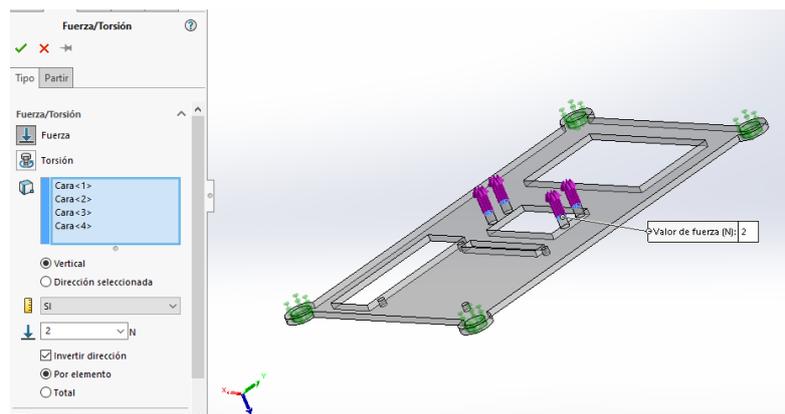
Nota: Para cada una de las piezas mecánicas se puede observar valores que puedan generar una confiabilidad en el funcionamiento del prototipo

Una vez comprobado el funcionamiento de cada una de las piezas, se procede a realizar un Diseño Generativo, que permite decidir si la eliminación del volumen de material es factible o no. Primero, es necesario definir los puntos de Geometría Fija, que serán los puntos de apoyo de cada uno de los elementos del prototipo (Figura 49).

Figura 49*Definición de Geometría fija*

Nota: La geometría fija tiene como finalidad seleccionar los puntos de apoyo de la pieza mecánica

Posteriormente, realiza un boceto para definir la carga externa (Figura 50) que la pieza mecánica estará soportando, en los puntos donde será aplicada la carga.

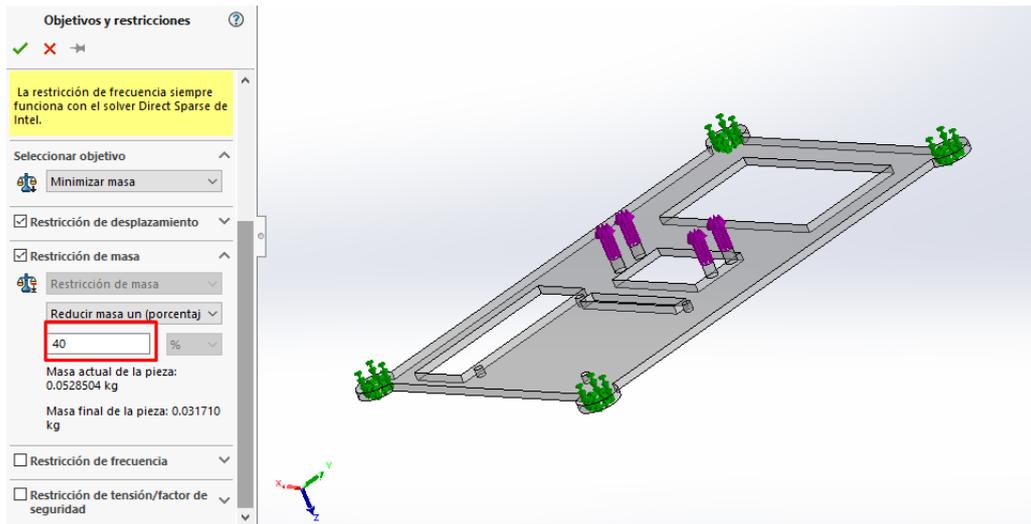
Figura 50*Definición de carga externa*

Nota: Las cargas externas representan el peso que la pieza, en el prototipo, debe soportar

A continuación, se propone el objetivo del análisis, para este fin se plantea reducir el peso de cada pieza a un valor del 30% al 40% (Figura 51).

Figura 51

Objetivo de análisis

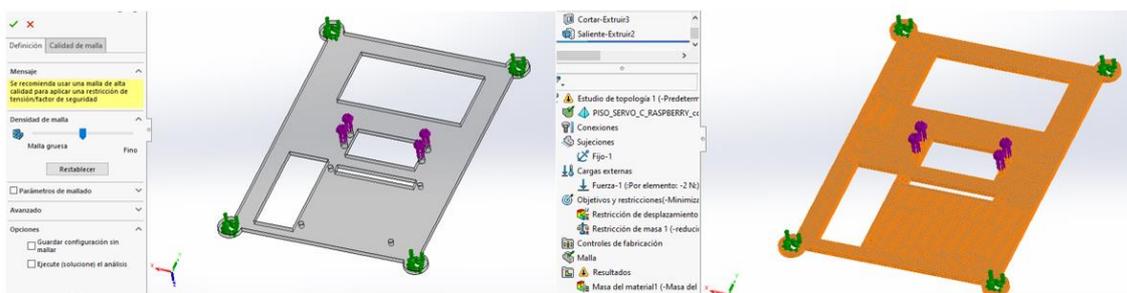


Nota: El objetivo del análisis puede variar según las necesidades del estudio que se requiere

Se define una malla por default entregada por el mismo software (Figura 52).

Figura 52

Definición de malla

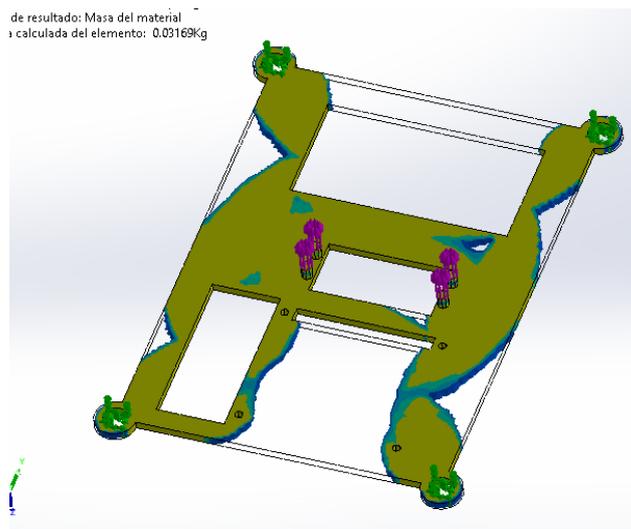


Nota: Dependiendo de las prestaciones de la computadora, se puede generar una malla más o menos fina

En la (Figura 53) se muestra el resultado de cada una de las piezas al igual que la cantidad de material que el software propone reducir para mantener las características mecánicas del mismo.

Figura 53

Diseño Generativo final

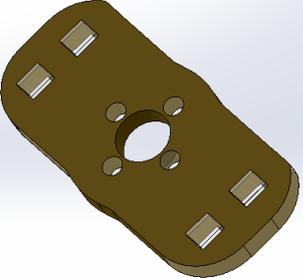
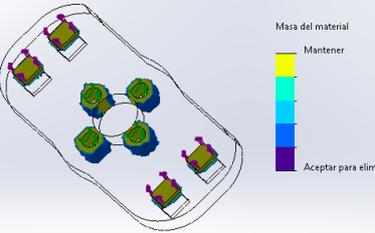
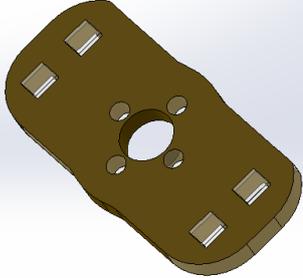
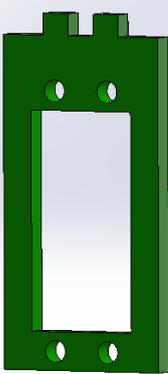
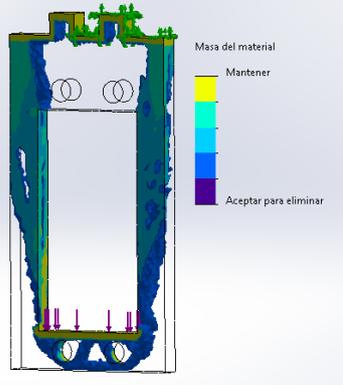
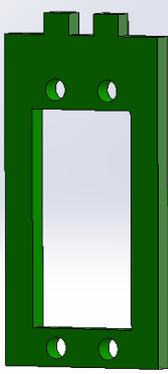
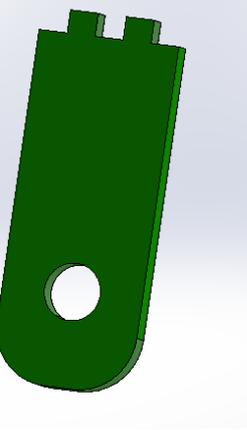
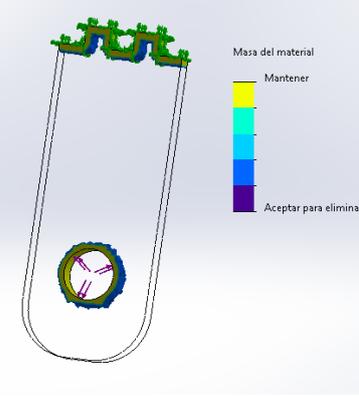
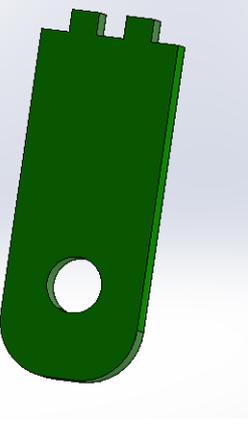


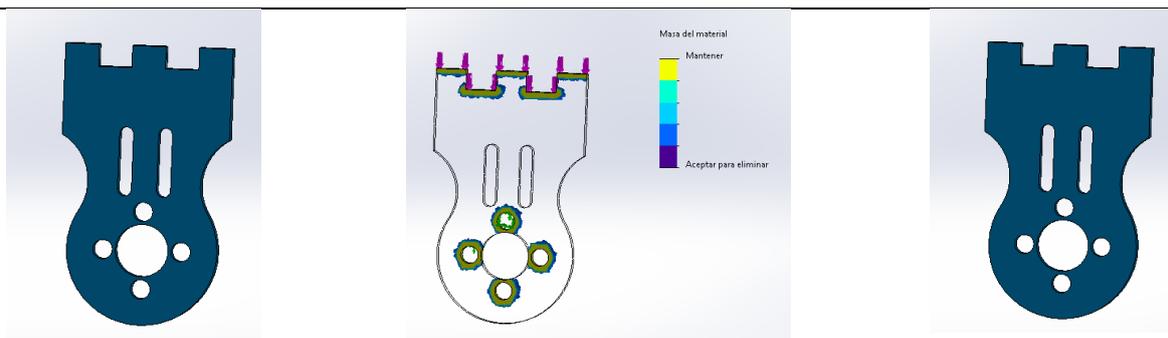
Nota: Las zonas de desgaste pueden o no ser objeto de eliminación, sin embargo, queda a elección de los investigadores

Para cada una de las piezas se decide, por parte de los investigadores, si es posible reducir la masa o no, y de ser posible, seguir los patrones que propone el software para mantener la estabilidad del comportamiento mecánico del mismo. En la (Tabla 7), se muestra un resumen del diseño generativo realizado para cada pieza, mostrando el diseño original el propuesto por el software y el diseño elegido.

Tabla 7

Resumen del diseño generativo por pieza

Diseño original	Diseño propuesto	Diseño final
		
		
		



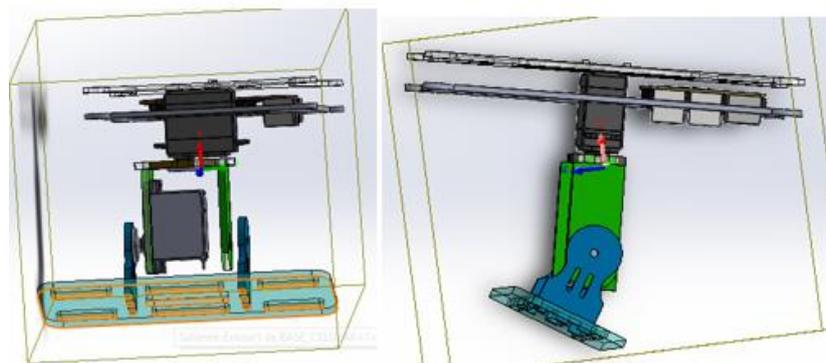
Nota: Los diseños generativos probados por el software producen cambios demasiado drásticos a la geometría del mismo

Análisis del centro de masa

Para el análisis, se consideró, por parte de los investigadores, que el centro de gravedad del prototipo se encuentre en el eje X, coincidiendo en el eje del motor guía, y que el eje Y coincida con el centro de la base del dispositivo celular. Para este fin, se consideró que cada una de las piezas sean simétricas para que el centro de cada una en el ensamblaje, permita que el objetivo sea tener en el centro del prototipo.

Figura 54

Análisis gráfico de centro de masa



Nota: Al ser una geometría compleja, es necesario visualizar el centro de gravedad mediante un software CAD

De acuerdo a la (Figura 54) se tiene el eje X como centro vertical del sistema, por lo que el modelo mantendrá una verticalidad apta para realizar el movimiento sin

desarrollar un torque innecesario que interfiera en el funcionamiento, y reduciendo el tiempo de vida de las piezas mecánicas.

Subsistema electrónico

El subsistema electrónico consiste de componentes encargados del seguimiento de objetivos, dentro de lo cual se considerará la adquisición de imágenes que deberá comunicar a la tarjeta controladora, realizar el control y retroalimentar a los actuadores

Sistema de adquisición de imagen

La adquisición de imágenes es una de las características fundamentales para el desarrollo del proyecto, por lo que se han detallado los requerimientos necesarios representados en la (Tabla 8).

Tabla 8

Requerimientos sistema de adquisición de imagen

Ítem	Parámetro	Valor
1	Procesador	Alta capacidad
2	Resolución de la cámara	Mínimo de 5 Megapíxeles
3	Peso	El menor posible
4	Costo	El menor posible
5	Tasa de velocidad	Alta velocidad de transmisión

Nota: Los valores subjetivos del sistema deben ser objetivos para la realización del mismo prototipo

En base a los seguimientos planteados se determinaron tres conceptos diferentes que cumplan con dichas necesidades, las características de los conceptos se detallan en la (Tabla 9).

Tabla 9

Conceptos adquisición de imagen

Concepto	Características	Descripción grafica
A Teléfono móvil Redmi Note 9	<ul style="list-style-type: none"> • Memoria RAM de 4Gb • Almacenamiento 64 • Cámara frontal de 8 MP • Cámara trasera de 13 MP • Procesador Helio G80 • Sistema operativo Android 10 • Batería de 5020mAh 	
B Web Cam	<ul style="list-style-type: none"> • Generalmente poseen una resolución de 640x480 pixeles • La calidad de grabación se ve afectada por la velocidad el internet • Se utilizan en aplicaciones como video conferencias, video vigilancia • Algunos modelos poseen lente giratoria • Alimentación mediante USB 	
C Arducam	<ul style="list-style-type: none"> • Necesita de una tarjeta lógica como arduino o raspberry • Incorpora interfaz de esclavo SPI • Voltaje de alimentación de 3.3V a 5V • Sensor de imagen de 2MP OV2640 	

- Soporte para lentes de montura
- Interfaz I2C para configurar el sensor
- Posee bibliotecas de código abierto

Nota: Cada una de las opciones de cámara tienen como finalidad la obtención de datos a través de las imágenes

Para la selección del sistema de adquisición de imágenes se considera algunos de los conceptos que necesitan de una placa adicional para realizar el procesamiento de la imagen, ya que, al ser un algoritmo de visión artificial, consumirá recursos al realizarse de manera embebida.

Tabla 10

Matriz de selección sistema de adquisición de imágenes

								Concepto						
								A	B	C				
														
Criterio de selección	Peso	Calificación	Evaluación ponderada	Calificación	Evaluación ponderada	Calificación	Evaluación ponderada							
Procesador	35%	4	1.4	2	0.7	2	0.7							
Resolución	20%	4	0.8	3	0.6	2	0.4							
Peso	10%	3	0.3	4	0.4	4	0.4							
Costo	15%	3	0.45	4	0.6	4	0.6							
Tasa de transmisión	20%	4	0.8	3	0.6	3	0.6							
Total	100%		3.75		2.9		2.7							
Lugar			1		2		3							

Nota: La opción ganadora reduce costos de producción del prototipo

De acuerdo a la (Tabla 10), el primer concepto es el teléfono móvil. Uno de los puntos más importantes consiste en realizar el procesamiento utilizando los recursos del teléfono, por otro lado, un punto negativo es el costo. Sin embargo, al colocar un teléfono de gama media disponible o de uso personal, solo se necesitará la instalación de la apk.

Tarjeta controladora

Después de la adquisición de imagen se debe pasar por el procesamiento y por el algoritmo de visión artificial. Este paso se realiza de manera embebida en el celular por lo que se necesita una tarjeta capaz de controlar los motores. Uno de los aspectos importantes para la tarjeta es la comunicación ya que debe comunicarse mediante el celular y con la estación a tierra.

Tabla 11

Requerimientos tarjeta controladora

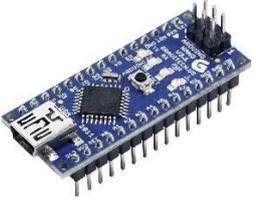
Ítem	Parámetro	Valor
1	Salidas PWM	Mínimo 2 salidas para el control de motores
2	Comunicación	Mínimo comunicación serial
3	Procesador	Capaz de realizar el control de los servomotores
4	Alimentación	5 voltios para utilizar la alimentación para los servomotores
5	Costo	El costo no debe de ser muy elevado

Nota: Los valores subjetivos de la tarjeta deben ser objetivos para la selección de la mejor opción

Según los requerimientos mínimos de la tarjeta controladora (Tabla 11), se analizó las más utilizadas para el desarrollo de proyectos como lo es la tarjeta esp32, arduino y raspberry. Las características de estas partes se observan en la (Tabla 12).

