



UNIVERSIDAD TÉCNICA DE AMBATO

**FACULTAD DE INGENIERÍA EN SISTEMAS, ELECTRÓNICA E
INDUSTRIAL**

CARRERA DE INGENIERÍA INDUSTRIAL

Tema:

**SISTEMA DE LOCALIZACIÓN DE PERSONAS EN CATÁSTROFES
UTILIZANDO VEHÍCULOS AÉREOS NO TRIPULADOS (UAV'S)**

Trabajo de titulación modalidad Proyecto de Investigación, presentado previo a la
obtención del título de Ingeniero Industrial

ÁREA: Automatización

LÍNEA DE INVESTIGACIÓN: Tecnologías de la información y Sistemas de
control

AUTOR: Fabian Misael Cango Sarango

TUTOR: Ing. Franklin Wilfrido Salazar Logroño, Mg.

Ambato – Ecuador

febrero – 2024

APROBACIÓN DEL TUTOR

En calidad de tutor del trabajo de titulación con el tema: SISTEMA DE LOCALIZACIÓN DE PERSONAS EN CATÁSTROFES UTILIZANDO VEHÍCULOS AÉREOS NO TRIPULADOS (UAV'S), desarrollado bajo la modalidad Proyecto de Investigación por el señor Fabian Misael Cango Sarango, estudiante de la Carrera de Ingeniería Industrial, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, me permito indicar que el estudiante ha sido tutorado durante todo el desarrollo del trabajo hasta su conclusión, de acuerdo a lo dispuesto en el Artículo 17 del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato y el numeral 6.3 del instructivo del reglamento referido.

Ambato, febrero 2024.

Ing. Franklin Wilfrido Salazar Logroño, Mg.

TUTOR

AUTORÍA

El presente trabajo de titulación con el tema: SISTEMA DE LOCALIZACIÓN DE PERSONAS EN CATÁSTROFES UTILIZANDO VEHÍCULOS AÉREOS NO TRIPULADOS (UAV'S), es absolutamente original, auténtico y personal y ha observado los preceptos establecidos en la Disposición General Quinta del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato. En tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, febrero 2024.



Fabian Misael Cango Sarango

C.C. 1104826407

AUTOR

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato para que reproduzca total o parcialmente este trabajo de titulación dentro de las regulaciones legales e institucionales correspondientes. Además, cedo todos mis derechos de autor a favor de la institución con el propósito de su difusión pública, por lo tanto, autorizo su publicación en el repositorio virtual institucional como un documento disponible para la lectura y uso con fines académicos e investigativos de acuerdo con la Disposición General Cuarta del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato.

Ambato, febrero 2024.



Fabian Misael Cango Sarango

C.C. 1104826407

AUTOR

APROBACIÓN DEL TRIBUNAL DE GRADO

En calidad de par calificador del informe final del trabajo de titulación presentado por el señor Fabian Misael Cango Sarango, estudiante de la Carrera de Ingeniería Industrial, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, bajo la Modalidad Proyecto de Investigación, titulado SISTEMA DE LOCALIZACIÓN DE PERSONAS EN CATÁSTROFES UTILIZANDO VEHÍCULOS AÉREOS NO TRIPULADOS (UAV'S), nos permitimos informar que el trabajo ha sido revisado y calificado de acuerdo al Artículo 19 del Reglamento para la Titulación de Grado en la Universidad Técnica de Ambato y el numeral 6.4 del instructivo del reglamento referido. Para cuya constancia suscribimos, juntamente con la señora Presidente del Tribunal.

Ambato, febrero 2024.

Ing. Elsa Pilar Urrutia Urrutia, Mg.
PRESIDENTE DEL TRIBUNAL

Ing. Santiago Altamirano Meléndez, Mg.
PROFESOR CALIFICADOR

Ing. Jesús Guamán Molina, Mg.
PROFESOR CALIFICADOR

DEDICATORIA

A mis padres Benigno y Ana Victoria que se han preocupado para que nada me falte y que pueda sobrellevar la carrera siempre con éxito.

A mi abuelita Petrona que siempre ha estado orando por mí y encomendando mis actividades ante Dios.

A mis abuelitos Delfina Sarango y Antonio Guallas por darme un hogar en un lugar desconocido.

A mis amigos y compañeros que nos apoyamos desde el inicio de la vida universitaria.

AGRADECIMIENTO

Agradecimiento profundo a mis padres Ana Sarango y Benigno Cango, a mi hermana Leidy, a mis abuelitas Petrona y Delfina por ser parte fundamental para poder sobrellevar mi carrera sin rendirme.

Agradecimiento sincero al Ing. Franklin Salazar que ha estado pendiente para que el proyecto se desarrolle con éxito.

Agradecimiento a mis compañeros Steven, Francis, Betty, Mario y Andrés que me han apoyado en diferentes fases del proyecto.

ÍNDICE GENERAL DE CONTENIDOS

PORTADA	i
APROBACIÓN DEL TUTOR	ii
AUTORÍA	iii
DERECHOS DE AUTOR	iv
APROBACIÓN DEL TRIBUNAL DE GRADO	v
DEDICATORIA	vi
AGRADECIMIENTO	vii
ÍNDICE GENERAL DE CONTENIDOS	viii
ÍNDICE DE TABLAS	xii
ÍNDICE DE FIGURAS	xiv
ÍNDICE DE ANEXOS	xix
RESUMEN EJECUTIVO	xx
ABSTRACT	xxi
CAPÍTULO I. MARCO TEÓRICO	1
1.1 Tema de investigación.....	1
1.1.1 Planteamiento del problema	1
1.2 Antecedentes investigativos	2
1.3 Fundamentación teórica	5
1.3.1 Vehículos Aéreos No Tripulados (UAV's).....	5

1.3.2 Aplicaciones de los UAV's en diferentes campos	7
1.3.3 Espectro electromagnético	10
1.3.4 Cámaras térmicas	11
1.3.5 Cámaras RGB	16
1.3.6 La física de la observación de imágenes de gama de infrarrojos térmicos TIR	18
1.3.7 Distancia de muestreo en el suelo (Ground Sample Distance, GSD)	25
1.3.8 Baterías LiPo para UAV	25
1.3.9 Sistemas FPV	27
1.3.10 Aprendizaje automático	30
1.3.11 Detección de personas	30
1.3.12 Red convolucional YOLO	33
1.3.13 Herramientas para el procesamiento de imágenes	35
1.4 Objetivos	36
1.4.1 Objetivo general	36
1.4.2 Objetivos específicos	36
CAPÍTULO II.- METODOLOGÍA.....	37
2.1 Materiales	37
2.2 Métodos.....	39
2.2.1 Modalidad de la investigación	39
2.2.2 Recolección de información.....	41
2.2.3 Procesamiento y análisis de datos	41

CAPÍTULO III.- RESULTADOS Y DISCUSIÓN	42
3.1 Desarrollo de la propuesta.....	42
3.1.1 Identificación y análisis de sistemas de localización y detección de personas en situaciones de catástrofe.....	42
3.1.2 Selección del sistema de localización	48
3.1.3 Requerimiento del sistema	49
3.1.4 Esquema general del sistema	51
3.1.5 Selección de los elementos de hardware para el sistema de localización	53
3.1.6 Altura de vuelo del UAV	64
3.1.7 Diseño eléctrico del sistema de localización térmico	65
3.1.8 Diseño eléctrico del sistema de localización RGB	67
3.1.9 Implementación del esquema eléctrico	68
3.1.10 Implementación de los diseños en 3D.....	68
3.1.11 Emisividad y ángulo de visión de la cámara (FOV)	69
3.1.12 Verificación del peso de todos los componentes.	72
3.1.13 Tiempo de vuelo.....	72
3.1.14 Configuración de vuelos programados autónomos en el software Ardupilot. 75	
3.1.15 Configuración del OSD.....	79
3.1.16 Configuración del sistema de captura de video y detección de personas	80
3.1.17 Resultados de los vuelos	103
CAPÍTULO IV.- CONCLUSIONES Y RECOMENDACIONES.....	110

4.1 Conclusiones	110
4.2 Recomendaciones.....	111
REFERENCIAS BIBLIOGRÁFICAS.....	112
ANEXOS.....	116

ÍNDICE DE TABLAS

Tabla 1. Subdivisión de espectros infrarrojos	11
Tabla 2. Características de los transmisores y receptores de video	29
Tabla 3. Materiales.....	37
Tabla 4. Softwares.....	38
Tabla 5. Procesamiento y análisis de datos	41
Tabla 6. Matriz de enfrentamiento	48
Tabla 7. Escala de calificación.....	48
Tabla 8. Evaluación de alternativas por el método de factores.....	49
Tabla 9. Requerimientos del sistema	51
Tabla 10. Esquema general del sistema y sus etapas	52
Tabla 11. Características generales de los UAV's	53
Tabla 12. Medidas de la estructura del UAV Matek.....	54
Tabla 13. Características del motor Racerstar.....	54
Tabla 14. Controlador eléctrico de velocidad	55
Tabla 15. Tarjeta controladora de vuelo	55
Tabla 16. Placa de distribución de energía.....	55
Tabla 17. Mini receptor Fky Sky	56
Tabla 18. Transmisor Fly Sky	56
Tabla 19. Batería FUN FLY.....	57

Tabla 20. Características del GPS	57
Tabla 21. Características de cámaras RGB	59
Tabla 22. Características de las cámaras térmicas	60
Tabla 23. Comparativa entre cámaras térmicas y RGB	61
Tabla 24. Características de transmisores de video	63
Tabla 25. Receptores de video	63
Tabla 26. Datos para calcular el GSD	65
Tabla 27. Consumo en amperios de los componentes	73
Tabla 28. Comandos para el vuelo automático	77
Tabla 29. Imágenes térmicas en diferentes escenarios.....	80
Tabla 30. Funciones de las métricas de modelos entrenados.....	90
Tabla 31. Análisis de resultados de dos redes neuronales entrenadas con diferentes épocas.....	92
Tabla 32. Resultado del entrenamiento de la red neuronal con YOLOv8l 100 épocas	93
Tabla 33. Sobrevuelo a diferentes alturas	103
Tabla 34. Precisión en la detección de personas	106
Tabla 35. Imágenes capturadas a diferentes ángulos de visión.....	108

ÍNDICE DE FIGURAS

Figura 1. UAV de ala fija.....	6
Figura 2. Black Hornet Nano (Foto: Richard Watt).....	6
Figura 3. Tricóptero de YI Technology	7
Figura 4. Cuadricóptero DJI Mavic 3Pro.....	7
Figura 5. Dron hexacóptero WIRIS & DJI M600 Pro	7
Figura 6. Aplicaciones de los UAV's.	8
Figura 7. Espectro electromagnético.....	10
Figura 8. Espectro electromagnético y visible	10
Figura 9. Espectro electromagnético e infrarrojo.....	11
Figura 10. Paleta de colores térmicos [25].....	12
Figura 11. White hot	12
Figura 12. Sepia	13
Figura 13. Arco iris	13
Figura 14. Balck hot.....	14
Figura 15. Alerta exterior.....	14
Figura 16. Ártico	15
Figura 17. Arco de hierro	15
Figura 18 . Fusión de colores primarios; a) aditivo, b)sustractivo.....	16
Figura 19. Ilustración de un campo de visión.	19
Figura 20. Ejemplo de escala de píxeles deseada	20