Tabla 12

Conceptos tarjetas controladoras

	Concepto	Características	Descripción grafica
A	Esp32	<ul style="list-style-type: none"> • Alimentación 5V • Comunicación wifia, bluedtood • Pines 36 GPIOs • Ram 520 Kb • Procesador dual core Xtense LX6 • Velocidad del reloj 160Mhz y 240Mhz • 16 puertos ADC • 2 puerto DAC 	
B	Arduino nano	<ul style="list-style-type: none"> • Alimentación 5V • Microcontrolador ATmega328 • Velocidad del reloj 16Mhz • 8 pines E/S analógicos • 22 pines E/S Digitales • 6 salidas PWM 	
C	Raspberry pi	<ul style="list-style-type: none"> • Procesador ARM 11 • RAM 512 MB • Lector de tarjeta micro-SD • Salida mini HMI • 40 pines GPIO 	

- 2 puertos usb
- Comunicación Wifi, Bluetooth
- Alimentación 5V

Nota: Para cada disposición se tiene unas ventajas y desventajas, entre las cuales aumentan o disminuyen el costo de creación

Para poder seleccionar la tarjeta controladora se toma en cuenta la función principal del controlador de los servomotores y la ejecución del seguimiento; la segunda función importante es que permite la comunicación con el teléfono y la estación en tierra. La comunicación se desarrolla mediante socket creado de un servidor en la tarjeta controladora y permite conectar nuestros dispositivos a dicha red.

Tabla 13

Matriz de selección tarjeta controladora

Concepto							
		A	B		C		
							
Criterio de selección	Peso	Calificación	Evaluación ponderada	Calificación	Evaluación ponderada	Calificación	Evaluación ponderada
Salidas PWM	15%	5	0.75	5	0.75	5	0.75
Comunicación	30%	3	0.9	2	0.6	4	1.2
Procesador	30%	2	0.6	2	0.6	4	1.2
Alimentación	10%	5	0.5	5	0.5	5	0.5
Costo	15%	3	0.45	4	0.6	2	0.3
Total	100%		3.2		3.05		3.95
Lugar			2		3		1

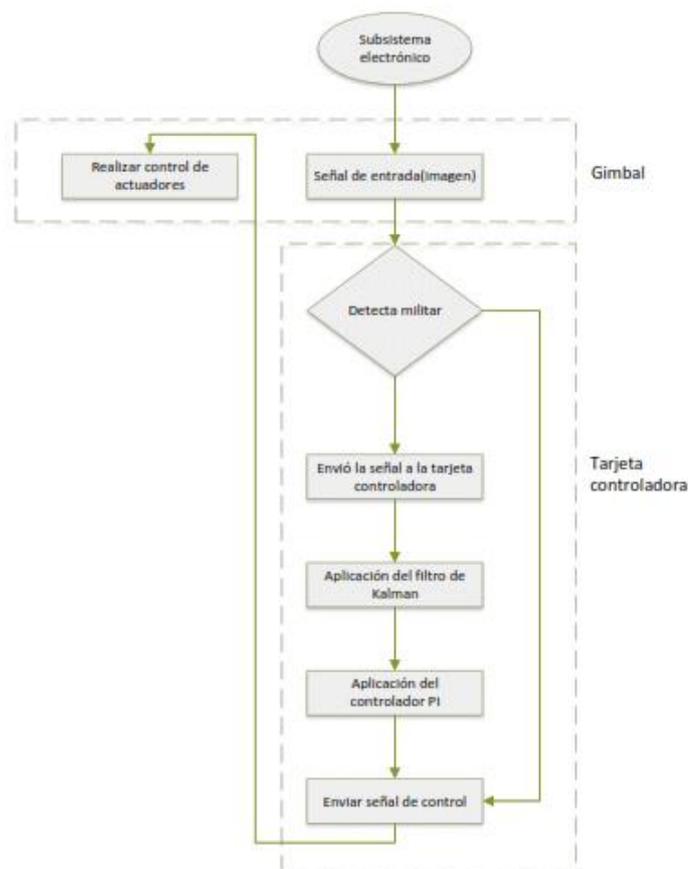
Nota: La selección de la tarjeta controladora será una parte fundamental en el control del mismo prototipo.

De acuerdo al análisis de la Tabla 13, una ventaja es la comunicación para la Raspberry que permite realizar un socket o un servidor, y conectar tanto la comunicación con el teléfono como la estación en tierra.

Después de realizar la selección de componentes en la (Figura 55) se observa el diagrama de flujo el funcionamiento del sistema electrónico que consta de la adquisición de imágenes mediante teléfono móvil, este envía los datos a la tarjeta controladora la cual procesa con el controlador y filtro de Kalman, luego realiza la comunicación para envía las señales de control a los motores.

Figura 55

Sistema de control de prototipo



Nota: El lazo de control presentado se puede observar con las señales a enviar desde el dispositivo móvil

Alimentación

Para determinar la batería del gimbal es necesario considerar los dos aspectos fundamentales: voltaje y corriente. La tarjeta controladora raspberry pi trabaja con un voltaje de alimentación de 5V, mientras que los servomotores trabajan a un rango 3.0-7.2V por lo que se utiliza la misma alimentación de la raspberry.

Dentro de los modelos comerciales de baterías lipo se encuentran los de 1 celda con un voltaje de 3.7V, dos celdas 7.4V, 3 celdas 11.1V, etc. La que más se acerca al voltaje seleccionado es la batería lipo de dos celdas, pero se debe de tener una etapa de reducción para obtener los 5V. Al tener la raspberry alimentación mediante conector microusb se colocó un módulo step down con salida a USB.

Después de analizar el voltaje se ejecuta el análisis de la corriente, es importante considerar que los servomotores consumen en 100mA hasta 1100mA y que la variación en este rango depende del torque que necesite el servomotor; por consiguiente, es necesario conocer el valor del torque y la corriente consumida. Se obtuvieron los datos de manera experimental y sus mediciones se visualizan en la (Tabla 14).

Tabla 14

Mediciones experimentales corriente servomotores

Medición servomotor 1	Corriente (mA)
1	62
2	52
3	48
4	61
5	56
Promedio	55.8
Medición servomotor 2	Corriente (mA)

1	38
2	37
3	23
4	31
5	28
Promedio	31.4

Nota: Podemos observar que tenemos una corriente estable media para cada uno de los servomotores

Figura 56

Medición de corriente servomotores



Nota: La medición de la corriente ayuda a tener una visión más clara del consumo del prototipo

Después de realizar el análisis de corriente como se observa en la (Figura 56), se elabora el dimensionamiento de la batería. La (Tabla 15) presenta los diferentes elementos que consumirán corriente. Se puede obtener la corriente máxima para la selección de la batería.

Tabla 15*Dimensionamiento de la corriente*

Elemento	Corriente(mA)
Raspberry pi	300
Servo 1	56
Servo 2	32

Nota: Las corrientes fueron medidas para la selección de la batería

$$I_{bateria} = I_{raspberry} + I_{servo1} + I_{servo2}$$

$$I_{bateria} = 300 + 56 + 32$$

$$I_{bateria} = 388$$

El análisis de corriente indica que las condiciones mínimas de la batería son 5V y 388mA; comercialmente el valor más cercano en voltaje es de 7.4, por tanto, la corriente se sobredimensiona en caso de que se necesite más corriente para los servos dependiendo del torque o la raspberry. Por tanto, se seleccionó una corriente de 1500mA, el modelo de la batería se presenta en la (Figura 57).

Figura 57

Batería de 7.4V y 1500mA



Nota: La batería Lipo debe tener un control de voltaje por el cual se pueda regular

Subsistema Control

Algoritmo de visión artificial

El algoritmo de visión artificial se entrenó para que detecte a personas, en esta investigación, a personas con vestimenta militar. El primer paso fue la selección del modelo Pre-entrenado, seguido la obtención de la base de datos, creación del archivo TFRecord, la configuración y el entrenamiento.

Modelo Pre-entrenado. TensorFlow nos ofrece una gran cantidad de modelo pre-entrenados que evidencian parámetros como la velocidad de repuesta y el mAP (media de precisión promedio). En la (Tabla 16) se presentan los modelos MobileNet la lista completa se puede consultar en (TensorFlow 2 Detection Model Zoo, TensorFlow 1 Detection Model Zoo), es importante considerar que los modelos fueron entrenados con una tarjeta Nvidia GeForce GTX TITAN X por lo que es probable que no se obtengan exactamente los mismos parámetros.

Tabla 16

Modelos Pre-entrenados

Nombre del modelo	Velocidad (ms)	Precisión mAP	Salidas
SSD MobileNet v2 320x320	19	20.2	Boxes
SSD MobileNet V1 FPN 640x640	48	29.1	Boxes
SSD MobileNet V2 FPNLite 320x320	22	22.2	Boxes
SSD MobileNet V2 FPNLite 640x640	39	28.2	Boxes
SSD ResNet50 V1 FPN 640x640 (RetinaNet50)	46	34.3	Boxes

SSD ResNet50 V1 FPN 1024x1024 (RetinaNet50)	87	38.3	Boxes
SSD ResNet101 V1 FPN 640x640 (RetinaNet101)	57	35.6	Boxes
SSD ResNet101 V1 FPN 1024x1024 (RetinaNet101)	104	39.5	Boxes
SSD ResNet152 V1 FPN 640x640 (RetinaNet152)	80	35.4	Boxes
SSD ResNet152 V1 FPN 1024x1024 (RetinaNet152)	111	39.6	Boxes

Nota: Para cada uno de los modelos disponibles en la red, se debe tener en cuenta la precisión, así como de la velocidad de inferencia

Dentro de las arquitecturas se consideró implementar dos opciones: yolo y ssd, las dos permiten observar una sola vez y realizar la detección. Uno de los inconvenientes fue la falta de actualizaciones de yolo, motivo por el cual se implementó redes ssd. El segundo parámetro de selección y una gran limitante fue la compatibilidad con los móviles debido a que reduce las opciones a utilizar MobileNet. Se observó que hay redes que cumplen con las características por lo se decidió ir por el modelo con resolución de 320x320, esto debido a que para la aplicación debe rápida al momento de generar una imagen más pequeña

El procesamiento se realizó rápidamente, aunque se tuvo que sacrificar un poco la precisión. Un ejemplo de funcionamiento de la red SSD MobileNet V2 FPNLite 320x320 se presenta en la (Figura 58).

Figura 58

Ejemplo modelo pre-entrenado



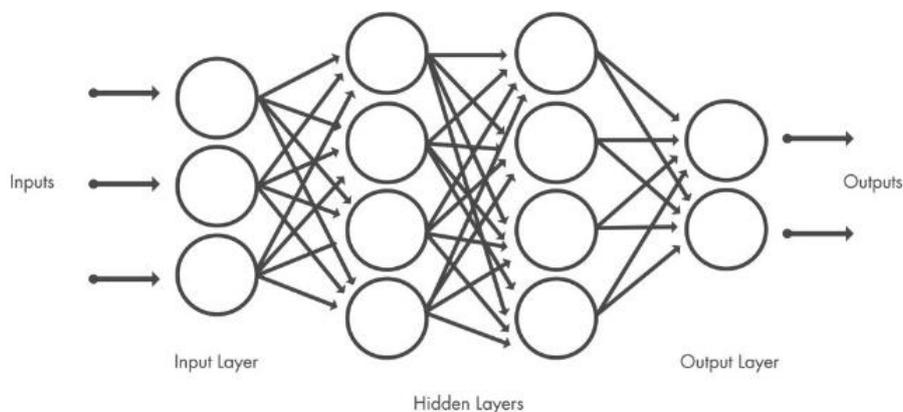
Nota: El modelo pre entrenado puede detectar varios objetos. Tomado de *Diseño de un sistema de reconocimiento automático de matrículas de vehículos mediante una red neuronal convolucional* (p.53), por Sánchez, Agustino, 2020

MobileNetV2 SSD FPN-Lite 320x320. MobileNetV2 es una arquitectura de red neuronal convencional diseñada para ser usada en dispositivos móviles de las tareas de detección y clasificación de imágenes. Está basado en una estructura invertida residual donde las conexiones residuales están entre las capas de cuello de botella. La capa de expansión intermedia usa convoluciones de bajo peso para filtrar sus características como una fuente de no-linealidad. La arquitectura MobileNetV2 completa contiene la capa de convolución completa con 32 filtros, seguido por 19 capas residuales de cuello de botella. (MobileNet, 2014)

Al igual que otras redes neuronales, este tipo de arquitectura está compuesta por una capa de entrada, una capa de salida y muchas capas intermedias ocultas. (Hudson, Hagan, & Demuth, 2014) Esto se observa en la (Figura 59).

Figura 59

Estructura de una red neuronal



Nota: Las capas ocultas de una red neuronal pueden ser vital para el objetivo de detección de objetivos. Tomado de *Deep Learning Toolbox Getting Started Guide*(p.130), por Hudson, Beale; Hagan, Martin & Demuth, Howard, 2019

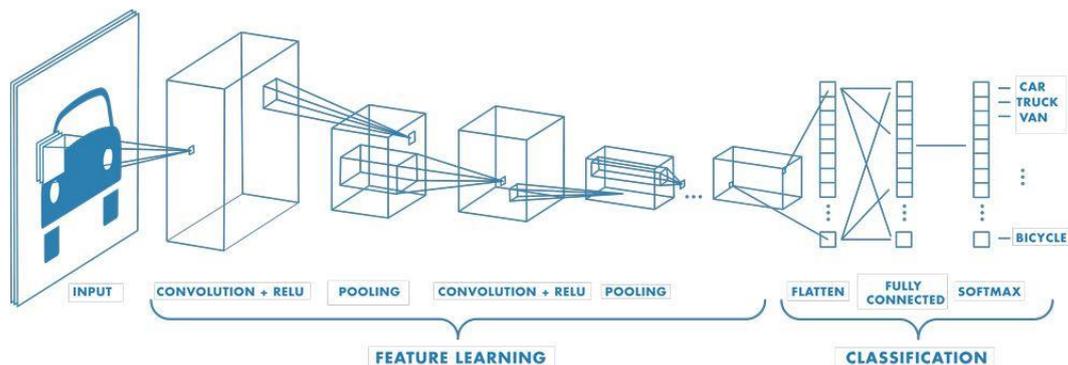
Estas capas realizan operaciones que alteran los datos con el objetivo de aprender características específicas de dichos datos. Las 3 capas más frecuentes son: convolución, activación o ReLU y Pooling

- **Convolución:** Somete las imágenes de entrada a un conjunto de filtros convolucionales, cada uno de los cuales activa ciertas características de las imágenes.
- **Unidad lineal Rectificada (ReLU):** Esta función permite realizar un entrenamiento mucho más rápido y eficaz asignando los valores negativos a un valor de cero.
- **Pooling:** Simplifica la salida al disminuir la tasa de muestreo no lineal, reduciendo así el número de parámetros que la red necesita aprender.

Estas operaciones se repiten en decenas o cientos de capas, de modo que cada capa aprende a identificar diferentes características. Las operaciones de cada capa se presentan en la (Figura 60).

Figura 60

Ejemplo de red con varias capas convolucionales



Nota: Las capas se describen mejor con una visualización de funcionamiento de cada una red CNN. Tomado de *Deep Learning Toolbox Getting Started Guide*(p.135), por Hudson, Beale; Hagan, Martin & Demuth, Howard, 2019

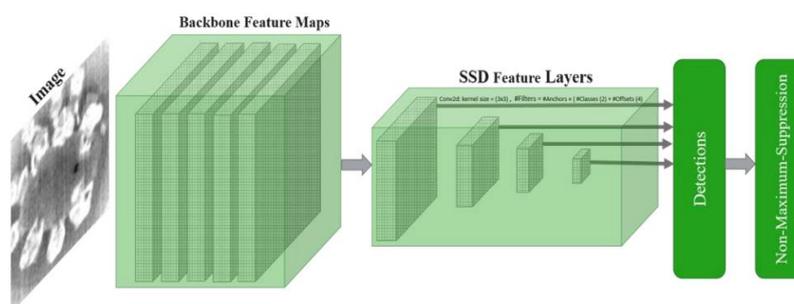
El modelo pre-entrenado MobileNetV2 ha sido entrenado con el dataset de imágenes COCO 2017 las cuales fueron escaladas a una resolución de 320x320. En la versión MobileNetV2 SSD FPN-Lite, tenemos una red base (MobileNetV2), una red de detección Single Shot Detector o SSD) y un extractor de características (FPN-Lite).

Single Shot Detector (SSD). El modelo SSD está basado en una red convolucional de retroalimentada que produce una colección de tamaño fijo de cuadros delimitadores y de puntaje (scores) por la presencia las instancias de clase de objeto en esos cuadros, seguido por un paso de no máximo de supresión para producir las detecciones finales. Las capas más superficiales están basadas en una arquitectura estándar usada para clasificación de imagen de alta calidad (truncada antes de cualquier otra capa de clasificación) el cual llamaremos la red base. El modelo CNN puede ser heredado de un modelo profundo entrenados con conjuntos de datos como COCO y

Imagenet. Las capas principales de SSD predicen la compensaciones y pesos asociados a un conjunto de cuadros delimitados pre-entrenados a diferentes escalas (GreenWaves, 2012). Esto se presenta en la (Figura 61).

Figura 61

Modelo SSD



Nota: El modelo SSD ayuda a la comprensión de las redes neuronales. Tomado de *Artificial Intelligence Techniques Used to Detect Object and Face in an Image: A Review* (p.126), por Deepika, P, 2012

Data set COCO 2017. Microsoft Common Object in Context (por sus siglas en inglés: MS COCO) es un amplio dataset de imágenes de escenas diarias conteniendo objetos comunes en su contexto natural. Las bases de datos como Imagenet cuales contienen millones de imágenes, las mismas que han permitido la investigación en detección y clasificación de objetos (Yi Lin & Maire, 2014).

El dataset fue creado para avanzar el resultado del estado del arte en el reconocimiento de objetos. MS COCO provee las el dataset y las métricas de evaluación para diferentes tareas de visión computacional, permitiendo a los usuarios, evaluar sus modelos de detección en el dataset.

El data set COCO 2017 está dividido en; dataset de entrenamiento, conteniendo alrededor de 118 mil imágenes, el dataset de validación conteniendo 5000 imágenes, y el dataset de prueba conteniendo alrededor de 41 mil imágenes. El dataset contiene

objetos desde 80 categorías para la detección de objetos. Una imagen ejemplo del dataset de validación de COCO 2017 se puede ver en la (Figura 62).

Figura 62

Imagen de dataset de validación COCO 2017



Nota: La imagen representa un caso visible y local de la detección de objetos. Tomado de *Evaluating the sensitivity of lightweight object detection models against adversarial perturbations*(p.109), por Ville, Mayra, 2012

Los modelos de detección de objetos en TensorFlow han sido pre-entrenados con el dataset de COCO 2017 y sus resultados de evaluación están basados en el porcentaje medio de la precisión de las métricas de evaluación en detección de COCO. Esto hace posible el uso del dataset de COCO 2017 y las sus métricas de detección. (Ville, 2012)

Base de datos. Para el presente proyecto se desarrolló el entrenamiento para la detención de personas con vestimenta militar. Se tomó como base el uniforme de camuflaje| del ejército ecuatoriano.

Para la obtención de imágenes se consideraron los parámetros de iluminación, escenarios, distancias, personas civiles y militares. Estos aspectos ayudarán para que la red neuronal no se confunda o se sobre entrene, ya que si se utilizan pocos escenarios la red neuronal se puede sobreajustar y funcionará bien solo para el escenario en el cual se entrenó, sin embargo, no funciona de la misma manera en otros escenarios.

Se desarrollo un algoritmo en Python que posibilita, a partir de un video, extraer imágenes cada cierta cantidad de frames. Posteriormente se realizó el proceso de filtrado y selección procurando extraer las imágenes que no se encuentren borrosas y que aporten información para el entrenamiento de la red como se puede observar en la (Figura 63).

Figura 63

Base de datos de entrenamiento



Nota: La base de datos fue tomado desde varios niveles de luminosidad, además de varios escenarios

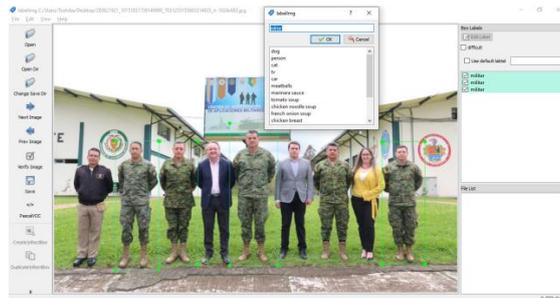
Se realizó una base de datos de 800 imágenes en diferentes escenarios dentro de la Universidad de las Fuerzas Armadas ESPE. Con el objetivo de tener mayor información para el entrenamiento de la red, se realizaron diferentes tomas: militares solos y militares con civiles, la base de datos ImageNet utiliza entre 500 a 1000 imágenes para ilustrar clase (Deng & Dong, 2014), por consiguiente, el rango de imágenes seleccionadas es adecuado, aunque cabe recalcar mientras más imágenes que aporten información mejor será el entrenamiento.

Con las imágenes seleccionadas se inició con el proceso de etiquetado mediante el programa labellmg, que permite colocar un cuadro delimitador del objetivo y su etiqueta dado que es una acción repetitiva se organizan en una lista de las imágenes,

mediante el ratón podemos acceder a comandos directos para agilizar el proceso (Fiedler, 2019) , se seleccionó la etiqueta militar (Figura 64).

Figura 64

Etiquetado de imágenes



Nota: Para el etiquetado es necesario exportar en un archivo compatible con el modelo que se va a entrenar

Después de etiquetar la imagen se guardó como archivo xml, que contiene información de ruta de la carpeta, las dimensiones de la imagen, el nombre de la etiqueta y las coordenadas donde se delimito el objetivo (Figura 65). Este proceso se repitió con cada imagen de la base de datos.

Figura 65

Archivo de etiquetado

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<annotation>
  <folder>I5.dataset militares</folder>
  <filename>504.jpg</filename>
  <path>C:\Users\Toshiba\Desktop\tesis\5. Algoritmo de vision artificial\I5.dataset militares\504.jpg</path>
  <source>
    <database>Unknown</database>
  </source>
  <size>
    <width>3129</width>
    <height>4694</height>
    <depth>3</depth>
  </size>
  <segmented>0</segmented>
  <object>
    <name>militar</name>
    <pose>Unspecified</pose>
    <truncated>1</truncated>
    <difficult>0</difficult>
    <bndbox>
      <xmin>632</xmin>
      <ymin>1532</ymin>
      <xmax>2279</xmax>
      <ymax>4694</ymax>
    </bndbox>
  </object>
</annotation>
```

Nota: Los datos generados por el etiquetado pueden ser visualizados y cambiados en el documento presentado

Creación de TFRecord. Con la base de datos etiquetados se realizó una división destinada a un 80% para el entrenamiento y un 20% para la validación. Este proceso realizó con comandos de Python. Los archivos en xml se transformaron a formato csv (archivo de celdas que indica la posición del objeto el etiquetado y la función que desarrolla). La conversión se ejecuta mediante comandos de Python (Algoritmo 1).

	Conversor	
Algoritmo 1	de xml	A csv

Función	XML_A_CSV(dirección de archivo)
	<pre> clases = [] lista_xml = [] for archivo_xml en glob.glob do arbol = [ET-parse] for miembro en arbol do clases [miembro].insert miembro valor = encontrar nombre de archivo lista_xml.insert valor NombreColumna = [NombreArchivo, ancho, alto, clase, Xmin, Ymin, Xmax, Ymax xml_df = formato(lista_xml columna = NombreColumna end for </pre>

end for

end funcion

Para poder entrenar a la red no es posible ingresar el formato csv, por lo tanto, se transformó a un formato que pueda entender TensorFlow. Este corresponde a tfrecord. Posteriormente se procedió con el entrenamiento.

Configuración de entrenamiento. Después de descargar el modelo pre-entrenado se procedió a descargar el archivo de configuración que posee el mismo nombre con la diferencia que su terminación es en config. Dentro del archivo se modificaron los siguientes parámetros: en un principio el modelo estaba entrenado para noventa clases, al solo tener una clase militar se cambió a 1, el segundo parámetro que se configuró son las rutas de los archivos de entramientos; al tenerlas vinculadas a Google drive, se copió la ruta donde se ubicaban. Finalmente se realizó el mismo procedimiento para los archivos de evaluación y las etiquetas.

El modelo por defecto se configuró para la clasificación. Se consideró el tamaño de imágenes a procesar simultáneamente, denominado el batch size, mientras más grande, se necesitará una mayor cantidad de ram, y al utilizar Google colab la ram permitida es de 12. Al finalizar el tiempo de colab, se generarán puntos de entrenamiento guardados denominados checkpoint que permiten continuar con el entrenamiento. Algunos de los parámetros de configuración se exponen en la (Figura 66).

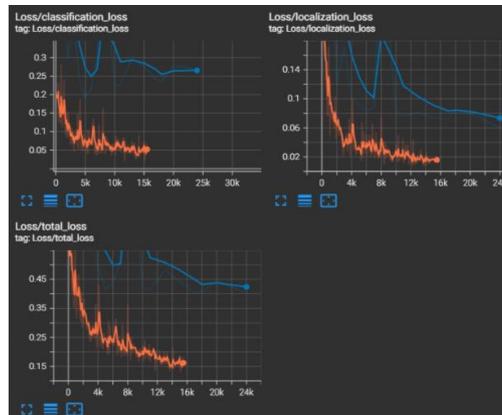
Figura 66

Parámetros de configuración

```
train_config: {
  fine_tune_checkpoint_version: V2
  fine_tune_checkpoint: "/content/gdrive/MyDrive/customTF2/data/ssd_mobilenet_v2_fpnlite_320x320_coco17_tpu-8/c
  fine_tune_checkpoint_type: "detection"
  batch_size: 64
  sync_replicas: true
  startup_delay_steps: 0
  replicas_to_aggregate: 8
  num_steps: 20000
  data_augmentation_options {
    random_horizontal_flip {
      ,
    }
  }
}
```

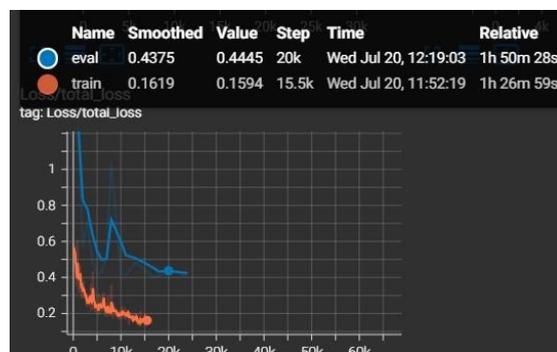
Nota: Existen varios parámetros que son necesarios cambiar para la generación de un entrenamiento óptimo

Entrenamiento del modelo. Con los datos configurados se ejecutó el entrenamiento durante 3 horas, el tiempo fue reducido debido que el modelo ya había sido entrenado anteriormente. Mientras se realizaba el entrenamiento con el tensorflow board se analizaron las gráficas. Se identificaron dos graficas de perdida y de entrenamiento vs la evaluación; mientras la pérdida baja se verificaba que la evaluación también disminuya es decir se busca que la función de error sea mínima (Villada & Muñoz, 2016), esto indicaría que el modelo se estaba entrenando de manera correcta. De acuerdo a la (Figura 67), la pérdida tuvo un descenso, lo que indica que el modelo se entrenó correctamente. Sin embargo, la gráfica de evaluación subió, señalando que el modelo se sobre ajustó. En caso de que la gráfica continuara en ascenso, se tendría que detener el modelo y revisar la base de datos, pero pasos más adelante bajó, por lo que se continuó con el entrenamiento.

Figura 67*Graficas de perdidas*

Nota: La función de pérdida puede ser vital al momento del momento de detener el entrenamiento del modelo

En base a la (Figura 68), la evolución debe bajar al igual que el entrenamiento. Se observa que, aunque se produjo un pico, nuevamente se generó el descenso. Esto indica que el modelo más óptimo se encuentra en el punto más bajo y no necesariamente debe de ser el último punto.

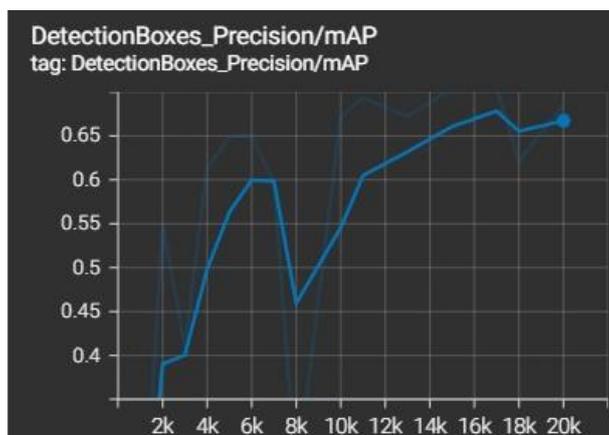
Figura 68*Entrenamiento vs evaluación*

Nota: Las gráficas que muestra una paridad en la evaluación indican un entrenamiento eficiente

Al contrario que la gráfica de pérdida, la ilustración de precisión debe aumentar (Figura 69). Se debe considerar que cuando la precisión es del 100% el modelo se sobreajusta, es decir, aprendió de memoria las imágenes de entrenamiento, pero no será muy efectiva con otros parámetros.

Figura 69

Grafica de precisión del modelo



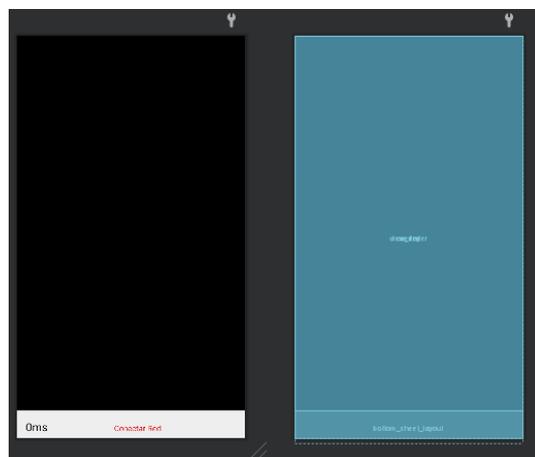
Nota: La precisión, al ir en aumento, genera una confiabilidad al sistema

Diseño de la aplicación

La aplicación procesará el algoritmo de visión artificial y la detección del objetivo. Se desarrollo en Android Studio el cual es un entorno open source lo que ha generado que se mejore la experiencia del usuario y siendo esta una clave del éxito (Mederos, 2020). Se realizó el diseño gráfico de la aplicación: En la pantalla completa se transmite la imagen de la cámara que servirá como entrada al algoritmo para determinar si hay un militar o no, en la parte inferior se encuentra una barra que indica si el teléfono está conectado o no al servidor, al igual que la velocidad de transmisión (Figura 70), se debe considerar los siguientes aspectos para el desarrollo: familiaridad, consistencia mínima sorpresa y recuperabilidad (Muñoz, 2016).

Figura 70

Diseño gráfico de la aplicación

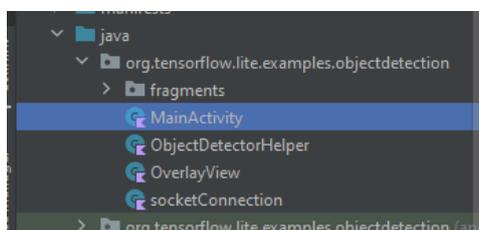


Nota: La interfaz gráfica de visualización en móvil ayuda a verificar la conexión y la velocidad de inferencia del mismo

Es importante considerar que la velocidad de transmisión depende del procesador del teléfono (Amaguaña, Collaguazo, Tituaña, & Aguilar, 2018). En celulares de gama media (utilizado) la velocidad varía entre 90-110 ms, mientras que en teléfonos de gama alta como el Google pixel 4 la velocidad bajo entre 20-50ms. De igual manera en teléfonos de gama baja o con RAM saturada hay una velocidad entre 190-220 ms (Grijalva, Cueva, Ramírez, & Aguilar, 2019). Cabe mencionar que la velocidad de transmisión es de vital importancia para que el controlador responda de manera óptima. En la estructura de la aplicación existen cuatro clases diferentes (Figura 71).

Figura 71

Clases de la aplicación



Nota: Existen varias clases que permiten varias funcionalidades a la misma aplicación

La clase MainActivity es el principal para conceder los permisos a la aplicación y hace uso del gps (Merizalde, Aguilar, & Calderón, 2020), cámara y demás. La segunda clase es una de las más importantes ObjectDetectorHelper, contiene al archivo model_fp16.tflite para el algoritmo de visión artificial entrenado anteriormente. Además, se realizó el post procesamiento para obtener los parámetros de los puntos medios del objetivo y de la pantalla como se presenta en el (Algoritmo 2).

Algoritmo 2 Determinación de puntos medios

if result \neq null **then**

for resultado in result **do**

 BoundingBox = resultado

 arriba = BoundingBox.arriba

 abajo = BoundingBox.abajo

 izquierda = BoundingBox.izquierda

 derecha = BoundingBox.derecha

 x = derecha-izquierda

 y = arriba-abajo

```

        AreaCuadro = x*y
    end for
end if

```

La clase CameraFragmet realiza la conversión de la imagen transmitirla hacia el web server, mientras que la clase socketConnetion desarrolla la comunicación entre la aplicación y el controlador. Estas clases se profundizarán a detalle en el apartado de comunicaciones.

Realizar zoom. El zoom es el tercer eje del sistema implementado mediante software, se lo efectuó en la clase camera Fragment. Para realizar el zoom se analizará el área delimitada por el rectángulo dibujado al detectar al objetivo, además, dependiendo del tamaño del área se hará zoom. La cámara del celular realizar un zoom entre 0 a 1 por lo que se optó que cada cierta área se aumente un zoom de 0.2 hasta llegar al máximo.

En la programación en Android se desarrollaron las funciones zoom y nozoom manejadas a través de la variable linearZoom. Con el objetivo de determinar cierta área, la variable linearZoom se actualiza haciendo un zoom en dos niveles diferentes, cuando el objetivo ya no es detectado el zoom vuelve a su valor preestablecido., el código de la función mencionada se evidencia en el (Algoritmo 3).

Algoritmo 3 Función Zoom