Figura 21. Medio absorbente y medio detector.....	21
Figura 22. (a)Fracción de la luz de las fuentes transmitida al pasar a través del aire húmedo. (b) Cambio de temperatura observada real como resultado de la absorción y emisión por niebla.	22
Figura 23. Bandas MWIR y LWIR relativas a las ventanas atmosféricas.	24
Figura 24. Emisividad y FOV [27].	25
Figura 25. Diagrama básico FPV.	27
Figura 26. Sistema de transmisión y recepción de datos	28
Figura 27. Cámara FPV.	28
Figura 28. Sistema de transmisión	29
Figura 29. Estructura de un sistema de visión artificial [35]	31
Figura 30. Etapas de un sistema de visión artificial [34].	32
Figura 31. Detección de objetos con YOLO	33
Figura 32. mAP COCO vs número de Parámetros (pesos y sesgos) – YOLOv8 [36]	34
Figura 33. Detección de personas [38].....	43
Figura 34. Software Mission Planner [39]	44
Figura 35. Sistema de detección térmica.....	44
Figura 36. Funcionamiento del sistema LiDAR	45
Figura 37. Sistemas de Radio Frecuencia mediante UAV's [41]	46
Figura 38. Video transmisor VTX 801 con micrófono	47
Figura 39. Algoritmos de detección de personas	47

Figura 40. Diagrama de cableado [39].....	58
Figura 41. Conexión de la cámara y ESC [43].....	58
Figura 42. Diseños en 3D. a) Soporte, b) protector de para cámara Tau 2.....	62
Figura 43. Soportes para aterrizaje y despegue.....	62
Figura 44. Definición de pines del conector de E/S principal [45].....	66
Figura 45. Distribución de pines E/S.....	66
Figura 46. Esquema eléctrico de conexión.....	67
Figura 47. Diagrama de conexión de la cámara RGB.....	68
Figura 48. Implementación del esquema eléctrico.....	68
Figura 49. Implementación del diseño en 3D.....	69
Figura 50. Boceto de una plataforma aérea.....	71
Figura 51. Campo de visión proyectada en el suelo.....	72
Figura 52. Peso del UAV.....	72
Figura 53. Ejecución del software Mission Planner de Ardupilot.....	75
Figura 54. Conexión UAV a Mission Planner. a) Cable de datos; b) Software.....	75
Figura 55. Modos de Vuelo. a) Radio control; b) Software.....	76
Figura 56. Ajuste de velocidad de vuelo.....	76
Figura 57. Punto HOME o “H”.....	77
Figura 58. Trayectoria y verificar la altura.....	78
Figura 59. Escribiendo la trayectoria de vuelo en el UAV.....	78
Figura 60. Configurar OSD.....	79

Figura 61. Activación de los parámetros de OSD.....	79
Figura 62. Distribución de la base de datos	83
Figura 63. Software Makesense.ai	84
Figura 64. Detección de objetos en el software MakeSense	84
Figura 65. Crear etiquetas	85
Figura 66. Asignación de etiquetas	85
Figura 67. Exportación de las etiquetas.	86
Figura 68. Distribución de las imágenes etiquetadas	86
Figura 69. Versiones de YOLOv8 [36].....	87
Figura 70. Entorno de ejecución de Google Colab	87
Figura 71. Interfaz gráfica de YOLOv8 y base de datos.....	88
Figura 72. Descargar librerías	88
Figura 73. Instancia de modelo	88
Figura 74. Descomprimir archivos.zip.....	88
Figura 75. Archivo con extensión .yaml	89
Figura 76. Selección de rutas	89
Figura 77. Código para el entrenamiento de la red neuronal	89
Figura 78. Carpeta de resultados de la red neuronal	90
Figura 79. Resultados del entrenamiento de la red neuronal con YOLOv8l utilizando 80 épocas	91
Figura 80. Resultados del entrenamiento de la red neuronal con YOLOv8l utilizando 100 épocas	91

Figura 81. Librerías utilizadas.....	95
Figura 82. Lectura del modelo entrenado	96
Figura 83. Portada	96
Figura 84. Interfaz con portada	97
Figura 85. Cerrar la ventana.....	98
Figura 86. Captura del video.....	98
Figura 87. Segmentación del modelo.....	100
Figura 88. Visualización del video.....	100
Figura 89. Ventana principal de visualización de videos.....	102
Figura 90. Bucle de eventos	102
Figura 91. Precisión en la detección vs altura del UAV	106
Figura 92. Vuelo con varias personas en el FOV	107
Figura 93. Detección y localización de personas	109
Figura 94. Localización en Google Maps	109

ÍNDICE DE ANEXOS

Anexo A. Ficha de Cumplimiento	116
Anexo B. Reglamento de la aviación civil en Ecuador para el pilotaje de UAV's [46].	117
Anexo C. Código desarrollado en Python.....	117

RESUMEN EJECUTIVO

El avance de las nuevas tecnologías para sistemas de búsqueda y rescate a nivel mundial ha generado un desarrollo significativo en nuestro país, por ello esta investigación tuvo como objetivo el diseño de un sistema que permite la detección y localización de individuos en situaciones de catástrofe, mediante la integración de Vehículos Aéreos no Tripulados (UAV's), una cámara térmica y otra RGB.

Para incorporar la cámara térmica al UAV, se diseñó un soporte que permitió regular el ángulo de visión, y se crearon cuatro soportes impresos en poliuretano termoplástico para protegerla durante despegues y aterrizajes. Una vez en funcionamiento, se inició la captura de la firma térmica de la radiación del ser humano y su transmisión a la estación terrestre, donde se procesó el video en tiempo real para obtener una interfaz gráfica que permitió la visualización de la cámara térmica y la cámara RGB junto con los datos de localización.

El UAV fue probado a diferentes alturas, horarios y ubicaciones para simular escenarios reales, generando más de 1000 imágenes utilizadas para entrenar una red neuronal en YOLOv8l con una precisión del 90%. Tras las pruebas del sistema, se determina que la altura de vuelo óptima para la detección de la persona es de 16 metros, con una inclinación de la cámara térmica menor a 45 grados para evitar la reflectividad en el campo de visión.

Palabras claves: UAV's, cámaras térmicas, catástrofe, detección y localización.

ABSTRACT

The advancement of new technology for research and rescue systems has developed significantly in our country. Therefore, the objective of this investigation was to design a system which allows the detection and location of individuals in a case of a catastrophe, through the integration of Unmanned Aerial Vehicles (UAV's), a thermal camera and another RGB.

To incorporate the thermal camera, to the UAV, a support was designed to allow the viewing angle to be adjusted and four supports printed in thermoplastic polyurethane were created to protect it during takeoffs and landings. Once it begins functioning, the capture of the thermal signature of human radiation begins and its transmission to the ground station, where the video is processed in real time to obtain the graphic interface to allow the visualization of the thermal camera and RGB camera along with the data location.

The UAV was tested on different heights, times and locations to simulate realistic scenarios, generating more than 1000 images used to train a neural network in YOLOv8l with an accuracy of 90%. After various tests of the system, it was determined the optimal flight height for detecting a person is 16 meters, along with an inclination of the thermal camera less than 45 degrees to avoid reflectivity in the field of view.

Keywords: UAV's, thermal cameras, disaster, detection and location.

CAPÍTULO I. MARCO TEÓRICO

1.1 Tema de investigación

Sistema de localización de personas en catástrofes utilizando Vehículos Aéreos No Tripulados (UAV's).

1.1.1 Planteamiento del problema

Actualmente existen muchos desastres geológicos y de tránsito en el mundo, los cuales cobran vidas de cientos de personas, este tipo de desastres generalmente tienen la característica de brusquedad. Alrededor del mundo han existido una gran cantidad de víctimas y daños materiales por esta clase de eventos de movimientos en masa, debido a que numerosos asentamientos humanos se ubican en zonas de riesgos, los deslizamientos causados por lluvia son cada vez más comunes viéndose afectados tanto las personas como la naturaleza.

Los desastres en cualquier forma son perjudiciales para la salud y el bienestar de la población afectada, sin duda los desastres masivos afectan de manera desalmada a un gran número de víctimas, sean estos naturales o provocados por el hombre. Es el caso de la costa oeste de Sumatra Indonesia, en 2004 que es uno de los desastres naturales más devastadores de la historia con más de 90 000 personas fallecidas en Indonesia, por lo tanto, cuando ocurre un desastre, una respuesta rápida y eficaz es fundamental para ayudar a la población [1].

Nuestro país se encuentra en una zona sísmica y volcánica activa, por lo que los terremotos y deslizamientos de tierra son eventos naturales frecuentes. En los últimos años, el país ha experimentado varios terremotos significativos y deslizamientos de tierra. Uno de los terremotos más devastadores en la historia reciente de Ecuador, fue el terremoto de magnitud 7.8 que golpeó la costa ecuatoriana en la ciudad de Pedernales perteneciente a la provincia de Manabí, en abril de 2016. El sismo causó más de 661 fallecidas, 12 personas desaparecidas y 113 personas rescatadas, así como también importantes daños en infraestructuras y viviendas [2].

La provincia de Tungurahua se compone por seis mesetas: Píllaro, Quizapincha, Tisaleo, Quero, Huambaló y Cotaló, lo cual genera muchas pendientes en la provincia, y en los últimos 12 años ha triplicado su población, lo que ha determinado la utilización irracional del terreno, incluidas zonas de alta peligrosidad ante diversas catástrofes naturales, causando riesgo a la población y la infraestructura civil [3].

Pero ésta no es una problemática solo de la ciudad de Ambato, pues estos eventos ocurren en todo el Ecuador, en especial en la época de lluvias como lo menciona en [4] que durante el primer trimestre de 2023 las fuertes lluvias afectaron al menos 46 000 personas perjudicadas, y el 26 de marzo de 2023 en la ciudad de Alausí, en la provincia de Chimborazo, ocurrió un deslizamiento en donde hubo 38 fallecidos, 50 desaparecidos y 33 heridos lo cual es una cifra alarmante [5], que después de lo ocurrido el trabajo de rescate no suele ser pertinente debido a la situación geográfica y la falta de tecnología como los UAV's de rescate.

1.2 Antecedentes investigativos

Al realizar una detallada investigación bibliográfica al nivel nacional e internacional se logró encontrar investigaciones similares referentes al tema de investigación, los cuáles se utilizará como soporte principal de nuestra investigación, entre ellos se mencionan:

En Reino Unido desarrollaron una investigación que titula “Requisitos y limitaciones de los UAV's térmicos para búsqueda y rescate efectivos en operaciones marinas y Áreas costeras”, tuvo como objetivo desarrollar un sistema de localización de personas basados en imágenes de infrarrojos térmicos TIR (del inglés, Thermal Infrared), en donde se demostró que los UAV's equipados con infrarrojos térmicos tienen el potencial de usarse para búsqueda y rescate, ya que las imágenes que analizan son de tipo TIR. Las imágenes TIR permite diferenciar cuerpos calientes a diferencia de las imágenes de espectro visible (RGB) que todos los objetos aparecen con el mismo brillo, con estos análisis sobrevolaron un dron a diferentes alturas, con la cámara térmica FLIR Duo Pro R. Durante el sobrevuelo y el análisis de los datos obtenidos, a una altura de 50 msnm, un humano bocabajo de 1,7 m de largo aparecía con 25 píxeles de longitud, un humano visto desde arriba con la cabeza y los hombros de 0,5 m de

ancho tendría nueve píxeles, a una altura de 100 msnm, estos tamaños fueron de 15 píxeles (boca abajo) y 4 píxeles (vertical), respectivamente. En el estudio realizan un análisis de escenarios en donde se analiza la temperatura durante diferentes horas del día y concluyen que diferentes factores como el bosque, las rocas y el césped pueden variar la temperatura del cuerpo humano y con ello su detectabilidad que depende en gran medida de la diferencia de la temperatura del humano y su entorno [6].