```

if area < 23000 & MilitarDetectado == verdadero then

```

```

    if ContadorZoom1 > 20 then

```

```

        ZoomLineal = ZoomLineal +0.2

```

```

        SetCamara(ZoomLineal)

```

```

        ContadorZoom1=0
    end if

```

```
end if
if area < 81000 & MilitarDetectado == verdadero then
    if ContadorZoom1 > 20 then
        ZoomLineal = ZoomLineal +0.2
        if ZoomLineal < 0.0 then
            ZoomLineal = 0.0
        end if
        SetCamara(ZoomLineal)
        ContadorZoom1=0
    end if
end if
```

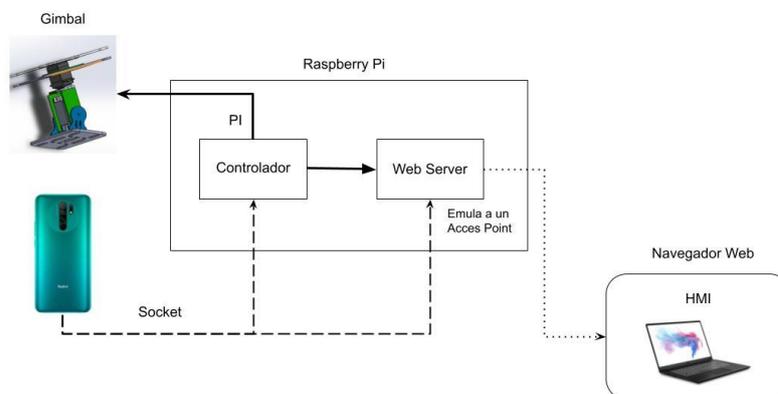
En un principio, el zoom se ejecutó en 5 niveles y funcionaba correctamente, además, cuando la cámara se encontraba de manera estática al realizar el seguimiento, se generaban movimientos pronunciados produciendo pérdida de imagen. Por este motivo se implementó solo dos niveles de zoom, admitiendo ampliar el rango de distancia y poder realizar un seguimiento. La ejecución de las pruebas se evidencia en la (Figura 72).

Figura 72*Pruebas zoom*

Nota: El zoom automático permite una visualización lejana del objetivo

Comunicación

La comunicación se la realizara en dos etapas como se observa en la (figura 73), la primera etapa consiste en una comunicación mediante socket para enviar los datos desde el teléfono hasta el controlador, también se envían los parámetros de video, GPS, valor de certeza al servidor web. La segunda etapa consiste en un servidor web para crear una página web y abrirla en la estación en tierra mediante el puerto 80.

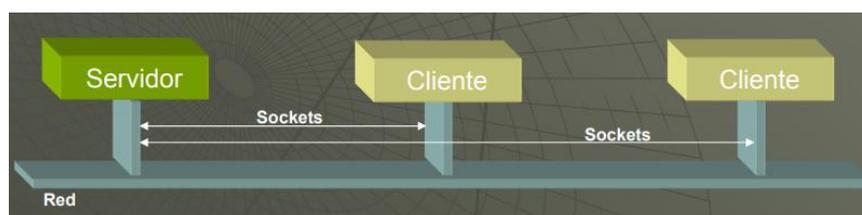
Figura 73*Diagrama de comunicación*

Nota: El diagrama de funcionamiento de la comunicación se puede representar de manera esquemática

Comunicación socket. Un socket realiza la comunicación entre un programa de cliente y uno de servidor a través de una red (Ardila, 2011). Por tanto, la arquitectura utilizada es cliente servidor (Figura 74). El servidor se encuentra a la espera de la solicitud de una conexión mientras que el cliente realiza la solicitud de conexión. Existen tres aspectos importantes para un correcto funcionamiento del socket, el primero la dirección ip que debe encontrarse dentro de una red, como segundo aspecto tenemos el puerto el cual no debe de ser los más utilizados para evitar ataques y el tercero protocolo que se implementara TCP o UDP (Guaraca, 2012).

Figura 74:

Arquitectura cliente-servidor



Nota: Gracias a la arquitectura cliente-servidor, permite la visualización de datos a través de varios clientes. Tomado de *Uso de sockets en la informática: Una Revisión Sistemática de la literatura*(p.2012), por Guaraca, Margoth, 2015

Para la aplicación el servidor se implementa en la raspberry. Se colocó el protocolo UDP ya que envía datos de dúplex en una sola dirección permitiendo que la información llegue rápidamente, está compuesto por una cabecera de 8 bytes seguido de los datos, se considera un protocolo tipo máximo esfuerzo puesto que envía los datos, pero no garantiza que llegue la información (Molina, 2016). El cliente para el sistema será el teléfono móvil que enviará al controlador los datos del punto medio de la cámara y el punto medio del objetivo del objetivo.

La implementación puede darse de dos formas: La primera es orientada a la conexión produciendo una comunicación fiable, sin pérdida de información y se recepta

en el mismo orden que se envía, por otro lado, la conexión orientada a no conexión envía datagramas de tamaños fijos, no es fiable y puede existir pérdidas de información. Para la presente investigación se implementó una orientada a la no conexión, se sacrifica un leve porcentaje de fiabilidad, pero el control podrá reaccionar rápidamente (Gómez & Machado, 2017).

Servidor. Para poder definir un socket es necesario tres componentes: una dirección IP, un protocolo y un número de puerto. Un servidor comienza a escuchar en el puerto asignado, el cliente que conoce la IP o el dominio solicita una conexión con el servidor, el servidor acepta la conexión y se puede decir que se encuentran conectados (Hernández U. , 2015). Para la creación del servidor se implementó en la tarjeta controladora Raspberry, se utilizó la comunicación TCP/IP mediante el puerto 4000, se definió una IP virtual como 127.0.0.1; mediante estos parámetros se programó el servidor.

Figura 75

Código implementación servidor raspberry

```
from _thread import *
import socket
import time
import requests

from flask import Flask
from flask import jsonify
```

Nota: Las librerías se encuentran en repositorios en línea

En la (Figura 75) se muestran los comandos principales para implementar el servidor, entre ellos se encuentran:

- `import socket`: importa la librería socket para realizar la comunicación

- `server_socket_app = socket(socket.AF_INET, socket.SOCK_STREAM)`: crea el objeto tipo socket para desarrollar la comunicación, entre los paréntesis se debe indicar el dominio, el tipo y el protocolo utilizados.
- `server_socket_app.bind(('', port_app))`: se debe especificar la IP en la cual se implementará el servidor. En este proyecto sería la IP de la raspberry y el puerto mediante el cual se comunicarán.
- `server_socket_app.listen(5)`: coloca al servidor escuchando alguna petición de los clientes. Para la implementación del código de comunicación se puede observar el (Algoritmo 4).

Algoritmo 4 Servidor en Raspberry

```
tamano_buffer = 104

puerto = 4000

URL = http: //127.0.0.1:80/controler

while verdadero do

    Imprimir("puerto" + número de puerto)

        if dato es verdadero then

            while verdadero do

                dato = ' '

                    dato = ' dato desde buffer'

            end while

        else if dato == falso then

            break

        end if

    end while
```

Cliente Android. Después de realizar el servidor sigue la creación del cliente, el cual se desarrollará en AndroidStudio para ser implementado en la misma apk de visión artificial. Android tiene muchas ventajas como apertura, sin límite de aplicaciones, rápida modificación entre otras, lo cual lo convierte en una buena opción para la implementación de la arquitectura (Pang & Jiang, 2010).

Figura 76

Implementación cliente AndroidStudio

```
class socketConnection {
    var isConnected=false
    var host = "192.168.1.13"
    var port = 4000
    public final var mSocket= Socket();
    val inetAddress: InetAddress = InetAddress.getByName(host)
    val socketAddress: SocketAddress = InetSocketAddress(inetAddress, port)
```

Nota: La implementación de cliente es posible gracias a la escritura de una ip fija

El envío de datos se realizará siempre y cuando la aplicación detecte la conexión del teléfono (Figura 76). Luego de verificar la dirección, se envían los datos de los puntos medios en las coordenadas en X y Y de la pantalla y del objetivo como lo indica el (Algoritmo 5), estos datos sirven para realizar el controlador. El proceso de comunicación del cliente consiste en los siguientes pasos, abrimos el socket, creamos las entradas y salidas para la comunicación, realizamos la comunicación y cerramos el socket (Joby, 2016)

Algoritmo 5 Envío de datos al servidor

if SocketEstaConectado **then**

Mensaje = MidX+", "+MidY+", "+Ancho+", "
+Alto+", "+Score+", "

```

+CoordIzq+"", "+CoordDer+", "+
CoordArriba+", "+CoordAbajo+", "
Socket.Enviar(Mensaje)
Conectado = verdadero

else if then

    Socket.Conectar

    Conectado = falso

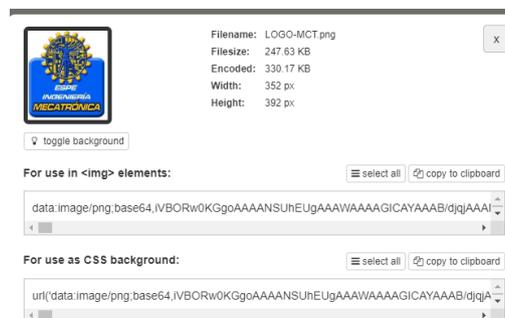
end if

```

Ciente HMI .Para la comunicación con el HMI que se encuentra en la estación en tierra se realizan dos procesos: En el primero, el teléfono enviará los datos mediante socket directo al web server, para lo cual se configurara una IP y el reenvío de puertos lo cual permitirá un acceso desde otro punto el cual deberá estar conectado a la red (Runia, 2015), para poder enviar los datos desde el teléfono hasta el web server se debe transformar los datos a base 64 (Figura 77). Dicha variable se obtiene con el mapa de bits y se convierte en un arreglo de bites, posteriormente se convierte el arreglo en base 64 (Borrayo, 2017).

Figura 77

Conversión de imagen a base 64



Nota: La conversión de imagen ayuda a una visualización fluida de los datos en tierra

Los datos en base 64 se enviarán al servidor mediante un socket cada 10 frames, este proceso se ejecuta cierto tiempo para no saturar a la aplicación. Una de las características que se intenta reducir al máximo es la latencia la cual indica el tiempo desde que se realizó la petición y el tiempo que se comienzan a visualizar los resultados (Murillo , 2010). Para realizar el envío al servidor se utiliza el comando sendhttp, los datos son emitidos en formato JSON (string de bajo peso para la transmisión de datos) el cual es un formato ligero para el intercambio de datos de fácil lectura y escritura para los seres humanos (Changi , 2011) . Finalmente, se envía el nombre y el valor. La ventaja es que se pueden enviar diferentes tipos de datos en un JSON (Figura 78), dentro del formato se despacharán los datos de la imagen en base 64, los datos de latitud, longitud y valor de certeza.

Figura 78

Formato JSON

```
'{"name": "John", "age": 30, "car": null}'
```

Nota: El formato JSON permite tener una estructura de librería aplicable a Python

Para la implementación del web server se utilizó la librería Flask la cual nos permite crear un entorno de web estructurado en Python, ofreciendo un sistema de manejo de sesiones, enrutamiento y conexiones a diferentes bases de datos (Castillo, 2020). Se usó esta librería para lanzar una página web que se requiere de un dominio y host, es decir, la librería simula un servidor en la nube permitiendo la comunicación mediante HTTP. Para poder realizar el ingreso de datos se utiliza el comando POST mientras que para extraer los datos el comando GET (Figura 79)

Figura 79

Métodos HTTP

```

@app.route('/controler', methods=['POST'])
def controler():
    response = {'message': 'success'}
    req=request.json
    global data_from_phone
    if(req['data']!='NODATA'):
        data_from_phone=req['data']
    print('data_fp:',data_from_phone)
    return jsonify(response)