Un estudio realizado en el Instituto de Investigación de Astrofísica, Universidad John Moores de Liverpool, en Reino Unido que titula “Optimización de estrategias de observación para el seguimiento de animales mediante cámaras infrarrojas térmicas montadas en drones”. En este artículo abordan principalmente problemas de contraste térmico debido al calor del suelo, la absorción y emisión de radiación infrarroja térmica por parte de la atmósfera, el oscurecimiento por la vegetación, efecto de la niebla, la optimización de vuelo de los UAV’s para un mejor equilibrio que con lleva cubrir un área grande y poder obtener imágenes e identificar con precisión los animales de interés, con la finalidad de mejorar la eficiencia del monitoreo TIR de animales. El sistema utilizado de UAV fue un DJI Matrice 600 con una cámara FLIR T640 TIR que cuenta con un lente de 13 mm y su finalidad fue detectar un conejo caribeño. Tomando en cuenta el tamaño y la resolución de la cámara obtienen la altura de vuelo exacta para detectarlo con la finalidad de que pareciera con al menos 10 píxeles en el campo de visión de la cámara [7].

Una investigación desarrollada por el politécnico de Turín, Italia [8], que titula “Programación de Tareas de emergencia para UAV’s multiservicio en escenarios Post desastre”, tuvo como objetivo una planificación multitarea aprovechando una flota de vehículos aéreos no tripulados equipados con un conjunto estándar de accesorios, sistema de monitoreo de video, interfaz de comunicación celular y un marco de montaje para el transporte de paquetes. En [8] desarrollaron dos algoritmos heurísticos, en los cuáles se toman en cuenta características como el arco de recorrido, tiempo de viaje, energía consumida y la capacidad residual de carga útil, esto con la finalidad de calcular que tipo de carga que puede llevar, cuantos minutos puede estar en vuelo, esto depende de la capacidad máxima de transmisión de la interfaz inalámbrica a bordo del UAV [8].

En una investigación desarrollada en Austria, tuvieron como objetivo la detección automatizada de personas en condiciones de poca visibilidad, que se puede mejorar significativamente al combinar imágenes de múltiples perspectivas antes de la clasificación de personas con otros objetos [9]. Emplearon la integración de imágenes mediante AOS (del inglés, Airborne Optical Sectioning), una técnica de imágenes de apertura sintética que utiliza cámaras no tripuladas para capturar campos de luz térmica no estructurados, para lograr esto con una precisión de recuperación de 96 y 93 %. Con la finalidad de encontrar personas perdidas o heridas en bosques densos, que generalmente no es factible con registros térmicos, pero se vuelve práctico con el uso de imágenes integrales AOS [9].

En Morelos, realizaron una investigación que titula “Robot aéreo seguidor de líneas usando visión artificial”, en [10] tiene como objetivo principal desarrollar un sistema de visión artificial que permita a un dron identificar en tiempo real una línea de color contrastante en el suelo, determinar su ubicación y seguirla. El sistema que desarrollaron fue diseñado para un dron modelo Bebop de la empresa francesa Parrot. Las pruebas realizadas fueron en un circuito utilizando cinta adhesiva para que el dron pudiera seguirlo [10], logrando esto con la conexión inalámbrica y el procesamiento digital de la imagen; en el que consiste un recorte de la imagen para obtener una región de interés, conversión de la imagen a HSV (del inglés, Hue Saturation Brightness), obtención de una máscara para discriminar colores, aplicación de la máscara a la imagen original, binarización de la imagen por medio de umbralización, obtención del centroide de la línea, y por último la transmisión de velocidades manteniendo al dron siempre en el centroide [10].

En Quito, desarrollaron una investigación para identificación de personas, el cual tiene como objetivo principal desarrollar un sistema de reconocimiento facial para la identificación de personas con un UAVS mediante redes neuronales artificiales [11]. Para desarrollar el sistema de identificación utilizaron el lenguaje de programación Python, en base a TensorFlow y Keras para crear la red neuronal artificial y HaarCascade como método de detección de rostros cuyo código fue albergado en una Laptop Lenovo que se conectara con el UAV para la adquisición de datos, preprocesamiento de imágenes, la comparación y procesamiento digital para el aprendizaje de la red neuronal [11].

1.3 Fundamentación teórica

A continuación, se detalla el marco teórico que se ha recopilado de diferentes fuentes bibliográficas para la sustentación de la investigación.

1.3.1 Vehículos Aéreos No Tripulados (UAV's)

Los UAV's (del inglés, Unmanned Aerial Vehicle, también conocidos como drones) comparten sus orígenes con la aviación tripulada, en [12] menciona que los pioneros como Cayley, Ninomiya, Du Temple, Langley, crearon las primeras aeronaves no tripuladas.

a. Sistemas de los UAV's

Un UAV es un sistema que lleva muchos subsistemas en su interior al mismo tiempo, pero forma parte de un sistema más complejo que compone varios elementos, tales como plataformas aéreas y terrestres, satélites y comunicaciones entre UAV [13]. La parte aérea se compone por tres sistemas diferentes: La plataforma aérea, la carga útil y una parte del sistema de comunicación. La parte terrestre incluye diversos sistemas: el sistema de control del UAV, equipo de comunicaciones y la estación de control terrestre (en inglés, Ground Control Station, GCS). La GCS permite la monitorización de todos los sensores por parte de los usuarios, todo esto para lograr el control sobre el UAV para diferentes misiones [13].

b. Clasificación de los UAV's

Los UAV's se clasifican principalmente en dos categorías distintas: UAV de ala fija y UAV de ala rotatoria [14]. A continuación, se describe detalladamente cada una de estas clasificaciones.

- ***UAV's de ala fija.*** Los UAV's de ala fija son aquellos que aprovechan de la aerodinámica de sus alas para su despegue horizontal, en la geometría de sus alas se puede encontrar con diferentes diseños como: convencional, canard, ala volante, delta, joined wing, rectangular o trapezoidal, con o sin flecha, ala alta, alas media o baja, también es posible clasificarlos en función de la geometría de su cola; estabilizador horizontal de cola, cola en V y doble cola [12]. El UAV's de ala fija se puede visualizar en la Figura 1.



Figura 1. UAV de ala fija

- ***UAV's de ala rotatoria o multirrotor.*** Los UAV's de ala giratoria aprovechar la sustentación que generan las palas del rotor para elevarse. También existen diferentes tipos de UAV's de ala rotatoria y éstas puede clasificarse como: UAV's tipo helicóptero, UAV's tipo ciclo giro y dependiendo de la configuración de sus rotores [12].

En los UAV's de tipo helicóptero su despliegue puede ser vertical, volar hacia adelante, hacia detrás y lateralmente, usando uno o más motores durante el vuelo [12]. Unos de los UAV's estilo helicóptero como se muestra en la Figura 2.



Figura 2. Black Hornet Nano (Foto: Richard Watt).

Los tricópteros son modelos equipados con tres motores uno en cada brazo como se puede ver en la Figura 3. Generalmente giran en dirección contraria para un vuelo uniforme, mientras que el motor trasero aporta estabilidad al UAV [15].



Figura 3. Tricóptero de YI Technology

Los cuadricópteros, tienen 4 motores en cada brazo para ser propulsado como se muestra en la Figura 4, son los modelos más vendidos por su precio y comúnmente conocidos como UAV's[15].



Figura 4. Cuadricóptero DJI Mavic 3Pro

Los hexacópteros son modelos que ofrecen una notable estabilidad durante el vuelo, razón por la cual se emplean combinados en grabaciones de video profesional. Estos dispositivos están equipados con seis motores en cada brazo para garantizar su estabilidad [15], como se ilustra en la Figura 5.



Figura 5. Dron hexacóptero WIRIS & DJI M600 Pro

1.3.2 Aplicaciones de los UAV's en diferentes campos

Los UAV's presentan una amplia gama de aplicaciones, como se ilustra en la Figura 6. Su versatilidad u utilidad se extienden en diversos campos, entre los que destacan la agricultura, logística, militar, geociencia y ambiental.

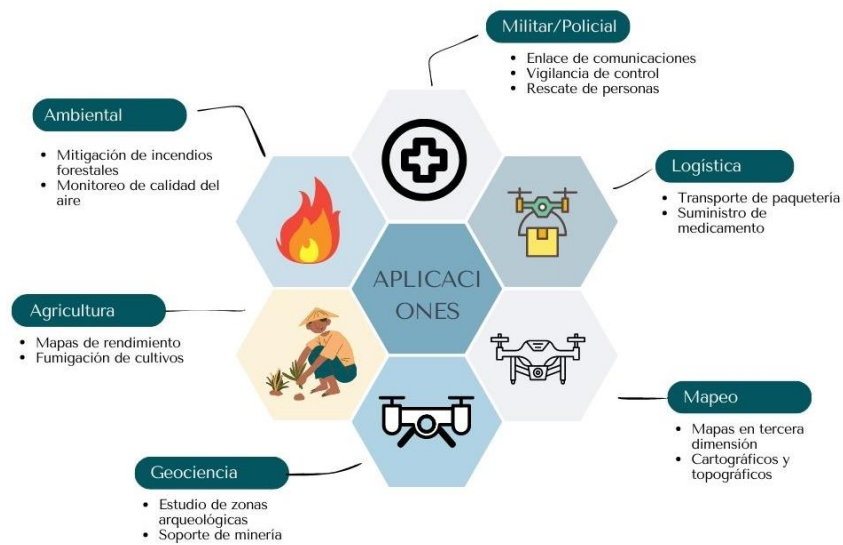


Figura 6. Aplicaciones de los UAV's.

a. Agricultura

En la agricultura los UAV's son utilizados en la elaboración, supervisión y control de sistemas agrícolas. Los métodos aplicados como las nubes de puntos, extracción geométrica de formas e inteligencia artificial como: Redes de Neuronas artificiales y también las máquinas de soporte vectorial, esto con la finalidad de generar mapas en tercera dimensión de la realidad del cultivo, predicción del rendimiento del cultivo e identificar posibles amenazas como plagas o malezas en los cultivos. Los UAV's han sido implementados para incorporar herramientas para fumigación de cultivos y riego de zanjas en diferentes tipos de cultivos[16].

b. Logística

En los UAV's se están implementando estrategias de transporte de paquetería, es el caso de la empresa ZIPLINE conocido también como los drones de sangre, ya que proveen acceso instantáneo a sangre a más de 11 millones de personas en Ruanda [17]. Estos vehículos han realizado misiones buscando desarrollar modelos a fin de presentar ventajas para que las empresas sean más rentables, por otra parte, estos vehículos cuentan con sistemas de supervisión por parte del consumidor y por parte de la empresa [18].

c. Militar

Los UAV's son parte principalmente de esta área, actualmente se usan para labores de control de inmigrantes, contrabando y comunicaciones. Estos vehículos han presentado avances basados en dispositivos electrónicos como sensores, radares, sistemas de posicionamiento global (en inglés, Global Position System GPS) y el sistema global de comunicaciones móviles (del inglés, Global System For Mobile Communications, GSM). Los UAV's se enfocan también en misiones de estabilidad de redes inalámbricas entre bases militares, retransmisiones de radio entre otras aplicaciones como el rescate de personas que para ellos se utiliza cámaras térmicas, micrófonos y cámaras de alta resolución para localizar personas [19].

d. Ambiental

Con el desarrollo de los UAV's se han acoplado para realizar censados, supervisión, prevención y cuidado de ecosistemas, bosques y animales. La información la obtienen mediante sensores térmicos, láser, biosensores y cámaras multiespectrales, estos sensores permiten obtener información detallada sobre condiciones ambientales, climatológicas, estrés híbrido y PH de ríos [20].

e. Geociencias

Los UAV's son herramientas innovadoras que permiten la evaluación de recursos mineros, estructuras rocosas y vestigios de construcciones antiguas. Estos dispositivos están equipados con una variedad de tecnologías avanzadas, incluyendo sensores de posicionamiento, cámaras térmicas, LiDAR (del inglés, Light Detection and Ranging) y cámaras RGB digitales [20].

f. Mapeo

Esta zona se beneficia del uso de UAV's para aplicaciones topográficas y de cartografía. Su función principal radica en la generación de mapas tridimensionales de alta resolución y precisión. Para lograr esto, se emplean sensores especializados como cámaras multiespectrales y sensores LiDAR [21].