@app.route('/detector',methods=['GET'])
def detector():
    response = {'message': data_from_phone}
    #print(jsonify(request))
    return jsonify(response)

```

Nota: Los métodos HTTP permite encapsular datos y poder enviar mediante sockets

Diseño del controlador

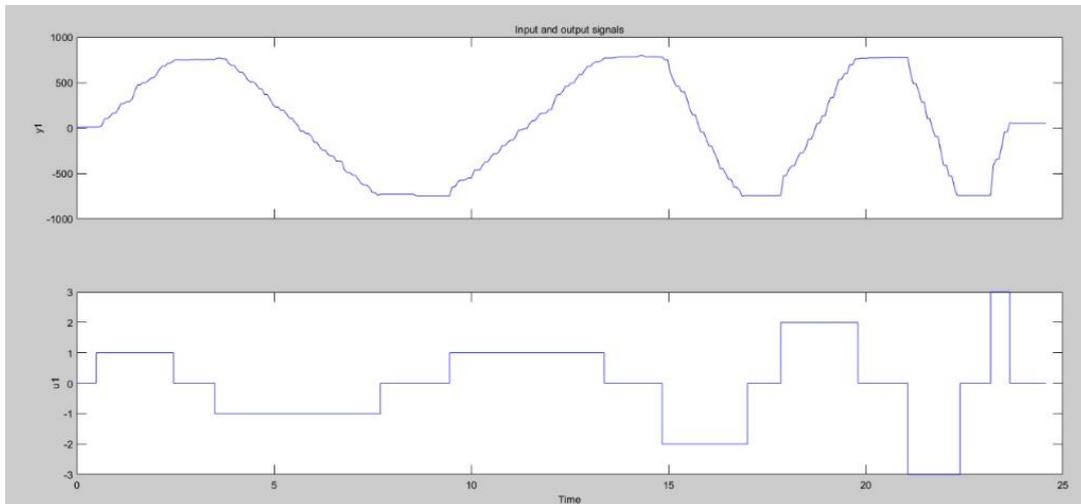
Para el diseño del controlador, al ser un sensor óptico, es necesario obtener los datos que el prototipo genera, es decir, generar movimiento en los servomotores que formarán una tasa de cambio de píxeles con relación al centroide de la imagen del objetivo detectado y calculado previamente. Para estos cálculos se hace uso del Software Matlab, y de su herramienta System Identification Tool.

La identificación del sistema es un término para un problema de estimación del modelo para sistemas dinámicas en la comunidad del control. Dos grandes avenidas pueden ser vistas para el desarrollo de la teoría y la metodología (Gevers, 1993)

Una característica principal de los sistemas dinámicos es que el futuro depende del pasado. Tomando una predicción de la salida $y(t)$ al tiempo, ambos siendo construidos por razonamiento o cuidadosamente calculado en un marco estocástico, depende de todos o de algunas entradas medidas previas y salidas. (Ljung, 2008). La toma de datos lo podemos observar en la (Figura 80).

Figura 80

Toma de datos de comportamiento de planta

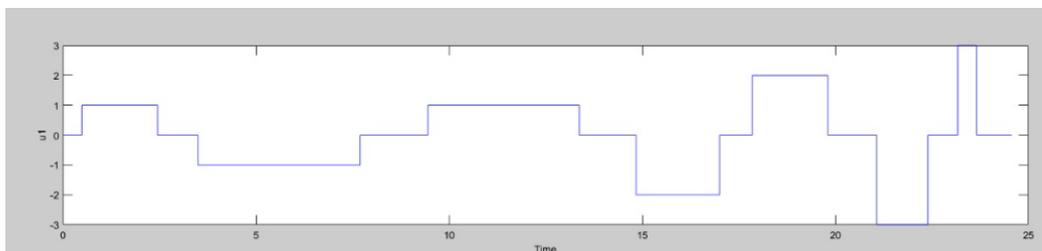


Nota: Los datos de entrada y salida pueden ser leídos por un DAQ para la interpretación de un software de identificación de planta

En la figura 81 se observa el movimiento de los servomotores de izquierda a derecha, además de generar una velocidad más rápida en cada uno de los movimientos. Se muestran así 3 niveles de velocidad.

Figura 81

Movimiento de servomotores

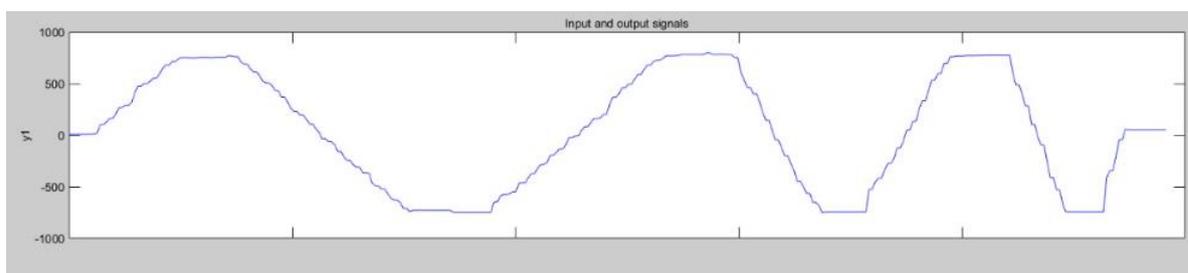


Nota: Los servomotores describen un movimiento de izquierda a derecha con un alza o baja en la velocidad

En la salida se tiene la tasa de cambio de píxeles, es decir un dato de *delta-píxeles*, ($\Delta píxel$). Estos representan el comportamiento al variar los números de píxeles en cada una de las iteraciones obtenidas, medidas desde el centroide de la imagen detectada. El comportamiento lo podemos observar en la(Figura 82).

Figura 82

Comportamiento de píxeles

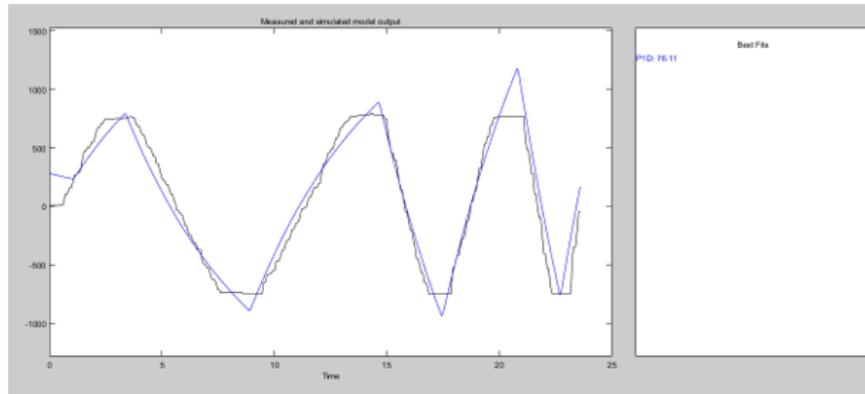


Nota: El comportamiento de los píxeles es el esperado con respecto al movimiento de los servomotores

Una vez obtenido el comportamiento de la planta, se procede a identificar la planta, obteniendo, con la ayuda de software

Figura 83

Estimación numérica de la planta



Gp =

$$\frac{230.2}{s + 0.7484}$$

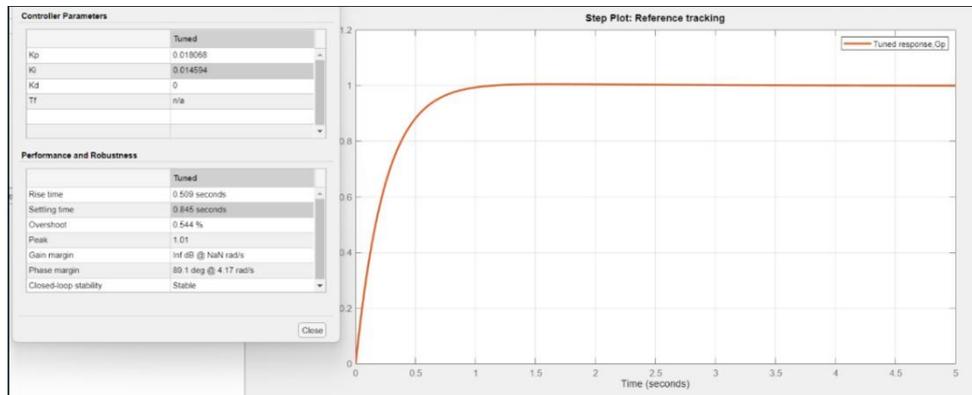
Continuous-time transfer function.

Nota: La estimación matemática ayuda a la manipulación de la planta para el desarrollo de controladores

$$tf = \frac{230.2}{s + 0.7484}$$

Según la (Figura 83), el comportamiento de la planta es modelado con la ecuación mostrada, con un 76.11% de precisión. A partir de este punto se procede a realizar el diseño del controlador PID por parte del Software de Matlab con la aplicación de PID Tuning (Figura 83). Con la ayuda de un controlador PID provee un grado suficiente de estabilidad para la variación de cambio de los píxeles. La simulación es realizada usando la aplicación de Matlab (Figura 84) y para la realización de la simulación es necesario usar una entrada del tipo step. (Aruna & S, 2020).

Figura 84

Diseño de controlador

Nota: El controlador PI resulta ser de buen comportamiento por el tiempo de muestreo

Se obtuvo un controlador que permita movimientos suaves. De acuerdo al diseño se observa que los valores son muy pequeños, destacando que no existe un valor de constante derivativa, por lo tanto, el controlador constaría de solo dos partes: Proporcional e Integral, lo que convierte a un controlador PI con las siguientes constantes:

$$k_p = 0.018068$$

$$k_i = 0.014594$$

$$k_d = 0$$

Una vez obtenido el controlador, se procede a realizar una limpieza de datos mediante un filtro de Kalman.

Filtro de Kalman. El filtro de Kalman es un grupo de ecuaciones matemáticas que proveen una solución computacional (recursiva) eficiente (Aguilar & Angulo, Real-Time Model-Based Video Stabilization for Microaerial Vehicles, 2016). El filtro es muy

poderoso en varios aspectos; soporta estimaciones del pasado, presente, incluso de estados del futuro. (Welch, An Introduction to the Kalman Filter, 1997).

Este filtro predice un proceso usando una forma de realimentación y control: el filtro estima el estado del proceso en un tiempo y después obtiene la realimentación en forma de mediciones ruidosas (Grijalva & Aguilar, Landmark-Based Virtual Path Estimation for Assisted UAV FPV Tele-Operation with Augmented Reality, 2019). Las ecuaciones del filtro de Kalman caen en dos grupos: i) ecuaciones de actualización de tiempo y ii) ecuaciones de actualización de mediciones. (Welch & Bishop, 2019)

Las primeras, tienen como objetivo la proyección en futuro del estado actual y estimar los errores y covarianza para obtener estimaciones en el tiempo (Aguilar, Merizalde, Calderón, & Carrera, 2020). Mientras que las ecuaciones de actualización de medida pueden ser vistas como ecuaciones correctoras. De hecho, el algoritmo final de estimación reúne los algoritmos de predicción-corrección para la resolución de problemas numéricos (Cárdenas, Arias, & Bravo, 2008).

Ahora teniendo estos criterios, se procede a limpiar la señal mediante la siguiente ecuación:

$$\hat{x}_{n,n} = \hat{x}_{n,n-1} + \frac{1}{n} (z_n - \hat{x}_{n,n} - 1)$$

Figura 85

Estructura del filtro de Kalman

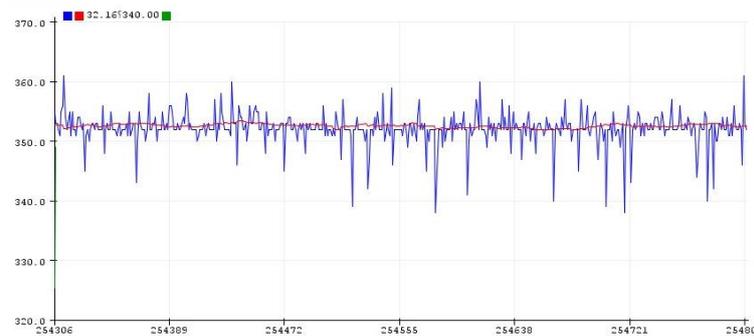
$$\boxed{\text{Estimación del estado actual}} = \boxed{\text{Predicción del estado actual}} + \boxed{\text{Factor}} \times \left(\boxed{\text{Medición}} - \boxed{\text{Predicción del estado actual}} \right)$$

Nota: El filtro de Kalman ayuda a predecir un estado futuro, lo cual es una ventaja para lograr un control más suave

El papel de este filtro en la entrada del controlador es muy importante ya que evita que los valores “basura” que se quedan en el buffer de la transmisión interfieran en la escritura de los ángulos en los servomotores. En la (Figura 85) se muestra el comportamiento del filtrado en la escritura de los servomotores.

Figura 86

Comportamiento de Filtro de Kalman



Nota: La visualización del filtro de Kalman ayuda a entender el comportamiento del mismo

Como se observa en la (Figura 86), al tener varios picos, gracias al Filtro de Kalman se obtiene una señal más limpia que permite un control más estable de cada uno de los servomotores.

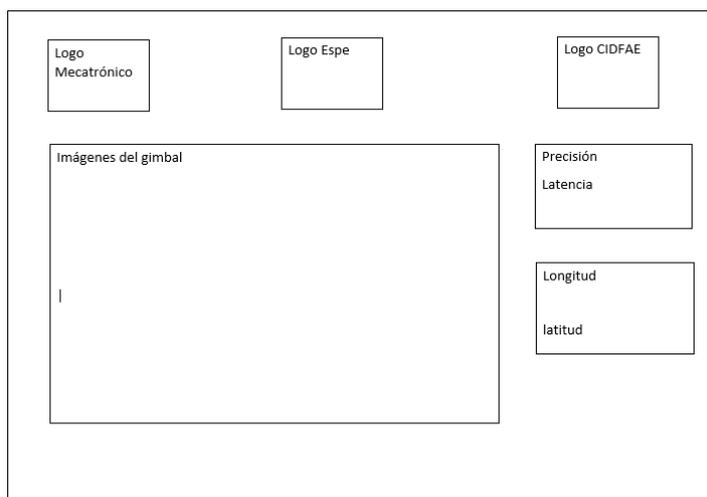
Diseño del HMI de estación en tierra

Como se mencionó anteriormente el HMI se implementará en un pc portátil en una página web. La comunicación se realizó mediante la implementación de un cliente como se describió en comunicación mediante socket.

El HMI se distribuyó de la siguiente manera: en la parte superior colocaron los logos junto con el nombre de la universidad, en la parte izquierda se encuentra un recuadro que transmitirá el video del celular, en la parte derecha están los datos de la precisión, así como los datos de longitud y latitud. Esto se evidencia en la (Figura 87)

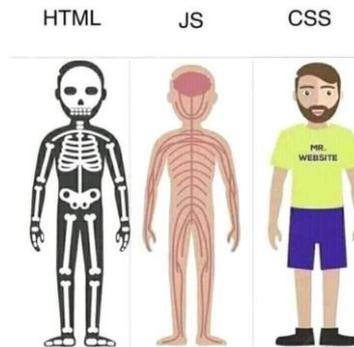
Figura 87

Diseño HMI



Nota: El diseño del HMI debe ser amigable con el usuario en tierra

La página web consta de tres partes Figura 88: La primera parte es el HTML, una estructura que toda página web debe tener, la segunda parte es JavaScript que ayuda al funcionamiento de la página, es decir, contiene botones acciones y demás, la tercera parte es CSS que corresponde al estilo o formato de la página y aquí se configuran parámetros como color, espacios, etc.

Figura 88*Diseño HTML*

Nota: La estructura simplificada de páginas web permiten el entendimiento del mismo

Para realizar la estructura HTML se descarga una plantilla para modificar el funcionamiento y la parte gráfica, la estructura se muestra en la (Figura 89) parte izquierda, mientras que en la parte derecha se observa el estilo de cada botón y color de fondo.

Figura 89*Programación página web*

```

<div class="content">
  <div class="row">
    <div class="column-left">
      <canvas id="viewport" class="viewport">
        Your browser does not support the HTML5 canvas tag.
      </canvas>
    </div>
    <div class="column-right">
      <div class="log">
        <div id="log-title">Detección de Militar</div>
        <div id="log-description">No detectado</div>
        <div id="log-percentage"></div>
      </div>
      <div class="log">
        <div id="log-title">GPS del Detector</div>
        <div id="gps-log-lat">Latitud: No recibido</div>
        <div id="gps-log-lon">Longitud: No recibido</div>
      </div>
    </div>
  </div>
</div>
</style>
</head>
<title>Militar Detector Web</title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
<meta name="viewport" content="width=device-width, initial-scale=1.0">
</head>
<body>
  <div class="topnav">
    <div class="imagen" >
      
    </div>
    <div class="imagen" >
      
    </div>
    <div class="imagen" >
      Prueba</b> | <b>Tiempo de Funcionamiento [min]</b> |
|---------------|---------------------------------------|
| 1             | 212                                   |
| 2             | 230                                   |
| 3             | 245                                   |
| 4             | 210                                   |
| Promedio      | 897                                   |

*Nota:* Gracias a las diferentes pruebas realizadas se puede obtener un valor medio de funcionamiento del prototipo.

De acuerdo al análisis, se tiene un tiempo de funcionamiento 3 horas con 44 min. El valor de dato encontrado depende de la capacidad del dron, ya que el objetivo es que el gimbal tenga igual o mayor tiempo de autonomía.

***Pruebas de comunicación***

Para poder determinar el alcance máximo de transmisión se realizaron pruebas en un espacio abierto sin interferencia de señal. La primera prueba ejecutó solo con el alcance de la red de la Raspberry (Tabla 19). Se comenzó con una distancia de 1 m hasta 5 m en cada iteración. Finalmente, los resultados muestran que el alcance de la Raspberry es de 25 m.

**Tabla 19**

*Distancia de transmisión sin repetidor*

| <b>Distancia(m)</b> | <b>Transmisión</b> |
|---------------------|--------------------|
| 5                   | si                 |
| 10                  | si                 |
| 15                  | si                 |
| 20                  | si                 |
| 25                  | si                 |
| 30                  | no                 |

*Nota:* La implementación de un repetidor puede mejorar la transmisión de datos a larga distancia

Se realizó un segundo experimento colocando un repetidor para ampliar el alcance de la red, se repitió el experimento al igual que en el apartado anterior y los resultados se presentan en la (Tabla 20). Se determinó que el alcance era grande motivo por el cual se tomaron datos cada 20 m.

**Tabla 20***Distancia de transmisión con repetidor*

| Distancia (m) | Transmisión |
|---------------|-------------|
| 20            | si          |
| 40            | si          |
| 60            | si          |
| 80            | si          |
| 100           | si          |
| 120           | si          |
| 140           | no          |

*Nota:* La implementación de un repetidor ayudó a la mejora con respecto a la distancia sin obstáculo de recepción de datos

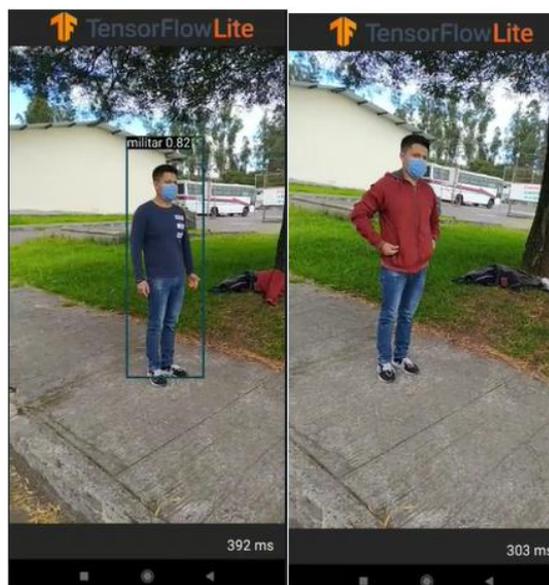
Al tener un repetidor que maximiza el rango se pudo alcanzar una distancia de 120 m en lugares abiertos, cabe recalcar que la red no requiere de internet por lo cual esto se puede implementar en misiones de lugares abiertos.

### ***Prueba de Red Neuronal***

Para las pruebas de detección de objetivos, en esta investigación, con una persona con uniforme militar, se probó en diferentes imágenes y escenarios, ubicados a varias distancias, así como con imágenes que no se encuentran en el set de datos de entrenamiento. Durante la prueba de la red neuronal se encontró que detectaba falsos positivos, para esto, la solución fue aumentar el número de imágenes de entrenamiento.

**Figura 96**

*Reconocimiento-pruebas de funcionamiento*



*Nota:* La detección de falsos positivos se debe mejorar mediante el entrenamiento con más imágenes

De acuerdo a la (Figura 96), se tiene falsos positivos, por tanto, se hizo un reentrenamiento con más imágenes para mejorar el algoritmo hasta que ya no detecte personas con ropa civil.

Para poder realizar una prueba experimental del prototipo y tener en consideración el alcance que el mismo va a poseer, se procedió a correr un diseño experimental de Taguchi, el cual se basa en arreglo ortogonales enfatizando el diseño robusto en la búsqueda de condiciones para lograr un comportamiento óptimo (Barrera , 2002).

En el diseño de Taguchi se busca las combinaciones óptimas de las variables de entrada dando lugar al diseño experimental, el cual consiste en un conjunto de pruebas durante la cual se realizan cambios sistemáticos y controlados a las variables de entrada para medir el efecto sobre la variable de salida (Yacuzzi & Martín, 2004). Dentro del

diseño se tienen variables independientes cuya variación de valores, afecta a la variable dependiente. Las variables independientes son llamadas factores. En la (Tabla 21) se muestra el uso de 3 factores y las variaciones de cada factor (niveles). En este diseño se consideraron 3 factores de la siguiente manera.

- Distancia: variaciones mediante el desplazamiento del objetivo
- Luminosidad: se realizó acorde a las horas del día, en la mañana a medio día y en al atardecer
- Altura: variaciones con el dron

**Tabla 21**

*Diseño experimental de Taguchi*

|   | FACTORES    | NIVELES |   |   |
|---|-------------|---------|---|---|
|   |             | 1       | 2 | 3 |
| A | Distancia   |         |   |   |
| B | Luminosidad |         |   |   |
| C | Altura      |         |   |   |

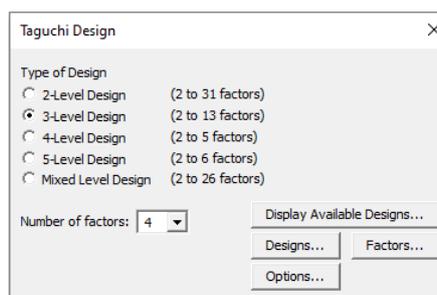
*Nota:* Los datos de experimentación ayudan a la determinación de condiciones de funcionamiento óptimas

Las salidas las consideramos dos, el primero es el nivel de certeza el cual nos indica cual es el porcentaje que la aplicación considera que es un militar, dicho dato nos envía de manera automática la aplicación. El segundo factor que se considero es la sensibilidad de la red, para lo cual se considera los falsos positivos los falsos negativos y se realizó el análisis correspondiente. En la (Figura 97) observamos que se utilizó el

experimento completo en el cual se corre 27 veces obteniendo en total 27 corridas al realizar 3 repeticiones, cada repetición en escenarios diferentes.

### Figura 97

#### *Diseño experimental de Taguchi minitab*

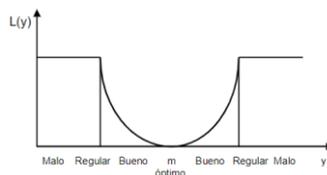


*Nota:* La determinación de los niveles de Taguchi pueden tener repercusión en el modelo

Taguchi nos indica que las pérdidas son función de la variación entre las características de calidad del producto respecto a su valor nominal. Taguchi desarrolla una función de pérdida de calidad la cual se representa con una función cuadrática con vértice en el valor nominal  $m$ , cuando los valores se alejan  $m$  la calidad decae y las pérdidas aumentan (Hernandez & Garcia , 2014). Esto se evidencia en la (Figura 98)

## Figura 98

### Criterio de calidad según Taguchi



*Nota:* La realización de la calidad tiene como finalidad la mejora de un sistema creado. Tomado de *la metodología de Taguchi en el control estadístico de la calidad* (p.129), por Hernandez, Alicia & Garcia , Liliana, 2014

El objetivo se centra en obtener la pérdida promedio por unidad como indicador de desempeño. Para analizar la pérdida promedio por unidad se debe distinguir tres casos según las características de la calidad (Hernandez & Garcia , 2014)

- Debe alcanzar un valor nominal (Caso: Nominal es el mejor)
- Debe alcanzar el menor valor posible (Caso: Menor es el mejor)
- Debe alcanzar el mayor valor posible (Caso: Mayor es el mejor)

La pérdida por unidad por cada caso lo podemos evidenciar en la (Figura 99)

## Figura 99

### Pérdida por unidad diseño de Taguchi

$$\bar{L} = K \times [\sigma^2 + (\mu - m)^2] = K \times \sigma^2 + K(\mu - m)^2 \quad (\text{Nominal es el mejor})$$

$$\bar{L} = K \times (\sigma^2 + \mu^2) \quad (\text{Menor es el mejor})$$

$$\bar{L} = \left(\frac{K}{\mu^2}\right) \times \left[1 + \frac{(3 \times \sigma^2)}{\mu^2}\right] \quad (\text{Mayor es el mejor})$$

*Nota:* Tomado de *Diseño y desarrollo de productos* (p.55), por Ulrich, K, 2013

Para el sistema se seleccionó dos casos para el análisis del valor de certeza se seleccionó más grande es mejor puesto que nuestro objetivo con el primer análisis consiste en obtener el valor de certeza más alto, mientras que para la sensibilidad se seleccionó de igual manera más alto es mejor puesto que mientras más sensible sea nuestra red significa que el número de falsos positivo o falsos negativos es mínimo.