1.3.3 Espectro electromagnético

El espectro electromagnético abarca las radiaciones electromagnéticas presentes en el espacio, propagándose en forma de ondas. Incluye una gama diversa, desde las ondas de baja frecuencia, como las de radio y microondas, hasta las de mayor energía y frecuencia alta, como los rayos X y los rayos gamma [22]. Esta representación se ilustra en la Figura 7. A partir de esta figura se derivan dos espectros de gran utilidad para el avance de tecnologías en la captura de imágenes RGB e imágenes térmicas. A continuación, se analizan detalladamente ambos factores.

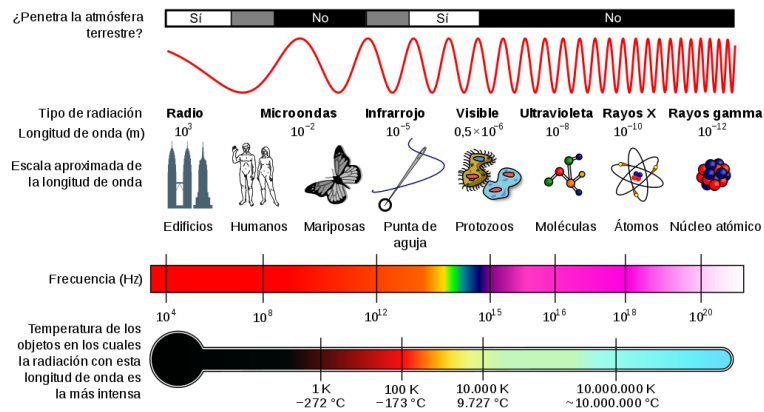


Figura 7. Espectro electromagnético

a. Espectro Visible

El espectro visible tanto para el ojo humano como para la mayoría de las cámaras RGB se encuentra en 400 nanómetros (nm) que pertenece al color violeta y 750 nm que pertenece al color rojo, en la Figura 8 muestra las radiaciones conocidas y sus rangos [22].

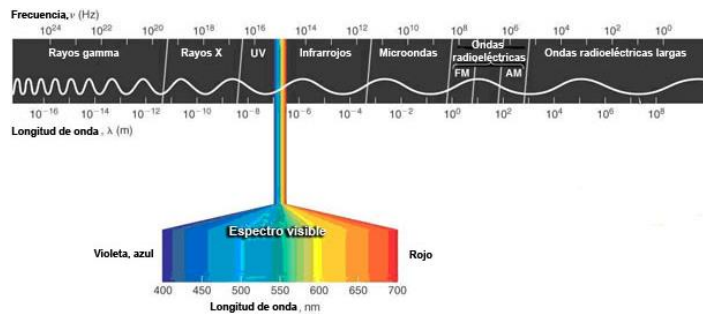


Figura 8. Espectro electromagnético y visible

b. Espectro infrarrojo

El espectro infrarrojo se encuentra por debajo del espectro visible, lo cual comprende longitudes de onda que varía desde 7 μm (Color rojo) hasta 14 μm , este fragmento a su vez se divide en sub espectros infrarrojos como se puede visualizar en la Tabla 1 [23].

Tabla 1. Subdivisión de espectros infrarrojos

Subdivisión de los espectros		Banda espectral
SIR	Short infrared (Infrarrojo corto)	0.9 – 1.7 μm
MIR	Middle infrared (Infrarrojo medio)	3 - 5 μm
FIR	Far infrared (Infrarrojo lejano)	7 – 14 μm

En la Figura 9 se muestra el espectro electromagnético focalizando el infrarrojo y sus subdivisiones que van desde 0.9 μm hasta 14 μm .

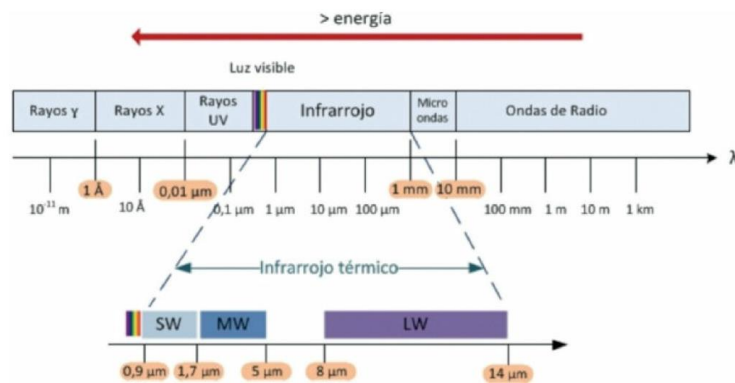


Figura 9. Espectro electromagnético e infrarrojo

1.3.4 Cámaras térmicas

Las cámaras térmicas, también conocidas como termográficas, son dispositivos que capturan la radiación infrarroja emitida por objetos, mide y convierte los datos infrarrojos en una imagen electrónica sin necesidad de contacto físico, permitiendo generar representaciones visuales basadas en el calor que emiten dichos objetos. Todos los objetos en la cámara térmica que tienen temperaturas mayores al cero absoluto, es decir -273.15 grados centígrados ($^{\circ}\text{C}$) emiten radiación infrarroja conocida como firma de calor [24]. En una cámara térmica cada píxel en la matriz del sensor reacciona a la energía infrarroja y produce una señal electrónica.

El procesador de la cámara toma cada píxel y lo procesa realizando un cálculo matemático para crear un mapa de calor de la temperatura del objeto, en el que cada color representa una temperatura diferente como se muestra en la Figura 10 que van desde los 10 °C hasta los 32 °C [25].

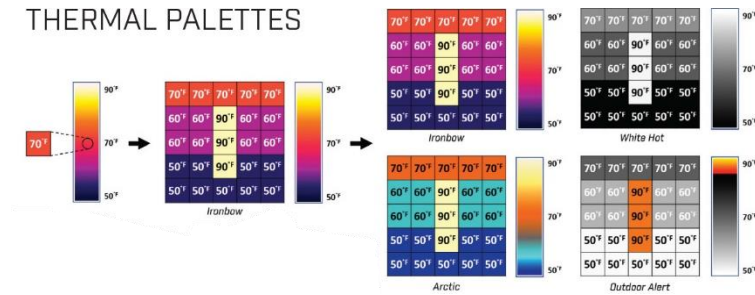


Figura 10. Paleta de colores térmicos [25]

Estas cámaras emplean paletas de colores como filtros, realizando píxeles con temperaturas específicas para mejorar el contraste y la nitidez de objetos identificables. Las paletas asignan colores según temperaturas precisas, ajustando tonos en áreas específicas para resaltar zonas clave sin perder información térmica general. [24]. Existen diferentes paletas de colores térmicas, como el blanco caliente (white hot), sepia, negro caliente (black hot), arco iris HC, alerta exterior, ártico y arco de hierro, a continuación, se explican cada uno de la paleta de colores [25].

a. White hot (Blanco caliente)

Esta gama de colores destaca los objetos con mayor temperatura en tonos blancos y aquellos más fríos en tonos negros, como se puede visualizar en la Figura 11. La versatilidad de la opción White Hot la hace especialmente útil para aplicaciones en entornos urbanos y paisajes que experimentan cambios constantes. [25].



Figura 11. White hot

b. Sepia

La paleta Sepia modifica la paleta White Hot al añadir un tono cálido y dorado como se puede ver en la Figura 12, lo que ayuda a disminuir el cansancio visual y mental, es especialmente adecuada para situaciones de vigilancia o exploración prolongada, ya que su espectro visual más específico proporciona comodidad a los usuarios durante largos periodos de observación [25].



Figura 12. Sepia

c. Rainbow HC (Arco iris)

En la Figura 13 se puede visualizar la paleta Rainbow HC la cual emplea una variedad de colores para resaltar variaciones mínimas de temperatura, lo que la hace perfecta para situaciones donde hay cambios térmicos pocos perceptibles. Al enfocarse en zonas con niveles similares de energía térmica, Rainbow HC puede identificar objetos y cambios leves de temperatura incluso en condiciones de contraste reducido[25].



Figura 13. Arco iris

d. Black hot (Negro caliente)

Black Hot es una variante inversa de White Hot, representando los objetos más calientes en tonos oscuros y los más fríos en tonos claros, en la Figura 14 se puede visualizar esta inversión de colores, comúnmente aplicada por fuerzas policiales y personas dedicadas a la caza, visualiza el calor corporal con gran claridad y realismo en las imágenes térmicas[25].

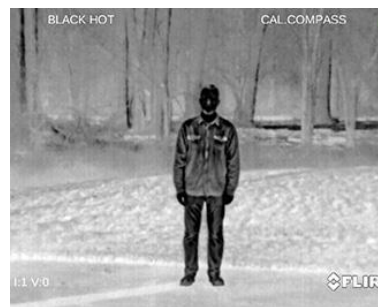


Figura 14. Balck hot

e. Alerta exterior (Outdoor Alert)

Esta paleta de colores se basa en el efectivo esquema Black Hot, Outdoor Alert fue creado con un objetivo específico, detectar de manera rápida el calor corporal. Al destacar el 10% más caliente de una escena con tonos vibrantes de naranja y amarillo, este esquema es ideal para entornos con contrastes marcados y proporciona una excelente detección del calor corporal durante la noche, como se puede ver en la Figura 15 [25].

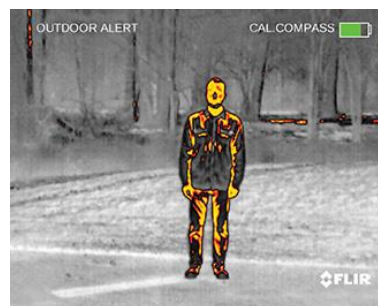


Figura 15. Alerta exterior

f. Ártico (Artic)

Esta paleta usando un tono dorado para señalar objetos calientes y tonos azules para los más fríos, con la capacidad de Rainbow HC para funcionar en entornos de bajo contraste, como se ilustra en la Figura 16. Esta combinación de colores permite identificar rápidamente las fuentes de calor utilizando diferentes tonalidades, mientras que los tonos más oscuros captan cambios sutiles de temperatura. [25].

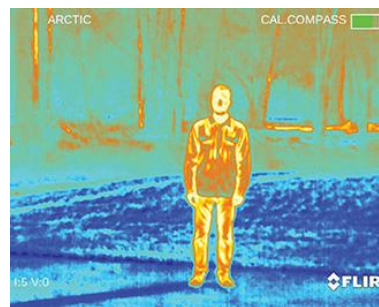


Figura 16. Ártico

g. Ironbow (Arco de hierro)

Es una paleta diseñada para detectar de manera veloz anomalías térmicas y el calor corporal, utiliza colores para representar la distribución del calor y resaltar detalles sutiles: los objetos calientes se presentan en tonos más cálidos y claros, mientras que los objetos más fríos se muestran en colores más fríos y oscuros, en la Figura 17 se puede visualizar estos colores [25].

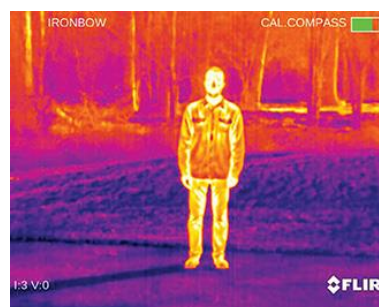


Figura 17. Arco de hierro

1.3.5 Cámaras RGB

Las cámaras RGB son dispositivos que mide la capacidad de luz dentro de un espectro electromagnético visible y los integra en tres colores primarios, rojo, verde y azul (RGB) que se visualiza como una imagen digital, mediante la fusión de colores como; primarios aditivos y sustractivos, como se puede ver en la Figura 18 [26].

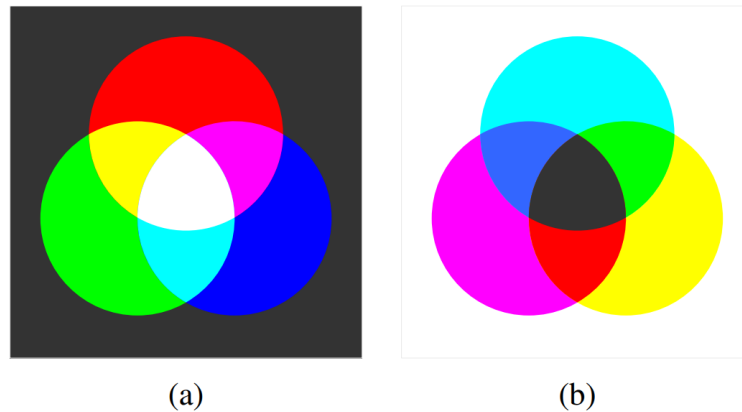


Figura 18 . Fusión de colores primarios; a) aditivo, b) sustractivo

Se destacan parámetros relevantes sobre las cámaras RGB como el tiempo de exposición, paso de muestreo, tamaño de viruta del sensor, sensibilidad ISO, ruido del sensor, espectro visible y aplicaciones en tecnología

a. Tiempo de Exposición

El tiempo de exposición controla la cantidad de luz que llega al sensor de la cámara. Este factor es esencial para determinar si una imagen estará sobreexpuesta (mucha luz) o subexpuesta (poca luz). Ajustar el tiempo de exposición permite capturar detalles en áreas tanto muy iluminadas como en zonas de poca luz.

b. Paso de Muestreo

El paso de muestreo se refiere al espacio físico entre las celdas del sensor de la cámara. Un sensor con un paso de muestreo pequeño tiene una mayor densidad de muestreo, lo que resulta en una mayor resolución de píxeles. Esta mayor resolución conlleva a imágenes más detalladas y nítidas.

c. Tamaño de Viruta del Sensor

El tamaño de la viruta del sensor se describe generalmente en pulgadas, comúnmente en tamaños como 1/4 o 1/2 de pulgada. Este factor influye en la cantidad de luz que puede capturar el sensor y puede afectar la calidad de la imagen, especialmente en condiciones de poca luz.

d. ISO (Sensibilidad a la Luz)

El ISO representa la sensibilidad del sensor a la luz. Una configuración de ISO más bajo necesita más luz para lograr una buena exposición, mientras que un ISO más alto puede capturar imágenes en entornos con menos luz, pero a menudo con más ruido.

e. Ruido del Sensor

El ruido del sensor está directamente relacionado con el tiempo de exposición y la ganancia del sensor ISO. Ajustar estos parámetros puede influir en la cantidad de ruido presente en una imagen, siendo una consideración crucial para obtener imágenes nítidas y claras.

f. Espectro Visible y Longitud de Onda

Las cámaras RGB capturan el espectro visible, que abarca longitudes de onda entre 0.4 y 0.7 micrómetros (μm). Este rango de longitud de onda corresponde a los colores que el ojo humano puede percibir, desde el violeta hasta el rojo. Las cámaras están diseñadas específicamente para capturar y procesar información dentro de este espectro, permitiendo la representación precisa de colores en las imágenes.

El control preciso de estos parámetros es esencial para obtener imágenes de alta calidad en fotografía y videografía aérea. Ajustar el tiempo de exposición, el ISO, el tamaño del sensor y otros factores permite a los profesionales capturar resultados óptimos en variadas condiciones de luz y escenarios. Comprender cómo estos parámetros interactúan entre sí es fundamental para lograr imágenes con la exposición correcta, nitidez y fidelidad de color, elementos cruciales en la captura aérea donde las condiciones lumínicas pueden ser cambiantes y variables.

1.3.6 La física de la observación de imágenes de gama de infrarrojos térmicos TIR

En [7] menciona que todos los objetos vivos emiten energía en forma de radiación electromagnética (Luz), la intensidad de luz del objeto depende directamente de la temperatura del mismo, generalmente los humanos les corresponde temperaturas corporales típicas de 20 – 30 °C, que se emiten en el rango de longitud de onda entre 8 y 14 micras ($8 \times 10^{-6} - 14 \times 10^{-6} m$) [7]. Esto determina que esta región del espectro electromagnético, conocida como TIR, sea una onda ideal para detectar seres vivos como animales o personas. La luminosidad L varía en función de la temperatura del objeto T , la Ecuación (1) se calcula la luminosidad.

$$L = A\sigma T^4 \quad (1)$$

Donde σ es la constante de Stefan-Boltzmann, A es el área de emisión en m^2 , todos los valores de temperatura se encuentran en unidad de Kelvin [7].

La luminosidad que se registra en la cámara termográfica puede verse afectada por varios factores como: condiciones meteorológicas, contraste de temperatura entre el objeto y el fondo, distancia entre el objeto de interés y la cámara, campo de visión FOV, efectos del oscurecimiento por la vegetación, propiedades físicas del pelaje del animal [27]. En [7], se demuestra la comprensión física a los problemas relacionados con la observación del entorno, el clima, la emisión de fondo, la distancia al objeto y el FOV de la cámara, la absorción por la atmósfera y el oscurecimiento por la vegetación.

a. Campo de visión (FOV) y altura del UAV

Para adquirir datos de un ser vivo homeotérmico, es importante tener datos como el tamaño, la temperatura corporal y el hábitat en caso de un animal. Para que una cámara térmica registre con precisión la temperatura de un objeto observado, el objeto debe aparecer más grande que un cierto tamaño mínimo en el campo de visión de la cámara, ya que cada píxel en un sensor térmico registra la temperatura de cualquier objeto dentro de la vista de ese píxel, si dentro de un píxel hay más de un objeto la temperatura

registrada será una combinación de las temperaturas de los dos objetos, como se puede visualizar en la Figura 19. Para aumentar la certeza de las mediciones de temperatura de un objeto de interés debe llenar por completo más píxeles de los que se comparten entre el objeto [7]. El área proyectada y el tamaño de los animales dentro del FOV dependen directamente de la altura del UAV.

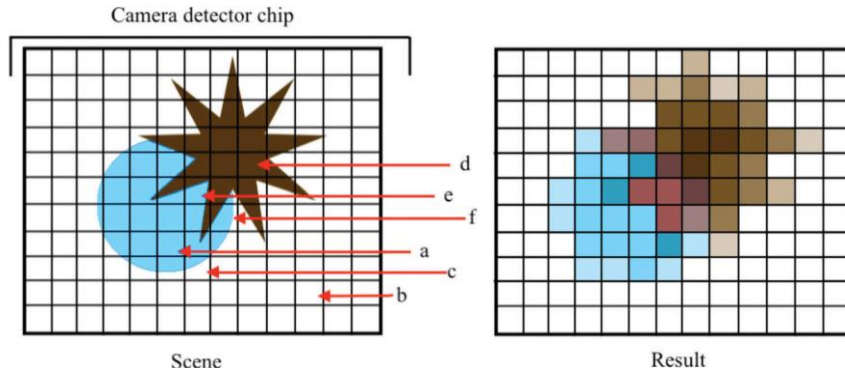


Figura 19. Ilustración de un campo de visión.

En la Figura 19, se puede visualizar la ilustración de un FOV de un chip detector dividido en una cuadrícula para representar los píxeles dentro de un detector. El campo de color azul es el objetivo de interés, un objeto que oscurece color café y el fondo blanco [7].

Cada cámara térmica tiene una escala de píxeles angulares intrínseca, que se define principalmente como el tamaño angular en grados que cubre cada píxel, mediante la Ecuación (2) [7].

$$\frac{\theta}{N_{\text{píxeles}}} = \rho_a. \quad (2)$$

Donde ρ_a es la escala angular en píxeles, θ es el FOV en grados, y $N_{\text{píxeles}}$ es el número de píxeles de la cámara o detector TIR. La escala angular de píxeles será aproximadamente la misma a los dos de los ejes horizontal x y eje vertical y. Como tal podemos definir la escala de píxeles, $\rho\rho$, de la cámara térmica, que es la longitud en X y Y direcciones cubiertas por cada píxel en metro en el suelo, en función de la altura “h” y está dada por la Ecuación (3);

$$h \tan(\rho_a) = \rho\rho. \quad (3)$$

El FOV proyectado de la cámara en el suelo en metros es simplemente la escala física de píxeles multiplicada por la cantidad de píxeles en el detector. Con esta relación, podemos calcular la altura de una cámara dada para alcanzar un área objetivo mínimo, longitud o ancho en el FOV, y el área cubierta en el FOV a esta altura. En la Figura 20 se presenta un ejemplo ilustrado en donde se supone que la escala de píxeles deseada está en el centro del campo de visión para una configuración en ángulo, el valor de la altura se convierte en R_M .