A continuación, se detalla la ejecución de las variaciones de cada factor. Para poder determinar los rangos de distancia para el experimento, se utilizó el dron fijo en una terraza, se comenzó a realizar mediciones desde 1 m inicialmente incrementando de forma gradual para medir el nivel de certeza del algoritmo (Tabla 22).

**Tabla 22**

*Valores de certeza variando distancia*

| <b>Distancia (m)</b> | <b>Valor de certeza (%)</b> |
|----------------------|-----------------------------|
| 2                    | 98-100                      |
| 4                    | 95-98                       |
| 6                    | 94-97                       |
| 8                    | 85-90                       |
| 10                   | 71-85                       |
| 12                   | 62-69                       |
| 14                   | 50-60                       |
| 16                   | -                           |

*Nota:* Vemos que concorde a la distancia, el valor de certeza se reduce considerablemente.

De acuerdo a los resultados, el objetivo se aleja y el valor de certeza disminuye, esto se debe al uso de una red que convierte la imagen a una resolución de 320x320; por tanto, cuando el objetivo es distante ya no se diferencian claramente los pixeles característicos del militar.

Se logró aumentar el rango de distancia al realizar zoom, sin embargo, el seguimiento se limita, pues los movimientos son más pronunciados. Es decir, mientras el objetivo está cerca la certeza es alta, se ha determinado el primer nivel a 5 m. El nivel máximo (inicio de problemas de detección) es a 15 m, este fue considerado como el nivel 3. Finalmente, el nivel dos será el intermedio es decir 10 m. Las pruebas de estos experimentos se evidencian en la (Figura 100).

### Figura 100

#### *Pruebas de distancia*



*Nota:* Obtenido de los investigadores

Las pruebas de luminosidad se realizaron a diferentes horas del día. Se determinaron horas donde la variación de luz sea mínima, como el los horarios de medio día y atardecer. Los lúmenes se midieron mediante una aplicación de Android. Los datos obtenidos de luminosidad se presentan en la (Tabla 23).

**Tabla 23***Valor de certeza variando la luminosidad*

| Luminosidad (lux) | Valor de certeza (%) |
|-------------------|----------------------|
| 25000             | 98-99                |
| 15000             | 95-98                |
| 10000             | 95-98                |
| 5000              | 92-96                |
| 1000              | 85-90                |
| 500               | 80-85                |
| 100               | 70-80                |

*Nota:* El valor de la luminosidad influye en el valor de certeza del sistema de visión

Realizando el análisis de la luminosidad se concluyó que cuando a gran cantidad de lúmenes, el algoritmo tiene un porcentaje de certeza alto, esto sucede de igual manera con la luminosidad media. Por otro lado, a luminosidad baja, la aplicación ya no podía distinguir correctamente los colores del militar, dando así un valor de certeza muy bajo o en ocasiones sin poder detectarlo.

Por consiguiente, se estableció una luminosidad alta a medio día, luminosidad media a las 6 de la tarde y luminosidad baja a las 6:45. En el transcurso entre las 12 y las 6 la variación es mínima, por lo que a partir de esa hora comenzaron a bajar los lúmenes de manera pronunciada. Las pruebas en base a esta variable se presentan en la (Figura101)

## Figura 101

### Pruebas de luminosidad



*Nota:* Obtenido de los investigadores

Para las pruebas de variación de altura, el objetivo se colocó en un punto fijo mientras que el dron se desplazó en altura. Se inició con la altura de 1 m y se aumentó 1 m en cada prueba, las mediciones realizadas se presentan en la (Tabla 24).

**Tabla 24**

*Valor de certeza variando la altura*

| Altura (m) | Valor de certeza (%) |
|------------|----------------------|
| 1          | 95-99                |
| 2          | 93-98                |
| 3          | 90-96                |
| 4          | 85-91                |
| 5          | 70-85                |

*Nota:* Conforme la altura aumenta, el valor de certeza del modelo se ve disminuido

Después de analizar los resultados se determinó que a una altura entre 1-5 m la aplicación mantiene un alto nivel de certeza. Sin embargo, al sobrepasar los valores mencionados, la certeza disminuye. Esto se debe al zoom que dificulta mínimamente el seguimiento. La disminución del porcentaje de certeza se evidencia en la (Figura 102). En base a lo concluido, se estableció el primer nivel en 1 m, el segundo nivel en 3 m y el tercer nivel en 5 m.

### Figura 102

#### *Pruebas de altura*



*Nota:* Obtenido de los investigadores

Se realizó otra variación en el experimento que consistió en aumentar más personas en la escena para que la aplicación diferencie y siga solo al objetivo. Se inició con solo una persona con uniforme de militar y posteriormente se aumentaron civiles. Los resultados se presentan en la (Tabla 25).

**Tabla 25:**

*Valor de certeza variando la cantidad de personas*

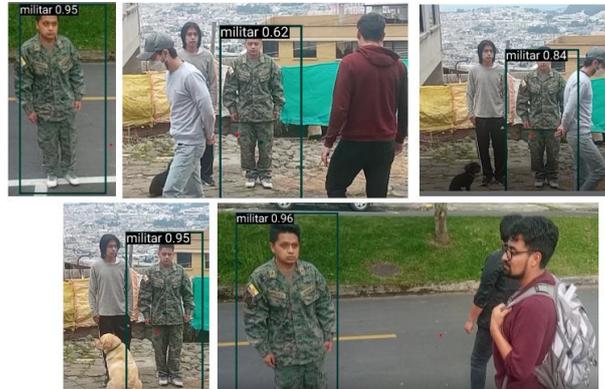
| <b>Personas (cantidad)</b> | <b>Valor de certeza (%)</b> |
|----------------------------|-----------------------------|
| 0                          | 95-99                       |
| 1                          | 90-95                       |
| 2                          | 80-90                       |
| 3                          | 60-80                       |

*Nota:* Obtenido de los investigadores

Los valores obtenidos muestran que al mantener en escena solo al militar se el valor de certeza es alto, sin embargo, al intervenir más personas en la escena se generan dos circunstancias: i) Cuando las personas se colocan cerca o tapando parcialmente al militar el valor de certeza disminuye mientras que, ii) al momento que las personas se encuentran distantes del militar el valor de certeza incrementa. Las pruebas de los ensayos se exponen en la (Figura 103). Aunque el factor no intervenga en el diseño experimental se consideró importante realizar pruebas del funcionamiento en estas circunstancias para poder determinar la existencia de falsos positivos o falsos negativos.

**Figura 103**

Pruebas realizadas variando cantidad de personas



*Nota:* La realización de pruebas con varias personas ayudan a la veracidad del sistema.

Con los rangos obtenidos se determinaron los tres a realizar los experimentos de la Tabla 26. Después del análisis, se obtuvo que al incrementar varias personas los resultados no variaban significativamente, por lo que se decidió colocar los tres factores más demostrativos.

**Tabla 26**

*Diseño de Taguchi 3 factores con 3 niveles*

| FACTORES |               | NIVELES |       |      |
|----------|---------------|---------|-------|------|
|          |               | 1       | 2     | 3    |
| A        | Distancia (m) | 4       | 8     | 12   |
| B        | Luminosidad   | Alta    | Media | Baja |
| C        | Altura (m)    | 1       | 2     | 3    |

*Nota:* Se describe los factores y niveles que se utilizaran para el diseño en minitab

El diseño experimental (Figura 104) se ejecutó en el programa minitab. El modelo completo consta de 27 experimentos, se analizó los frames durante 5 segundos de video para obtener los datos realizando de esta manera el promedio para el posterior análisis.

**Figura 104**

*Diseño experimental en minitab*

| ↓  | C1        | C2     | C3-T        | ↓  | C1        | C2     | C3-T        |
|----|-----------|--------|-------------|----|-----------|--------|-------------|
|    | Distancia | Altura | Luminosidad |    | Distancia | Altura | Luminosidad |
| 1  | 5         | 1      | Alta        | 13 | 10        | 3      | Baja        |
| 2  | 5         | 1      | Alta        | 14 | 10        | 3      | Baja        |
| 3  | 5         | 1      | Alta        | 15 | 10        | 3      | Baja        |
| 4  | 5         | 3      | Media       | 16 | 10        | 5      | Alta        |
| 5  | 5         | 3      | Media       | 17 | 10        | 5      | Alta        |
| 6  | 5         | 3      | Media       | 18 | 10        | 5      | Alta        |
| 7  | 5         | 5      | Baja        | 19 | 15        | 1      | Baja        |
| 8  | 5         | 5      | Baja        | 20 | 15        | 1      | Baja        |
| 9  | 5         | 5      | Baja        | 21 | 15        | 1      | Baja        |
| 10 | 10        | 1      | Media       | 22 | 15        | 3      | Alta        |
| 11 | 10        | 1      | Media       | 23 | 15        | 3      | Alta        |
| 12 | 10        | 1      | Media       | 24 | 15        | 3      | Alta        |
| 13 | 10        | 3      | Baja        | 25 | 15        | 5      | Media       |
| 14 | 10        | 3      | Baja        | 26 | 15        | 5      | Media       |
|    |           |        |             | 27 | 15        | 5      | Media       |

*Nota:* Se indica la manera en la que se debe de realizar las combinaciones

El diseño experimental se ejecutó en una zona abierta, se realizaron diferentes marcas en los niveles para facilitar su desarrollo. Después de correr todos los experimentos se generaron los valores, con la fila Certeza y sensibilidad pasamos los datos al programa minitab en el cual se realizó el análisis estadístico.

Para el análisis del comportamiento de la red neuronal se realiza mediante la matriz de confusión, esta permite comparar los valores predichos por el modelo de aprendizaje, con los valores reales. Esto se realiza mediante el siguiente modelo descrito en la Figura 105

**Figura 105**

*Modelo de matriz de confusión*

|                  |          | ACTUAL VALUES |          |
|------------------|----------|---------------|----------|
|                  |          | POSITIVE      | NEGATIVE |
| PREDICTED VALUES | POSITIVE | TP            | FP       |
|                  | NEGATIVE | FN            | TN       |

*Nota:* La matriz de confusión nos ayuda a determinar la veracidad del mismo modelo

De las cuales, se definen los siguientes valores de la matriz de confusión:

#### **Verdadero positivo (TP)**

- El valor predicho coincide con el valor real
- El valor real fue positivo y el modelo predijo un valor positivo

#### **Verdadero Negativo (TN)**

- El valor predicho coincide con el valor real
- El valor real fue negativo y el modelo predijo un valor negativo

#### **Falso positivo (FP): error tipo 1**

- El valor predicho fue predicho falsamente
- El valor real fue negativo pero el modelo predijo un valor positivo

#### **Falso negativo (FN): error de tipo 2**

- El valor predicho fue predicho falsamente

- El valor real fue positivo pero el modelo predijo un valor negativo

Sin embargo, estos valores funcionan siempre y cuando se pueda cuantificar los siguientes valores con su respectiva ecuación:

- **Recall:** De todas las clases positivas cuantas se predijo correctamente

$$Re = \frac{VP}{VN + FN}$$

- **Precisión:** De todas las positivas que se han predicho correctamente cuántas son realmente positivas

$$Pr = \frac{VP}{VP + FP}$$

- **Sensibilidad:** De todas las clases, cuantas se predijeron correctamente

$$Acc = \frac{VP + VN}{Total}$$

A continuación, se definen los ejemplos de cada uno de los valores en la matriz que se tomó en cuenta en cada prueba, como lo podemos observar en la (Tabla 27)

Tabla 27

*Ejemplo de los diferentes casos matriz de confusión*

| Verdadero positivo                                                                  | Falso positivo                                                                       |
|-------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------|
|    |    |
| Falso Negativo                                                                      | Verdadero negativo                                                                   |
|  |  |

*Nota:* Podemos observar que en las pruebas encontramos todos los casos, pero la detección se lo realizó al analizar fotograma mediante la ayuda de un programa de video

En análisis de los resultados se lo presentara en tres tablas en las cuales se mantendrán los parámetros de luminosidad podremos apreciar de una manera más clara las variaciones que se realizaron. En la (Tabla 28) se mantuvo una luminosidad alta y se realizó las variaciones y combinaciones de los parámetros de distancia y altura. En la (Tabla 29) se realizó el mismo diseño, pero se varió la luminosidad a media, mientras que en la (Tabla 30) la luminosidad se mantuvo en baja.

**Tabla 28***Diseño de Taguchi con Luminosidad Alta*

| <b>Distancia</b> |      | <b>5 (m)</b>   |                     | <b>10(m)</b>   |                     | <b>15(m)</b>   |                     |
|------------------|------|----------------|---------------------|----------------|---------------------|----------------|---------------------|
| <b>Salida</b>    |      | <b>Certeza</b> | <b>Sensibilidad</b> | <b>Certeza</b> | <b>Sensibilidad</b> | <b>Certeza</b> | <b>Sensibilidad</b> |
| Altura           | 1(m) | 96             | 0.85                | 92             | 0.82                | 87             | 0.81                |
|                  | 3(m) | 92             | 0.82                | 89             | 0.82                | 88             | 0.8                 |
|                  | 5(m) | 94             | 0.83                | 85             | 0.81                | 76             | 0.8                 |

*Nota:* Datos obtenidos con una luminosidad alta**Tabla 29***Diseño de Taguchi con Luminosidad Media*

| <b>Distancia</b> |      | <b>5 (m)</b>   |                     | <b>10(m)</b>   |                     | <b>15(m)</b>   |                     |
|------------------|------|----------------|---------------------|----------------|---------------------|----------------|---------------------|
| <b>Salida</b>    |      | <b>Certeza</b> | <b>Sensibilidad</b> | <b>Certeza</b> | <b>Sensibilidad</b> | <b>Certeza</b> | <b>Sensibilidad</b> |
| Altura           | 1(m) | 92             | 0.85                | 91             | 0.81                | 85             | 0.81                |
|                  | 3(m) | 93             | 0.83                | 90             | 0.8                 | 86             | 0.8                 |
|                  | 5(m) | 92             | 0.83                | 83             | 0.8                 | 84             | 0.79                |

*Nota:* Datos obtenidos con una luminosidad media

Tabla 30

*Diseño de Taguchi con Luminosidad Baja*

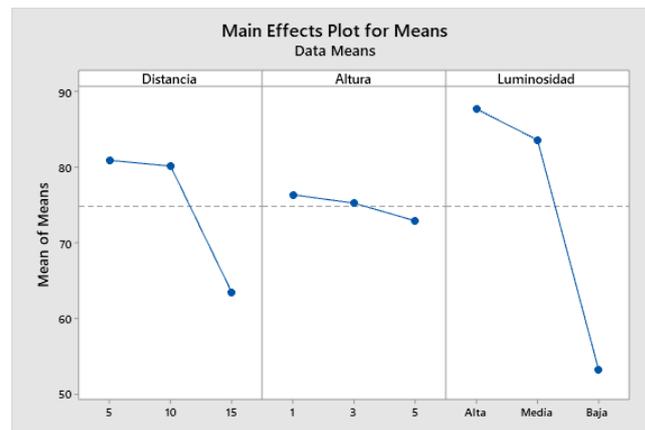
| Distancia |      | 5 (m)   |              | 10(m)   |              | 15(m)   |              |
|-----------|------|---------|--------------|---------|--------------|---------|--------------|
| Salida    |      | Certeza | Sensibilidad | Certeza | Sensibilidad | Certeza | Sensibilidad |
| Altura    | 1(m) | 74      | 0.77         | 60      | 0.7          | 52      | 0.7          |
|           | 3(m) | 72      | 0.7          | 53      | 0.68         | 49      | 0.67         |
|           | 5(m) | 61      | 0.7          | 52      | 0.65         | 50      | 0.69         |

*Nota:* Datos obtenidos con una luminosidad baja

### Análisis nivel de certeza

Figura 106

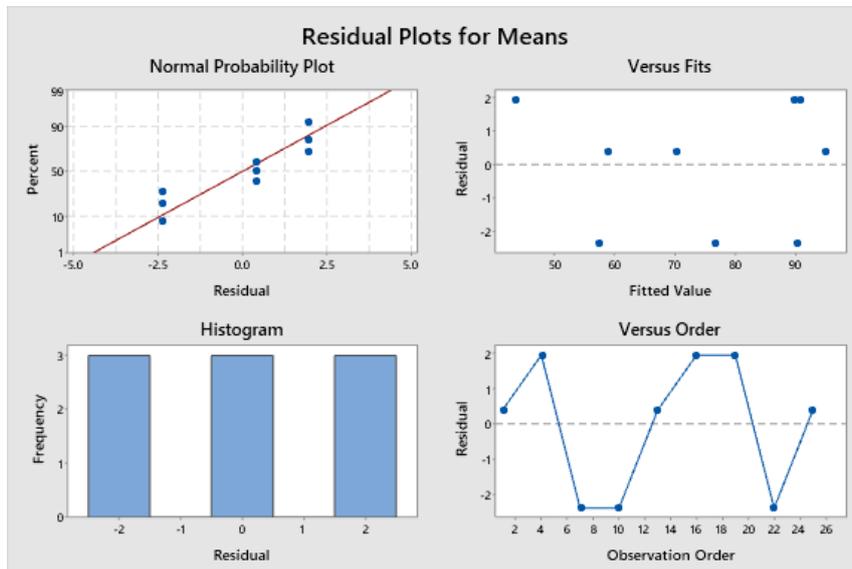
*Gráfico de efectos principales para medias*



*Nota:* Se puede analizar que la luminosidad es el factor que más afecta a la salida

**Figura 107**

*Gráficos de residuos para medias*



*Nota:* Graficas residuales que nos indican la dispersión de los datos

**Figura 108**

*Ejemplos de la realización del diseño experimental*



*Nota:* Ejemplos gráficos de algunas corridas del diseño experimental

Se realizó el análisis del nivel de certeza y como la variación de los factores afectan a nuestra salida, de acuerdo a la (Figura 108). El análisis de las medias indica el grado de afectación a la respuesta. Como primer factor se encuentra la luminosidad, que, al tener un nivel bajo, dificulta la distinción de los colores y los patrones característicos disminuyendo considerablemente el nivel de certeza.