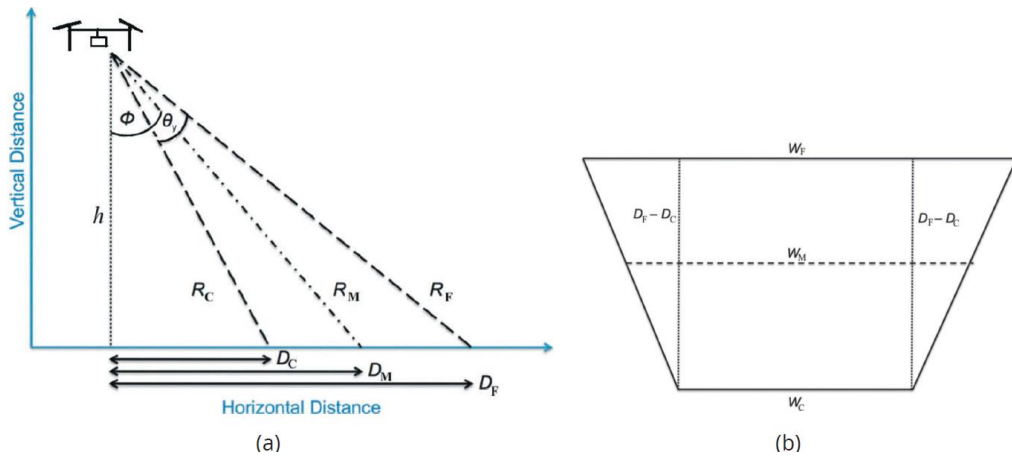


Figura 20. Ejemplo de escala de píxeles deseada

En la Figura 20, literal (a) Cámara en ángulo instalada en el UAV's donde; ϕ es el ángulo de la cámara con respecto al suelo, θ_y es el FOV angular de la cámara a lo largo de eje y, D_C, D_M, D_F , son las distancias más cercana, media y lejana a lo largo del campo de visión de la cámara, h es la altura del UAV. R_C, R_M, R_F las distancias entre el UAV y la parte inferior, media y superior del campo de visión. En el literal (b) El campo de visión proyectado que muestra el área en el suelo cubierta por la cámara resultante de esta configuración. W_C, W_M, W_F son los anchos del campo de visión (FOV) en el eje X de la cámara para las distancias más cercana, media y más lejana a lo largo del eje Y. Entre la cámara y el centro del FOV en el suelo, la relación entre la altura de vuelo del UAV viene dada por la Ecuación (4):

$$h = R_M \cos\left(\phi - \frac{\theta_y}{2}\right). \quad (4)$$

Donde ϕ es el ángulo de la cámara respecto a la tierra, θ_y es el campo de visión angular de la cámara a lo largo de su eje y.

b. Absorción por medio intermitente

Para realizar mediciones precisas de los objetos o animales, es necesario saber sobre las condiciones de observación. Por ende, las propiedades del medio por el que pasa la radiación emitida antes de llegar al detector o cámara tendrán un efecto en la señal recibida, ilustrado en la Figura 21. Donde $I_{\nu 0}$ es la intensidad de la luz infrarroja emitida por la fuente e I_{ν} es la intensidad de luz TIR recibida por el detector o cámara térmica, en una distancia S , a través de un medio con coeficiente de absorción de k_{ν} .

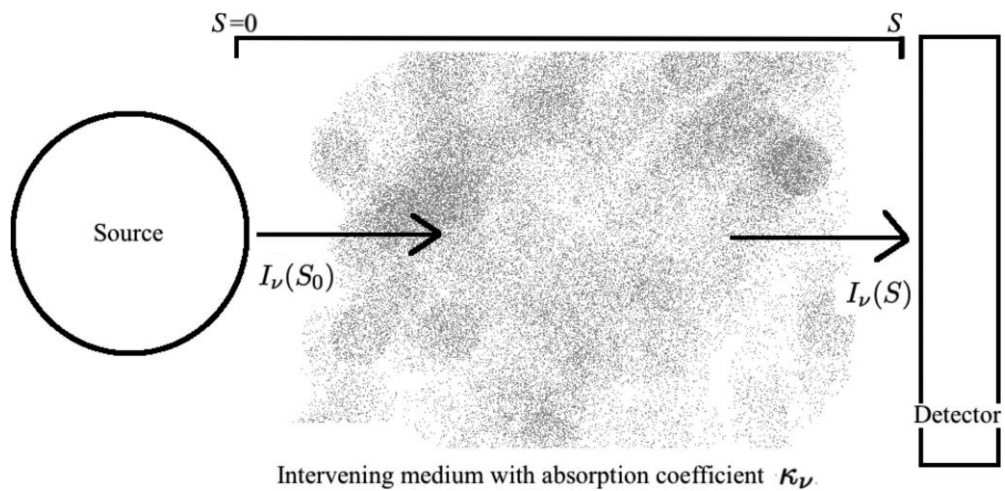


Figura 21. Medio absorbente y medio detector.

La emisión de radiación por moléculas en la atmósfera depende de la presión, la temperatura y la abundancia relativa de diferentes especies de moléculas [28]. Las moléculas presentes en la tropósfera inferior son nitrógeno, oxígeno y agua. La abundancia relativa de estas moléculas depende en gran medida por la humedad; por lo tanto, la cantidad de radiación TIR de una fuente que es absorbida por la atmósfera es en función de la temperatura, la humedad y la presión. Con esta definición, para determinar el efecto de la absorción atmosférica en la medición TIR realizada con cámaras térmicas en UAV's, se puede tomar datos de la base de datos de espectroscopia molecular HITRIAN, para una mezcla de gases representativa de la atmósfera de la tierra a varios valores de humedad, presión y temperatura.

c. El efecto de la niebla

La niebla es un problema importante para volar y observar con cualquier aeronave. Esta niebla se produce cuando el aire ya está saturado de vapor de agua, pero aún se produce la evaporación. Las gotitas de agua suspendidas en la niebla tienen diámetros de 1 y 10 micras y la densidad numérica está entre 1 y 100 cm^{-3} con gotas más grandes tiene una densidad numérica más pequeña por unidad de volumen. A medida que la luz visible pasa a través de la niebla, esas gotitas se dispersan; sin embargo, debido a su mayor longitud de onda, la luz TIR se dispersa menos. El resultado de esto es que se puede observar objetos a mayor distancia con la luz TIR a diferencia de la luz visible.

Un ejemplo del cambio posterior en la temperatura de un objeto que se observaría como resultado de la niebla se muestra en la Figura 22. En el literal a se analiza la fracción de luz de las fuentes transmitida al pasar a través del aire que contiene gotas de niebla suspendidas en diferentes longitudes de onda, $0.6 - 0.9\text{ }\mu\text{m}$ cubre el rango visible de luz, $8 - 14\text{ }\mu\text{m}$ cubierto por la luz térmica infrarroja. En el literal b, se muestra el cambio de temperatura observada de un objeto con la temperatura real $T_s = 25\text{ }^\circ\text{C}$ hasta una distancia de 40 metros no se ve afectado en gran medida por el rango de presiones experimentado dentro de la atmósfera terrestre, la humedad es $H = 100\%$ y la presión $P = 110\text{ Kpa}$.

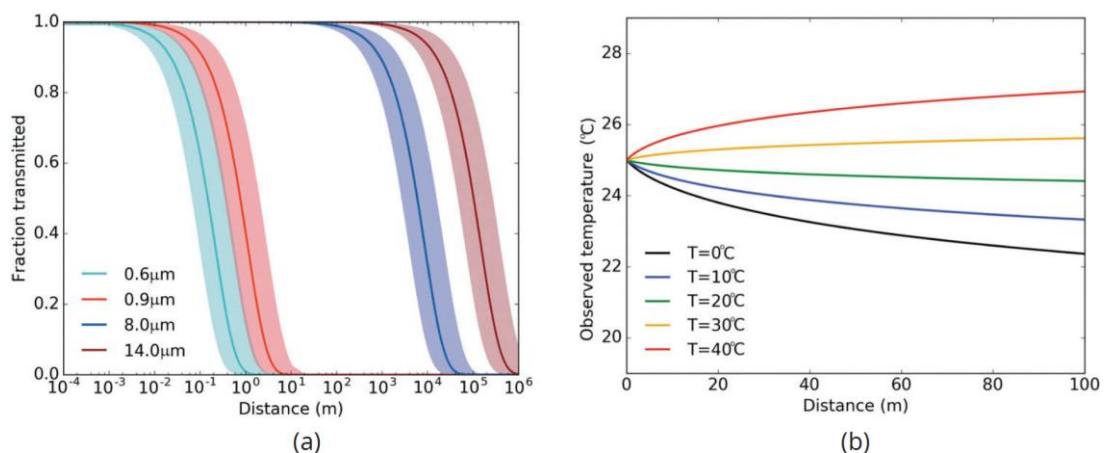


Figura 22. (a) Fracción de la luz de las fuentes transmitida al pasar a través del aire húmedo. (b) Cambio de temperatura observada real como resultado de la absorción y emisión por niebla.

d. Radiación óptica

Las cámaras termográficas pueden formar imágenes de animales o personas que tienen temperaturas más frías que sus fondos, aunque en la mayoría de los casos la pérdida de calor radiante que exhiben los animales supera la pérdida de calor radiante de sus fondos. Los principales factores que contribuyen a la pérdida de calor de un ser vivo en el campo son la conducción, convección, radiación y, en algunos casos, la evaporación de la superficie. De estos mecanismos de pérdida, sólo las pérdidas por radiación son capaces de contribuir de forma directamente a la firma térmica detectada por la cámara [27].

La energía emitida desde la superficie de un ser vivo depende no sólo de la temperatura de su superficie sino también de sus características superficiales, en el caso de los animales el tipo de piel y en los humanos la ropa que modera la emisividad del cuerpo humano, que es un factor que juega un papel importante en la formación de imágenes térmicas [27].

Todos los materiales emiten radiación en un espectro amplio y continuo de longitudes de onda. La cantidad de radiación disponible para realizar el mapeo térmico depende de la cantidad de radiación emitida por los objetos en la escena y la cantidad que realmente llega en el detector [27].

Para aplicaciones de campo, el rango de temperatura de las personas y sus entornos estará cerca de la temperatura ambiente o en un rango desde -22°C y 48°C . En este rango de temperatura, casi toda la radiación emitida en el campo por los objetos naturales (vegetación, agua, rocas, suelo, animales e incluso nieve) se produce en la amplia banda espectral infrarroja de 3 a $100\ \mu\text{m}$ y, por lo tanto, los objetos naturales se comportan como radiadores completos o cuerpos negros en este rango espectral [27].

Las cámaras termográficas están diseñadas para funcionar en dos bandas de longitud de onda específicas en el infrarrojo. La primera banda de imágenes térmicas cubre la banda IR de 3 a $5\ \mu\text{m}$ o de onda media (MWIR) y la segunda banda de imágenes térmicas cubre la banda IR de 8 a $12\ \mu\text{m}$ o de longitud de onda larga (LWIR), como se puede visualizar en la Figura 23 [27].

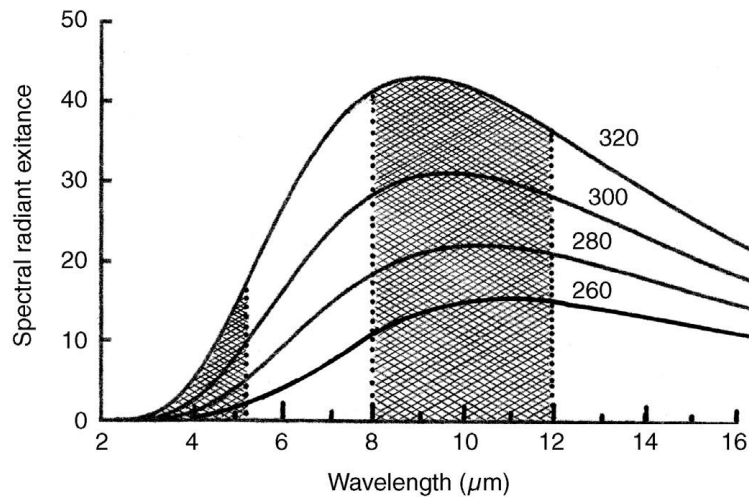


Figura 23. Bandas MWIR y LWIR relativas a las ventanas atmosféricas.

e. *Emisividad*

La emisividad es la capacidad de un cuerpo para emitir energía radiante, para poder capturar esta señal de radiación se utilizan las cámaras térmicas y hacerlo de manera adecuada depende directamente de una colección de parámetros que cambian continuamente, incluyendo la temperatura, el ángulo de visión y una serie de características de superficie (suciedad, condensación, ropa, irregularidades u otras características) [27].

Para poder capturar la emisividad de un cuerpo es necesario calcular el ángulo de visión de la cámara, en la Figura 24 se visualiza la caída de la emisividad a medida que el ángulo de visión se acerca a los 90° [27].

Los buenos absorbentes son objetos con superficies que tienen emisividades superiores a 0.9, mientras que los buenos reflectores tienen emisividades inferiores que 0.2. La mayoría de los mamíferos tienen emisividades superiores a 0.9, lo que los hace buenos absorbentes y reflectores deficientes. En [27] menciona que la piel humana tiene emisividad de 0.98.

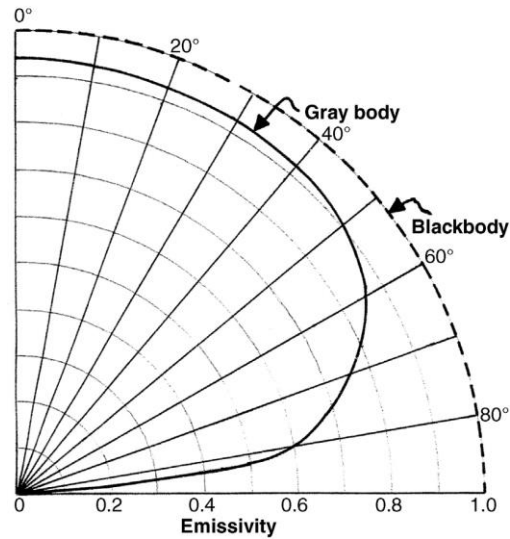


Figura 24. Emisividad y FOV [27].

1.3.7 Distancia de muestreo en el suelo (Ground Sample Distance, GSD)

En telemetría y teledetección el GSD se refiere a la resolución espacial de una imagen o de datos recopilados desde una plataforma aérea o satelital. En términos simples, representa la distancia en el suelo que cada píxel de la imagen representa o cubre, para encontrar esta distancia es necesario calcular el ancho del sensor y para ello se ocupa la Ecuación 5 [29].

$$\text{Ancho del sensor} = \text{Resolución} \times \text{distancia entre píxeles} \quad (5)$$

Para encontrar el GSD es necesario tener distintos parámetros como la longitud del sensor en mm, la altura en metros (m), la distancia focal en mm y ancho de imagen en píxeles (px), para obtener el GSD es necesario utilizar la Ecuación 6 [29].

$$GSD = \frac{\text{Ancho del sensor} * \text{Altura}}{\text{Distancia focal de la cámara} * \text{Ancho de imagen}} \quad (6)$$

1.3.8 Baterías LiPo para UAV

Las baterías de polímero de litio (LiPo) son baterías recargables que utilizan iones de litio, este elemento tiene la responsabilidad de poner en funcionamiento los UAV's, en la Ecuación (7) se describe los parámetros para calcular el tiempo de vuelo [30].

$$t_{flight} = \frac{Q}{\frac{1000}{I_{flying\ load}}} * 60 \quad (7)$$

En donde:

t_{flight} es el tiempo de vuelo en minutos

Q es la capacidad de la batería LiPo en mAh

$I_{flying\ load}$ es la corriente de la batería consumida por los motores, otros equipos que asumen la carga de vuelo seleccionada y el peso del UAV.

La corriente máxima consumida de la batería a plena carga de vuelo se calcula con la Ecuación (8)

$$I_{max\ full\ load} = I_{other} + I_{motor} * N_{motors} \quad (8)$$

En donde:

I_{other} es la corriente máxima en amperios que consume varios equipos del UAV excluyendo sus motores.

I_{motor} es la corriente máxima en amperios que consume un solo motor

N_{motors} es el número de motores

La corriente consumida de la batería con la carga de vuelo seleccionada se determina con la Ecuación (9).

$$I_{flying\ load} = I_{max\ full\ load} * L_{flying} \quad (9)$$

En donde:

$I_{max\ full\ load}$ es la corriente consumida por todo el equipo del helicóptero a plena carga;

L_{flying} es la carga de vuelo (porcentaje), que se basa en el tipo de vuelo y el peso del UAV

1.3.9 Sistemas FPV

Los sistemas FPV (del inglés, First Person View) o visión en primera persona, consiste en una cámara a bordo de un UAV que captan imágenes que son transmitidas en tiempo real en el cual se puede ver las imágenes en un monitor o gafas FPV. Esto le da al piloto la impresión de que está sentado encima del UAV y lo lleva a una experiencia inmersiva similar a la realidad virtual, conocido también como video pilotaje [31].

a. Frecuencia en el que trabaja

La mayoría de UAV actuales se controlan con un radio control que utiliza la banda de frecuencia de 2,4 GHz. Si el objetivo es incorporar un sistema FPV en un UAV comercial se debería trabajar en la banda de frecuencia de 5,8 GHz u otras frecuencias que ofrezcan un ancho de banda considerable que permita transmitir datos a velocidades altas [32].

b. Como funciona un sistema FPV

Es un sistema bastante simple con componentes distintos de diferentes especificaciones y funciones, generalmente todos los sistemas FPV utilizan una cámara FPV, un video transmisor (VTX), un video receptor (VRX), unas gafas o un monitor (FPV) y antenas. La cámara se encuentra en la UAV y lo que está captando se envía en tiempo real desde un VTX en el UAV a el VRX situado en el suelo y finalmente el video se muestra en un monitor o en gafas FPV como se muestra en la Figura 25 [32].

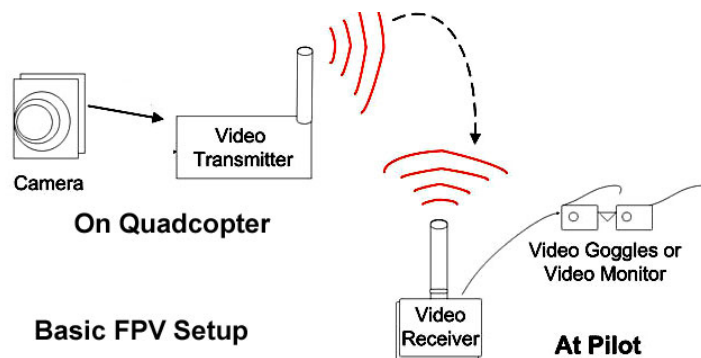


Figura 25. Diagrama básico FPV.

En la Figura 26 se ilustra un esquema con todos sus componentes electrónicos que se utilizan para realizar un sistema FPV.

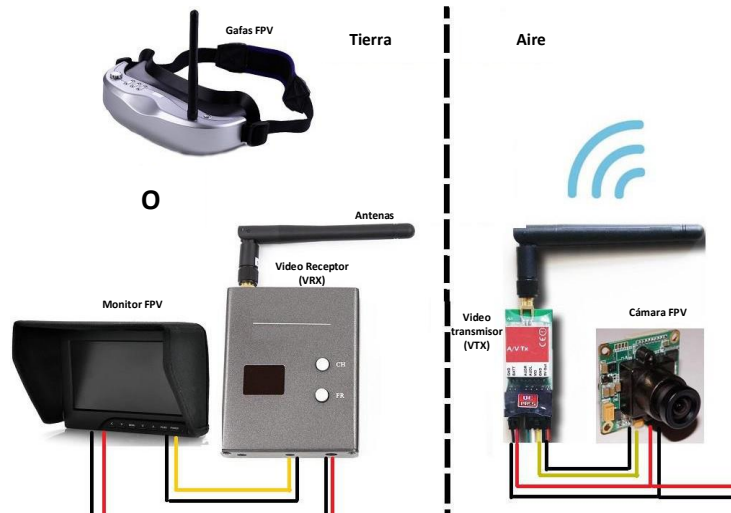


Figura 26. Sistema de transmisión y recepción de datos

c. Cámaras FPV

Las cámaras FPV están diseñadas específicamente para tener una baja latencia, y proporcionar al mismo tiempo el mayor rango dinámico y así proporcionar una gran eficiencia en los entornos cambiantes de luz en la naturaleza. Este dispositivo o capturador de video se puede ver en la Figura 27, debido a su alta eficiencia, las cámaras FPV son generalmente analógicas y no son capaces de captar videos en HD (del inglés, High Definition) o alta definición [32].



Figura 27. Cámara FPV.

d. Video transmisor y receptor VTX/VRX

Un VTX es un video transmisor, esencial para un sistema FPV y tiene la función de enviar la señal de video capturado para la cámara a un receptor VRX, de modo que la

imagen pueda ser visible en un monitor o unas gafas FPV, como se puede observar en la Figura 28. Los VTX pueden usar diferentes bandas de frecuencia, como 1,2 GHz, 2.4 GHz, sin embargo, la frecuencia más usada es la 5.8 GHz porque las antenas son más pequeñas[32].



Figura 28. Sistema de transmisión

En la Tabla 2 se analiza los principales elementos que conforman un sistema de transmisión y recepción FPV.

Tabla 2. Características de los transmisores y receptores de video

Transmisión de video	Cámaras y sensores	Para los sistemas FPV las cámaras deben contar con salida analógica en NTSC O PAL.
	Latencia	La latencia en los sistemas FPV es el tiempo que tarda en llegar la imagen en el video receptor, se mide en milisegundos (ms).
	Alcance	El alcance es la distancia máxima en el que se puede enviar y recibir señal de un UAV's, depende directamente de la potencia del video transmisor.
	Frecuencia de transmisión	5,8 GHz: Alta capacidad de transmitir video de calidad a corta y media distancia. 2.4 GHz: Se utiliza para los radios de control de los UAV's. 1.2 GHz y 1.3 GHz: Útiles para sistemas FPV que requieren mayor alcance.
Recepción de video	Antenas	Omnidireccional: Emiten/captan señales en todas las direcciones Direccional: Emiten/captan la señal en una sola dirección [33].
	Monitores	Permiten visualizar la señal de video en tiempo real capturado por el UAV.
	Registro	Algunos monitores o gafas FPV permiten grabar la señal de video en una tarjeta de memoria, lo cual es importante para realizar análisis [33].

Para el desarrollo del proyecto se ha determinado utilizar el sistema FPV, ya que la cámara térmica FLIR Tau 2 emite señal analógica en formato NTSC y PAL, así como también los sistemas FPV trabajan con una señal 5.8 GHZ el cual es más rápida en la transmisión de datos a corta y media distancia.

1.3.10 Aprendizaje automático

El aprendizaje automático conocido por su nombre en inglés, Machine Learning, representa una división de la Inteligencia Artificial enfocada en crear metodologías que habilitan a las computadoras para aprender de manera autónoma. Se puede dividir en varios tipos de aprendizaje automático [34].

a. Aprendizaje supervisado

Este tipo de aprendizaje implica entrenar un modelo con un conjunto de datos que incluye ejemplos de etiquetado realizadas por un humano.

b. Aprendizaje no supervisado

En este caso, el modelo se entrena con datos que no tienen etiquetas o clasificaciones previas. El objetivo principal es permitir que el modelo encuentre patrones, estructuras o características intrínsecas en los datos por sí mismo [34].

c. Aprendizaje por refuerzo

Este tipo de aprendizaje se basa en un sistema de recompensas y castigos. El modelo, llamado agente, toma decisiones dentro de un entorno y recibe retroalimentación en forma de recompensas o penalizaciones según las acciones que realiza [34].