El segundo factor es la distancia, el zoom permite aumentar este factor, pero a la vez, reduce la precisión en el seguimiento; es decir, tener una mayor distancia dificulta encontrar al objetivo. el tercer factor es la altura, que al igual que la distancia, tiende a bajar el nivel de certeza al aplicar el zoom.

### Figura 109

*Respuesta para medias*

#### Response Table for Signal to Noise Ratios

Larger is better

| Level | Distancia | Altura | Luminosidad |
|-------|-----------|--------|-------------|
| 1     | 37.86     | 37.16  | 38.79       |
| 2     | 37.86     | 37.33  | 38.38       |
| 3     | 35.79     | 37.02  | 34.34       |
| Delta | 2.07      | 0.31   | 4.45        |
| Rank  | 2         | 3      | 1           |

*Nota:* Se muestra el orden de los factores que más afectan a la respuesta

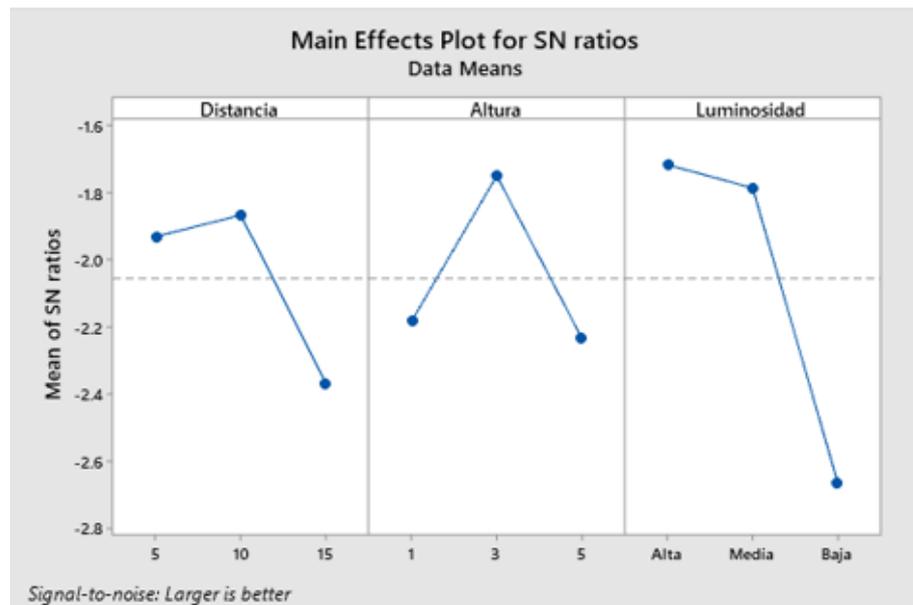
Se determinó que el factor que influye en su mayoría a la aplicación es la luminosidad baja, ya que los colores y los rasgos característicos no se distinguen bien en estas condiciones (Figura 109). Otro de los factores que perturba es la distancia, pues a mayor distancia el algoritmo tiende a fallar y se dificulta el seguimiento, por consiguiente, existe una disminución en la certeza.

Dentro del análisis se concluyó que los parámetros óptimos para un mejor desempeño será a una corta distancia, con una altura de 1 metro y luminosidad media. Sin embargo, en las condiciones que generan un bajo nivel de certeza la base de datos puede ser entrenada a mayor profundidad.

### Análisis de sensibilidad

**Figura 110**

*Análisis de Sensibilidad realizado en minitab*



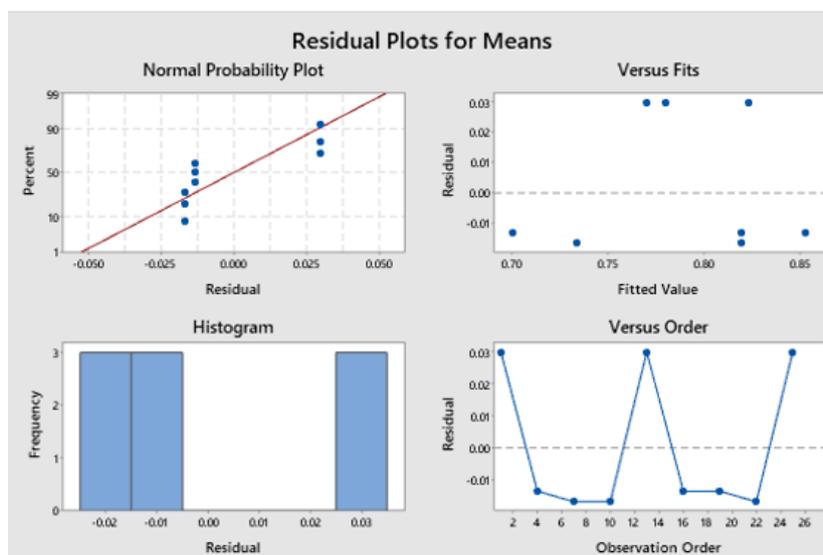
*Nota:* Como podemos analizar el factor que más influye es la luminosidad

Una vez obtenidos la (Figura110) del comportamiento podemos concluir que, para lograr una detección constante, con valores por encima del promedio, las condiciones de luminosidad deben permanecer en un nivel alto y medio, esto debido que, si la luminosidad se encuentra en un nivel bajo, la red va a empezar a tener bajos niveles de certeza, esto debido a que, la arquitectura del modelo reescala las imágenes de entrenamiento, validación y pruebas a una resolución máxima de 320x320 píxeles,

lo que realiza una pérdida de características en su entrenamiento, pero a su vez impide la realización de una inferencia que mantenga todas las características de las imágenes.

**Figura 111**

*Gráficas residuales realizadas en minitab*



*Nota:* Análisis del tipo de dispersión obtenidos en minitab

A través de la (Figura 111) podemos concluir que los datos recolectados muestran una dispersión normal, lo que nos indica la aleatoriedad de los mismos y realizar un análisis e interpretación de los datos.

### ***Pruebas de seguimiento***

Para el seguimiento del objetivo, se implementó un “tracker” que permite el movimiento de los servomotores habilitando el enfoque y encuadre del mismo dentro de la imagen.

Este “tracker” se realizó de la forma mencionada ya que no existen de momento, librerías que permitan la transferencia de funciones desde OpenCv hasta Android, pues Android no brinda soporte a estas librerías. De ser así se puede usar librerías de

medianflow para el uso de algoritmos de flujo óptico y poder implementarlo en el seguimiento del objetivo.

La solución al problema es que, para seguir al objetivo se debe definir el cuadro (bounding box). Para saber el centro de la imagen se tiene el punto medio de la pantalla, lo que permite aplicar el controlador implementado anteriormente.

### Figura 112

*Cálculo de error en imagen*



*Nota:* El punto medio de la imagen nos ayuda para realizar el control

En base a la (Figura 112) se tiene en el centro de la pantalla el punto de color rosa, mientras el punto de color verde es el punto medio del cuadro detector. Mediante la obtención de coordenadas, se puede aplicar el controlador implementado en la tarjeta controladora.

**Figura 113***Seguimiento de objetivo militar*

*Nota:* las imágenes representan el seguimiento que se hizo al objetivo

Se presenta en la (Figura 113) que el seguimiento continúa conforme el objetivo se acerque a la cámara, además, ambos puntos que marcan el centroide de la pantalla como del bounding box, tratan de juntarse, y lograr que la imagen permanezca centrada y que se pueda enfocar de la mejor manera posible. Este control permite un movimiento limitado de la imagen, sin embargo, admite un rango de visión apto para realizar el seguimiento en varios ambientes.

## Capítulo V

### Conclusiones y recomendaciones

#### *Conclusiones*

- Para el diseño de las piezas que conformarán el prototipo, es necesario el uso de materiales resistentes y de bajo peso, con el objetivo de evitar afectar el torque de los servomotores, piezas fundamentales del movimiento.
- Se concluye que el uso de la manufactura aditiva con PLA es una de las mejores opciones para probar varios diseños a fin de que éste sea personalizado, sin embargo, el tiempo de manufactura de las piezas es un factor a considerar si se requiere realizar varios modelos y comprobar su funcionamiento y/o rendimiento.
- Para mantener un modelo físico estable y hecho con manufactura aditiva (impresión 3D) con PLA, es preciso mantener un relleno de no menos del 80% para no sufrir un torque innecesario y que las fibras se disuelvan en el funcionamiento.
- Se concluye que el diseño tipo “rompecabezas” disminuye considerablemente el tiempo de ensamblaje del prototipo, además de reducir el uso de piezas de fijación como tornillos y pernos.
- El diseño generativo ayuda a reducir el peso de las piezas mecánicas manteniendo el comportamiento estático de cada una, sin embargo, al ser piezas que no superan los 20 cm de largo y 15 cm de ancho, resulta complicado para el software encontrar la manera de reducir aún más el peso de cada pieza.
- Se concluye que la comunicación mediante socket permite una velocidad de transmisión entre el teléfono móvil y la raspberry de 70-90 ms, es decir, un rango entre 11-14 fps. Al tener una comunicación mediante cable se obtiene una velocidad estándar entre 25-30 fps, por lo que, al obtener alrededor de la mitad

del estándar mediante comunicación inalámbrica se considera óptimo para detección en tiempo real.

- El rango de alcance de la comunicación entre el gimbal y la estación en tierra mediante web server es de 25m, sin embargo, la aplicación de un repetidor de señal cerca de la estación en tierra, aumenta el rango hasta 120m en zonas abiertas. Este valor se puede ampliar más al incrementar los repetidores de señal y no es necesario una conexión a internet.
- Se concluye que el algoritmo de visión artificial detecta con una alta certeza a militares que se encuentran caminando, más aún tiene inconvenientes con militares sentados o acostados. Esto se debe a la base de datos no posee suficientes imágenes con esas características.
- El factor que influye principalmente en el algoritmo de visión artificial es la luminosidad debido a los colores y rasgos característicos poco distinguibles. Sin embargo, lugares con más luminosidad presentan el nivel de certeza elevado.
- Se puede concluir que mediante el zoom se amplía el rango de distancia de detección, pero al extender con el zoom completo, se complica el seguimiento. El incremento máximo y óptimo corresponde a un rango de 12 m.
- El sensor óptico entrega una señal un poco ruidosa, permitiendo que el controlador presente pequeños saltos. Estas variaciones fueron corregidas con un filtro de Kalman sin sacrificar el tiempo de procesamiento.

### **Recomendaciones**

- Se recomienda la exploración de ensamblaje mediante piezas realizadas en 2D para reducir el tiempo de elaboración de cada una, así como la reducción de elementos de fijación como piezas angulares, tornillos, pernos, etc.
- El uso indirecto de los servomotores es una alternativa mejorable, es decir, realizar un sistema de transmisión de movimiento mediante poleas o engranajes, esto con el fin de reducir el “cabeceo” del movimiento.
- Se recomienda realizar el diseño generativo con un software que ejecute el análisis/cálculo en la nube (como por ejemplo Fusion360), esto con el fin de reducir el uso de recursos de la computadora local, y acelerar el proceso de cálculo.
- Para mejorar la velocidad de transmisión, se recomienda ocupar la aplicación en teléfonos con procesadores más potentes como el pixel 4 de Google, que es altamente recomendado en la mayoría de ejemplos para el desarrollo de visión artificial.
- Se recomienda realizar pruebas de alcance con más repetidores para poder ampliar el rango de transmisión, de igual manera se recomienda no utilizarlo en zonas con interferencia como árboles o edificios para evitar la disminución del rango.
- Se recomienda ampliar la base de datos de entrenamiento con militares en distintas posiciones y diferentes escenarios, esto ayudara a que la aplicación sea mucho más robusta.
- La incorporación de una cámara de visión nocturna es una alternativa para la detección de militares en ambientes oscuros, cabe recalcar que para la visión

nocturna se necesitan una base de datos con ese tipo de características para el entrenamiento.

- Se recomienda el uso de dos niveles de zoom, ya que al aumentar los niveles se generan cambios bruscos al controlador y en ocasiones se pierde el objetivo.
- Por último, se recomienda la implementación de filtros de Kalman para evitar señales de ruido y no confundir al controlador.

## Bibliografía

- Acevedo, J. (Julio de 2013). *Principios básicos del funcionamiento de la radiofrecuencia en el tratamiento del dolor crónico*.
- Aguilar, W. G., & Angulo, C. (2013). Estabilización robusta de vídeo basada en diferencia de nivel de gris. *VIII Congreso de Ciencia y Tecnología ESPE*.
- Aguilar, W. G., & Angulo, C. (2014). Real-time video stabilization without phantom movements for micro aerial vehicles. *EURASIP Journal on Image and Video Processing*, 1, 1-13.
- Aguilar, W. G., & Angulo, C. (2016). Real-Time Model-Based Video Stabilization for Microaerial Vehicles. *Neural Processing Letters*, 43(2), 459-477.
- Aguilar, W. G., & Morales, S. (2016). 3D Environment Mapping Using the Kinect V2 and Path Planning Based on RRT Algorithms. *Electronics*, 5(4), 70.
- Aguilar, W. G., Álvarez, L., Grijalva, S., & Rojas, I. (2019). Monocular Vision-Based Dynamic Moving Obstacles Detection and Avoidance. En *Lecture Notes in Computer Science* (págs. 386-398).
- Aguilar, W. G., Angulo, C., & Costa-Castello, R. (2017). Autonomous Navigation Control for Quadrotors in Trajectories Tracking. En *Lecture Notes in Computer Science* (págs. 287-297).
- Aguilar, W. G., Angulo, C., & Pardo, J. A. (2017). Motion intention optimization for multirotor robust video stabilization. *Chilean Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON)*. Pucón, Chile.
- Aguilar, W. G., Calderón, M., Merizalde, D., Amaguaña, F., & Tituaña, J. (2020). Visual and Inertial Data-Based Virtual Localization for Urban Combat. En *International Conference of Research Applied to Defense and Security* (págs. 65-74).
- Aguilar, W. G., Casaliglla, V. P., & Pólit, J. L. (2017). Obstacle Avoidance Based-Visual Navigation for Micro Aerial Vehicles. *Electronics*, 6(1), 10.

- Aguilar, W. G., Cobeña, B., Rodríguez, G., Salcedo, V. S., & Collaguazo, B. (2018). SVM and RGB-D Sensor Based Gesture Recognition for UAV Control. *International Conference on Augmented Reality, Virtual Reality and Computer Graphics* (págs. 713-719). Springer.
- Aguilar, W. G., Luna, M. A., Moya, J. F., Abad, V., Parra, H., & Ruiz, H. (2017). Pedestrian detection for UAVs using cascade classifiers with meanshift. *IEEE 11th international conference on semantic computing (ICSC)*.
- Aguilar, W. G., Luna, M. A., Moya, J. F., Abad, V., Ruiz, H., Parra, H., & Angulo, C. (2017). Pedestrian Detection for UAVs Using Cascade Classifiers and Saliency Maps. En *Lecture Notes in Computer Science* (págs. 563-574).
- Aguilar, W. G., Luna, M. A., Moya, J. F., Luna, M. P., Abad, V., Ruiz, H., & Parra, H. (2017). Real-time detection and simulation of abnormal crowd behavior. En *Lecture Notes in Computer Science* (págs. 420-428).
- Aguilar, W. G., Manosalvas, J. F., Guillén, J. A., & Collaguazo, B. (2018). Robust Motion Estimation Based on Multiple Monocular Camera for Indoor Autonomous Navigation of Micro Aerial Vehicle. *International Conference on Augmented Reality, Virtual Reality and Computer Graphics* (págs. 547-561). Springer.
- Aguilar, W. G., Merizalde, D., Calderón, M., & Carrera, A. (2020). Monte Carlo-Based Localization for Kidnapped Robot Problem. En *International Conference of Research Applied to Defense and Security* (págs. 101-109).
- Aguilar, W. G., Quisaguano, F. J., Alvarez, L. G., Pardo, J. A., & Zahira, P. (2018). Monocular depth perception on a micro-UAV using convolutional neuronal networks. En *Lecture Notes in Computer Science* (págs. 392-397).
- Aguilar, W. G., Quisaguano, F. J., Rodríguez, G. A., Alvarez, L. G., Limaico, A., & S, S. D. (2018). Convolutional neuronal networks based monocular object detection and depth perception for micro UAVs. En *Lecture Notes in Computer Science* (págs. 401-410).

- Aguilar, W. G., Rodríguez, G. A., Álvarez, L., Sandoval, S., Quisaguano, F., & Limaico, A. (2017). Visual SLAM with a RGB-D Camera on a Quadrotor UAV Using on-Board Processing. En *Lecture Notes in Computer Science* (págs. 596-606).
- Aguilar, W. G., Salcedo, V., Sandoval, D., & Cobeña, B. (2017). Developing of a Video-Based Model for UAV Autonomous Navigation. *Communications in Computer and Information Science*, 94-105.
- Akella, R. (16 de 3 de 2018). *IndustryWeek*. Obtenido de <https://www.industryweek.com/technology-and-iiot/article/22025329/what-generative-design-is-and-why-its-the-future-of-manufacturing>
- Altair. (2010). Obtenido de <https://web.altair.com/generative-design-report>
- Amaguaña, F., Collaguazo, B., Tituaña, J., & Aguilar, W. G. (2018). Simulation System Based on Augmented Reality for Optimization of Training Tactics on Military Operations. *International Conference on Augmented Reality, Virtual Reality and Computer Graphics* (págs. 394-403). Springer.
- Andrea, C. C., Byron, J. Q., Jorge, P. I., Inti, T. C., & Aguilar, W. G. (2018). Geolocation and Counting of People with Aerial Thermal Imaging for Rescue Purposes. *International Conference on Augmented Reality, Virtual Reality and Computer Graphics* (págs. 171-182). Springer.
- Araujo, B. (2018). *SISTEMA DE PROCESAMIENTO DIGITAL DE IMÁGENES PARA REALIZAR TRACKING AUTOMÁTICO DE OBJETOS DURANTE MISIONES DE VIGILANCIA Y RECONOCIMIENTO UTILIZANDO SISTEMAS ELECTROÓPTICOS DEL CIDFAE*. Universidad Técnica de Ambato.
- Ardila, H. J. (2011). *Implementación cliente servidor mediante sockets*.
- Aruna, R., & S, J. C. (2020). Modeling, system identification and design of fuzzy PID controller for discharge dynamics of metal hydride hydrogen storage bed. *International Journal of Hydrogen Energy*, 4703-4719.
- Ayora, M. J. (2012). *Clasificación de imágenes usando redes neuronales convolucionales en Python*.

- Barrera, J. (2002). LOS DISEÑOS DE TAGUCHI CONTRA LOS DISEÑOS CLÁSICOS DE EXPERIMENTOS. *Conciencia tecnológica*.
- Basantes, J., Godoy, L., Carvajal, T., Castro, R., Toulkeridis, T., Fuertes, W., . . . Addison, A. (2018). Capture and processing of geospatial data with laser scanner system for 3D modeling and virtual reality of Amazonian Caves. *IEEE Ecuador Technical Chapters Meeting (ETCM)*. Samborondón, Ecuador.
- Bhattacharjee, S., & Somashekhar, G. (2017). Artificial Intelligence to Impart Surveillance, Tracking, & Actuation on Suspicious Activities. *Hyderabad, India: IEE*, 1-5. doi:<https://doi.org/10.1109/IACC.2017.0016>
- Borase, Rakesh, P., Maghade, D. K., Sondkar, S. Y., & Pawar, S. N. (2021). A Review of PID Control, Tuning Methods and Applications. *International Journal of Dynamics and Control*, 18-27. doi:<https://doi.org/10.1007/s40435-020-00665-4>
- Borrayo, V. (2017). Sistema servidor-cliente seguro para la verificación de autenticidad de archivos por medio de algoritmos digest HASH reestadarizado73. *IEEE Xplore*.
- Cabras, P., Rosell, J., Pérez, A., Aguilar, W. G., & Rosell, A. (2011). Haptic-based navigation for the virtual bronchoscopy. *18th IFAC World Congress*. Milano, Italy.
- Caiza, O., Aguilar, W. G., Albán, P., & Fernández, Y. (2020). Kinect and Manipulator-Based Sample Collection System for Military Robot. En *Smart Innovation, Systems and Technologies* (págs. 75-87).
- Calderón, M., Aguilar, W. G., & Merizalde, D. (2020). Visual-Based Real-Time Detection Using Neural Networks and Micro-UAVs for Military Operations. En *Smart Innovation, Systems and Technologies* (págs. 55-64).
- Capraro, F. (3 de Julio de 2018). *Telemetría agrícola*. Obtenido de [https://inta.gob.ar/sites/default/files/telemetria\\_agricola\\_cai2018\\_ppt.pdf](https://inta.gob.ar/sites/default/files/telemetria_agricola_cai2018_ppt.pdf).
- Cárdenas, J. A., Arias, M. A., & Bravo, V. A. (2008). *ANÁLISIS Y APLICACIÓN DEL FILTRO DE KALMAN A UNA SEÑAL CON RUIDO ALEATORIO*.
- Carlletti, E. (2018). *Comunicacion BusI2C*.

- Castillo, C. (2020). *DESARROLLO DE UNA APLICACIÓN WEB PARA LA GESTIÓN DE LOS SERVICIOS TÉCNICOS EN EL CENTRO DE DIAGNÓSTICO AUTOMOTRIZ LA PERLA DEL SINÚ*. Cordoba.
- Cedeño, R. (2019). *I Feria de desarrollo tecnológico para la defensa*.
- Changi , G. (2011). Esquema de servicios para Televisión Digital Interactiva, basados en el protocolo REST-JSON. *Cadernos de Informática* .
- Christopher, B. (2006). *Pattern Recognition and Machine Learning*. Springer.
- CONTROLOP. (13 de Enero de 2021). *Homeland Security Company, Surveillance Systems»*. *Controp Precision Technologies (blog)*. Obtenido de <https://www.controp.com/about-us/>
- Deepika, P., Chauhan, S., & Narayan, N. (2017). Artificial Intelligence Techniques Used to Detect Object and Face in an Image: A Review. *3rd International Conference on Computational Intelligence and Networks (CINE)*, 6-9. doi:<https://doi.org/10.1109/CINE.2017.20>
- Delgado, J. (5 de Agosto de 2013). *CONTROL DE VELOCIDAD PARA MOTOR DC BRUSHLESS SIN SENSORES*. Obtenido de <https://repository.javeriana.edu.co/bitstream/handle/10554/13593/DelgadoJoseDavid2013.pdf>.
- Deng, J., & Dong, W. (2014). ImageNet: A Large-Scale Hierarchical Image Database.
- El Naqa, I., & Murphy, J. M. (2015). What Is Machine Learning? (I. E. Naqa, R. Li, & M. J. Murphy, Edits.) *Machine Learning in Radiation Oncology*, 3-11. doi:[https://doi.org/10.1007/978-3-319-18305-3\\_1](https://doi.org/10.1007/978-3-319-18305-3_1)
- Escobar, M. (2010). *CONTROL DE SENSORES ELECTRO-ÓPTICOS EN AVIONES NO TRIPULADOS Y TRATAMIENTO DE IMÁGENES UTILIZANDO MÁQUINAS DE VECTORES SOPORTE*. Universidad Carlos III de Madrid , Madrid.
- Falconí, L. G., Pérez, M., & Aguilar, W. G. (2019). Transfer learning in breast mammogram abnormalities classification with mobilenet and nasnet.

- International Conference on Systems, Signals and Image Processing (IWSSIP)*, (págs. 109-114).
- Falconi, L. G., Perez, M., Aguilar, W. G., & Conci, A. (2020). Transfer learning and fine tuning in breast mammogram abnormalities classification on CBIS-DDSM database. *Adv. Sci. Technol. Eng. Syst. J*, 154-165.
- Falconi, L. G., Perez, M., Aguilar, W. G., & Conci, A. (2020). Transfer learning and fine tuning in mammogram bi-rads classification. *IEEE 33rd International Symposium on Computer-Based Medical Systems (CBMS)*, (págs. 475-480).
- Fiedler, N. (2019). ImageTagger: An Open Source Online Platform for Collaborative Image Labeling.
- Galarza, J., Pérez, E., Serrano, E., Tapia, A., & Aguilar, W. G. (2018). Pose estimation based on monocular visual odometry and lane detection for intelligent vehicles. En *Lecture Notes in Computer Science* (págs. 562-566).
- Galindo, R., Aguilar, W. G., & Reyes Ch, R. P. (2019). Landmark based eye ratio estimation for driver fatigue detection. En *Lecture Notes in Computer Science* (págs. 565-576). Germany: Springer.
- Gašparović, M., & Jurjevic, L. (2017). Gimbal Influence on the Stability of Exterior Orientation Parameters of UAV Acquired Images. *Sensors*, 17(2), 401. doi:<https://doi.org/10.3390/s17020401>
- Gevers, M. (1993). Towards a joint design of identification and control? En *Essays on Control* (págs. 111-151). Springer.
- Gómez, O., & Machado, A. (2017). Arquitectura de integración basada en socket para sistemas distribuidos. *Espacios*.
- GreenWaves. (2012). Building occupancy management solution using the.
- Grijalva, S., & Aguilar, W. G. (2019). Landmark-Based Virtual Path Estimation for Assisted UAV FPV Tele-Operation with Augmented Reality. En *Lecture Notes in Computer Science* (págs. 688-700).

- Grijalva, S., Cueva, G., Ramírez, D., & Aguilar, W. G. (2019). HSVM-Based Human Activity Recognition Using Smartphones. En *Lecture Notes in Computer Science* (págs. 217-228).
- Guaraca, M. (2012). Uso de sockets en la informática: Una Revisión Sistemática de la literatura.
- Hernandez, A., & Garcia, L. (2014). LA METODOLOGÍA DE TAGUCHI EN EL CONTROL ESTADÍSTICO DE LA CALIDAD. *Universidad Tecnológica Nacional-Facultad Regional Bahía Blanca*.
- Hernandez, U. (2015). Sockets en Java (cliente y servidor). *Academia*.
- Hilkert, J. (2008). «Inertially Stabilized Platform Technology Concepts and Principles. *«Inertially Stabilized Platform Technology Concepts and Principles, 28(1), 26-46.*  
doi:<https://doi.org/10.1109/MCS.2007.910256>
- Howard, A., Zhu, M., & Kalenichenko, D. (2017). MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *arXiv*.
- Hudson, B., Hagan, M., & Demuth, H. (s.f.). Deep Learning Toolbox Getting Started Guide. 130. Obtenido de <https://la.mathworks.com/discovery/convolutional-neural-network-matlab.html>
- Janiesch, C., Zschech, P., & Heinrich, K. (2021). Machine Learning and Deep Learning. *Electronic Markets, 31(3), 685-95.* doi:<https://doi.org/10.1007/s12525-021-00475-2>
- Jara-Olmedo, A., Medina-Pazmiño, W., Mesías, R., Araujo-Villaroel, B., Aguilar, W. G., & Pardo, J. A. (2018). Interface of Optimal Electro-Optical/Infrared for Unmanned Aerial Vehicles. En *Smart Innovation, Systems and Technologies* (págs. 372-380).
- Jara-Olmedo, A., Medina-Pazmiño, W., Tozer, T., Aguilar, W. G., & Pardo, J. A. (2018). E-services from Emergency Communication Network: Aerial Platform Evaluation. *International Conference on eDemocracy & eGovernment (ICEDEG)* (págs. 251-256). IEEE.

- Joby, A. (2016). *Socket programming wifi chat app for android smartphone*.
- Juan, M. (2019). *Fundamentos de radiacion y radiocomunicacion*.
- Kanospe, C. (2006). PID Controls. *IEEE Control Systems*, 26(1), 30-31.  
doi:<https://doi.org/10.1109/MCS.2006.1580151>
- Ljung, L. (2008). Perspectives on System Identification. *IFAC Proceedings Volumes*, 7172-7184.
- Manrique, E. (2019). Machine Learning: análisis de lenguajes de programación y herramientas para desarrollo. *Machine Learning*, 15.
- Martinez, P. (2018). «Modulación de coherencia óptica con dispositivos electro-ópticos con aplicaciones en detección de campos electricos. Obtenido de [http://catarina.udlap.mx/u\\_dl\\_a/tales/documentos/lem/martinez\\_p\\_by/capitulo3.pdf](http://catarina.udlap.mx/u_dl_a/tales/documentos/lem/martinez_p_by/capitulo3.pdf)
- Massiris, M., Delrieux, C., & Fernández, J. Á. (2020). Detección de equipos de protección personal mediante red neuronal convolucional YOLO. *Actas de las XXXIX Jornadas de Automática. 1022*, págs. 5-7. Badajoz: Universidade da Coruña. Servizo de Publicacións.  
doi:<https://doi.org/10.17979/spudc.9788497497565.1022>.
- Mederos, M. (2020). *Primer Acercamiento al Android Studio*. Habana.
- Menéndez, G. (2019). Diseño generativo: el estado del arte. *Técnica Industrial*, 44-49.
- Merchán Fernando, G. S. (2014). Mejoras en el entrenamiento de esquemas de detección de sonrisas basados en AdaBoost. *I+ D Tecnológico*, 17-30.
- Merizalde, D., Aguilar, W. G., & Calderón, M. (2020). Autonomous Navigation Based on Proportional Controller with GPS Setpoint for UAV in External Environments. En *Smart Innovation, Systems and Technologies* (págs. 89-99).
- Michael, K. (2013). *Proceedings of the Twenty-Fourth ACM Symposium on Operating Systems Principles*. New York, NY: ACM.

- MobileNet. (2014). *MobileNetV2 SSD FPN*. Obtenido de <https://docs.edgeimpulse.com/docs/edge-impulse-studio/learning-blocks/object-detection/mobilenetv2-ssd-fpn>
- Molina, E. (2016). Protocolo de datagramas de usuario (UDP). *redtauros*.
- Muñoz, C. (2016). *DISEÑO Y DESARROLLO DE UN APLICATIVO MÓVIL EN LA PLATAFORMA ANDROID STUDIO PARA LA EMPRESA ALLIED ELECTRONICS & Pereira*.
- Murillo, J. (2010). IMPLEMENTACIÓN DE UN SERVIDOR FTP UTILIZANDO EL MODELO CLIENTE/SERVIDOR MEDIANTE EL USO DE SOCKETS EN LENGUAJE C UNIX CON EL FIN DE MEJORAR LOS TIEMPOS DE RESPUESTA EN LA RED. *uniciencia*.
- Murillo, J. (2012). *Fundamentos de radiación y radiocomunicación*.
- Musalem, R. (3 de Mayo de 2019). «Control Mediante Lógica difusa. Obtenido de <http://www2.elo.utfsm.cl/~elo377/documentos/Fuzzy.pdf>
- Orbea, D., Moposita, J., Aguilar, W. G., Paredes, M., León, G., & Jara-Olmedo, A. (2017). Math Model of UAV Multi Rotor Prototype with Fixed Wing Aerodynamic Structure for a Flight Simulator. En *Lecture Notes in Computer Science* (págs. 199-211).
- Orbea, D., Moposita, J., Aguilar, W. G., Paredes, M., Reyes, R. P., & Montoya, L. (2017). Vertical take off and landing with fixed rotor. *Chilean Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON)*. Pucón, Chile.
- Oscar, A. (2018). Mobile Object Detection using TensorFlow Lite and Transfer Learning.
- Pang, B., Nijkamp, E., & Wu, Y. N. (2020). Deep Learning With TensorFlow: A Review. *Journal of Educational and Behavioral Statistics*, 45(2), 227-48. doi:<https://doi.org/10.3102/1076998619872761>
- Pang, H., & Jiang, L. (2010). Research of Android Smart Phone Surveillance System. *IEEE Xplore*.

- Pardo, J. A., Aguilar, W. G., & Toulkeridis, T. (2017). Wireless communication system for the transmission of thermal images from a UAV. *Chilean Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON)*. Pucón, Chile.
- Precup, R.-E. (2011). A survey on industrial applications of fuzzy control. *Computers in Industry*, 213-226.
- Runia, M. (2015). Servidor web Raspberry Pi. *Internacional Sistemas Embebidos y Aplicaciones*.
- Salazar, J. (2005). *Redes inalámbricas*. Upper Saddle River, NJ: Pearson Prentice Hall. Obtenido de <http://books.google.com/books?id=MvdSAAAAMAAJ>
- Salcedo, V. S., Aguilar, W. G., Cobeña, B., Pardo, J. A., & Zahira, P. (2018). On-board target virtualization using image features for UAV autonomous tracking. En *Lecture Notes in Computer Science* (págs. 384-391).
- Segarra, D., Caballeros, J., Aguilar, W. G., Samà, A., & Rodríguez-Martín, D. (2019). Orientation Estimation Using Filter-Based Inertial Data Fusion for Posture Recognition. En *Lecture Notes in Computer Science* (págs. 220-233).
- Telechanda, B., Villacís, M., & Rosales, A. (Junio de 2019). «Control PTZ de un Sistema Electro-Óptico utilizando una Plataforma Robótica».
- Tenezaca, D., Canchignia, C., Aguilar, W. G., & Mendoza, D. (2019). Implementation of Dubin Curves-Based RRT\* Using an Aerial Image for the Determination of Obstacles and Path Planning to Avoid Them During Displacement of the Mobile Robot. En *Smart Innovation, Systems and Technologies* (págs. 205-215).
- Ulrich, K., & Eppinger., S. (2013). *Diseño y desarrollo de productos*.
- Vásquez, A. (2017). *GEOMETRÍA DE MASAS*.
- Velasco, R. (3 de Junio de 2014). *Estudio de la aplicación del diseño generativo al diseño conceptual arquitectónico*. Obtenido de <https://riunet.upv.es/bitstream/handle/10251/54972/VELASCO%20-%20EGA-F0066%20Estudio%20de%20la%20aplicaci%C3%B3n%20del%20dise%C3%B>

- 1o%20generativo%20al%20dise%C3%B1o%20conceptual%20arquitect....pdf?sequence=3.
- Villada , F., & Muñoz, N. (2016). Redes Neuronales Artificiales aplicadas a la Predicción del Precio del Oro. *Scielo*.
- Ville, M. (2012). Evaluating the sensitivity of lightweight object detection models against adversarial perturbations. 109.
- Welch, G. (1997). *An Introduction to the Kalman Filter* (Vol. 6).
- Welch, G., & Bishop, G. (2019). 82.
- Wongkamchang, P., & Sangveraphunsiri, V. (2013). *Department of Mechanical Engineering*. Obtenido de Control of Inertial Stabilization Svstems Using Robust Inverse Dynamics Control And Adaptive Control.
- Yacuzzi, E., & Martín, F. (2004). EL DISEÑO EXPERIMENTAL Y LOS MÉTODOS DE TAGUCHI: CONCEPTOS Y APLICACIONES EN LA INDUSTRIA FARMACÉUTICA. *Universidad del CEMA*, 4.
- Yi Lin, T., & Maire, M. (2014). Microsoft COCO: Common Objects in Context.
- Zurita, M. (2018). *DISEÑO DE UN SISTEMA DE GEOLOCALIZACIÓN DE OBJETIVOS EN TIERRA BASADO EN EL SENSOR ELECTROOPTICO SEO D1(SEO D2)*. Universidad de las Fuerzas Armadas ESPE, Quito.
- Zurita, M., Aguilar, W. G., & Enríquez, V. (2019). Toward the Development of Surveillance and Reconnaissance Capacity in Ecuador: Geolocation System for Ground Targets Based on an Electro-Optical Sensor. En *Smart Innovation, Systems and Technologies* (págs. 123-133).

## Apéndices