Para la detección de personas se utiliza el aprendizaje supervisado ya que al algoritmo se le presenta las muestras de entrenamiento previamente etiquetadas por un humano.

1.3.11 Detección de personas

La detección de personas es un concepto que cae dentro del campo de la visión artificial, aunque frecuentemente se utiliza el poder de las redes neuronales para lograr esta tarea. El objetivo principal es desarrollar algoritmos y sistemas que puedan reconocer la presencia humana, distinguirla de otros objetos y, en muchos casos, incluso identificar características específicas de las personas, como su ubicación, postura, gestos o acciones [34].

a. **Visión Artificial en la Detección de Personas:**

- **Visión Artificial:** Es uno de los campos de la inteligencia artificial y la ingeniería que se enfoca de emular la capacidad humana de percibir, interpretar y comprender la visión a través de un sistema de computación. Este campo implica el procesamiento de imágenes digitales para obtener información con la finalidad de identificar patrones, reconocer objetos, realizar seguimiento de movimientos y tomar decisiones basadas en la extracción de la información visual [35].

La estructura de la visión artificial involucra varias etapas y procesos para interpretar y comprender la información visual, en la Figura 29 se puede visualizar el proceso de entrada, procesamiento y salida [35].

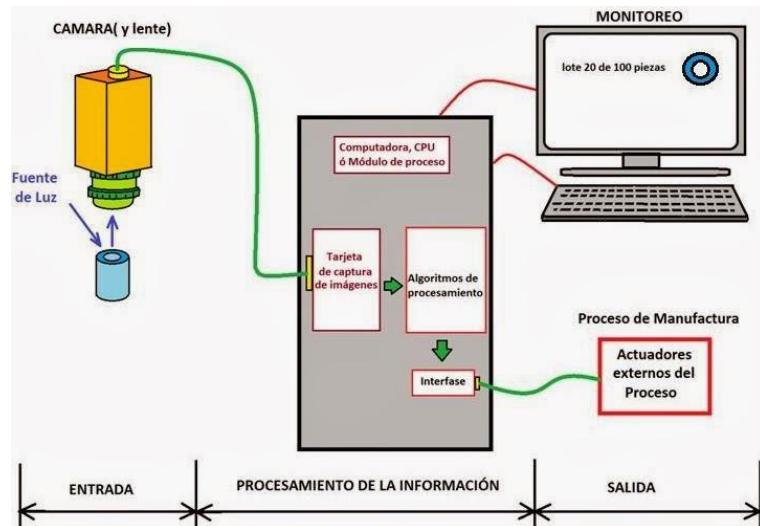


Figura 29. Estructura de un sistema de visión artificial [35]

Los procesos que constan de un sistema de visión artificial son:

- Captura de la imagen
- Conversión en formato digital
- Procesamiento mediante una computadora
- Obtención de resultados

- **Componentes de un sistema de visión artificial:** Los elementos que abarcan un sistema de visión artificial son los sensores de imagen, procesamiento de visión y comunicación [35].

Iluminación: Es una parte importante en el campo de la visión artificial cuando la captura de las imágenes es por cámaras RGB y no es un requisito cuando la captura de imágenes es con cámaras térmicas

Lentes: Son los encargados de definir la calidad de la imagen, algunas cámaras están compuestas por dos tipos de lentes, las cámaras de lentes intercambiables y de lentes fijos.

Sensor de imagen: Es un componente que en base a su tamaño dicta la calidad de la fotografía por ende los detalles a considerar dentro del tratamiento de la imagen son mayores.

Procesamiento de visión: Aquí se extrae la información captada de una imagen digital, mediante el uso de softwares.

- **Etapas de un sistema de visión artificial:** Un sistema consta de diferentes etapas que están relacionadas en hardware y software, en la Figura 30 se ilustra estas etapas.

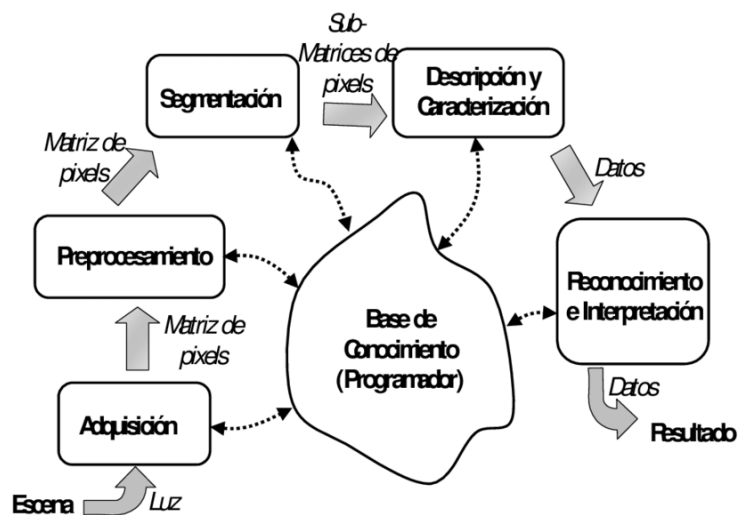


Figura 30. Etapas de un sistema de visión artificial [34].

Las etapas de un sistema de visión artificial se presentan a continuación.

Captura de imagen digital: Se requiere una imagen nítida y de alta claridad para asegurar un procesamiento excelente.

Digitalización de la imagen: Después de la captura, la imagen se dirige al preprocesamiento para mejorar su calidad.

Segmentación: Esta etapa destaca por dividir la imagen en fragmentos o secciones, siendo crucial para el reconocimiento de objetos.

Resultados de la segmentación: Aquí se incluyen los puntos (características internas) y las fronteras (propiedades externas) de las regiones identificadas.

Parametrización: Se extraen datos cuantitativos y rasgos distintivos para diferenciar objetos en la imagen.

Reconocimiento e interpretación: Etiquetado y clasificación de los objetos identificados en la imagen para su análisis.

1.3.12 Red convolucional YOLO

YOLO (por sus siglas en inglés. You Only Look Once) es un sistema que utiliza una red neuronal convolucional para la detección de objetos en tiempo real. Esta red divide la imagen en regiones y predice múltiples cajas delimitadoras (en inglés, bounding box) así como también la probabilidad de detección de las clases de entrenamiento por cada caja delimitadora. En la Figura 31 se muestra una ejecución del sistema sobre una imagen de entrada [34].

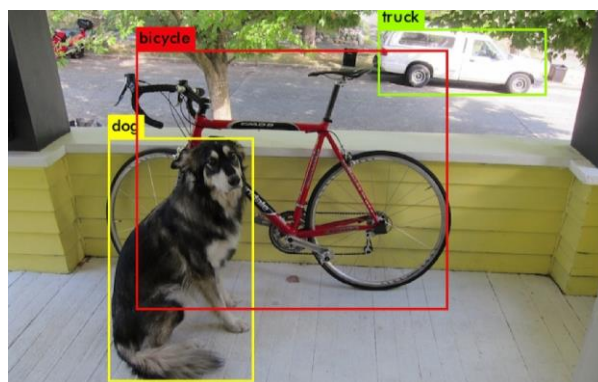


Figura 31. Detección de objetos con YOLO

a. YOLOv8

YOLOv8 representa la iteración más reciente en la línea de detectores de objetos en tiempo real YOLO, ofreciendo un desempeño vanguardista en términos de velocidad y precisión. Con mejoras significativas basadas en sus predecesores, esta versión presenta nuevas funcionalidades y optimizaciones, consolidándose como una elección destacada para diversas tareas de detección de objetos en una amplia gama de aplicaciones [34].

En enero de 2023, Ultralytics lanzó YOLOv8 como una mejora respecto a su predecesor, YOLOv5, convirtiéndolo en un modelo altamente preciso en el conjunto de datos COCO. El modelo mediano logra un impresionante 50,2% de mAP cuando se evalúa con los datos COCO, como se puede visualizar en la Figura 32. Además, introduce una interfaz gráfica amigable mediante un paquete PIP bien organizado como novedad destacada [34].

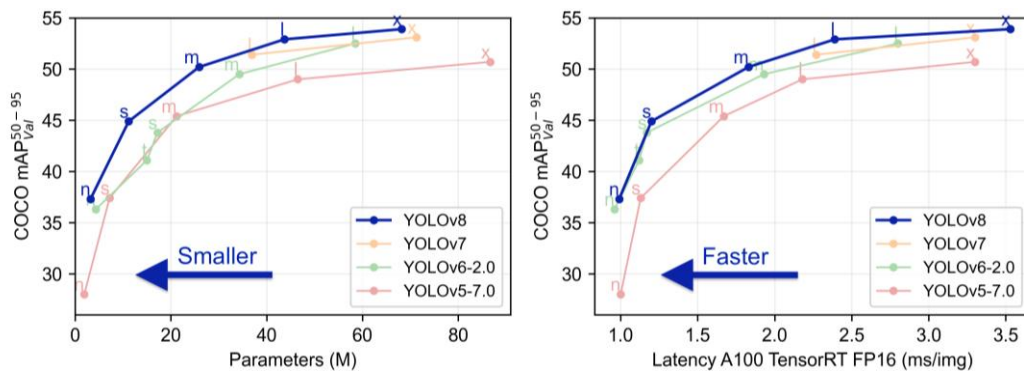


Figura 32. mAP COCO vs número de Parámetros (pesos y sesgos) – YOLOv8 [36]

mAP (mean Average Precision): Es una medida utilizada para evaluar la precisión de los modelos de detección de objetos y representa un promedio de las precisiones obtenidas a diferentes niveles de recuperación (capacidad de un modelo para encontrar todos los elementos relevantes). Cuanto mayor sea el valor de mAP, mejor será la precisión del modelo en la detección de objetos [36].

Parámetros (M) en millones: Se refiere a la cantidad de parámetros utilizados en el modelo neuronal, los modelos más complejos generalmente tienen más parámetros, lo que genera un rendimiento mejorado, pero también a una mayor complejidad computacional [36].

1.3.13 Herramientas para el procesamiento de imágenes

Para obtener la detección de las personas en tiempo real es necesario hacer uso del procesamiento de imágenes. Para este tipo de tareas existe una biblioteca muy utilizada para el procesamiento de imágenes como OpenCV.

a. OpenCV

OpenCV, acrónimo de Open Source Computer Vision, representa una librería de código abierto destinada a la visión por computadora y el aprendizaje automático, inicialmente concebida por Intel en 1999, esta herramienta ha evolucionado gracias a una comunidad global de desarrolladores, quienes han contribuido a su constante desarrollo y mantenimiento a lo largo del tiempo [34].

Algunas de las funcionalidades que ofrece OpenCV actualmente incluyen:

Adquisición y procesamiento de imágenes en tiempo real a través de dispositivos como cámaras y luego procesarlos en tiempo real mediante técnicas de procesamiento de imágenes y video [34].

Identificar y clasificar características específicas del rostro humano (como los ojos, la nariz, la boca y la forma de la cara)

Permite corregir las distorsiones de la imagen y ajustar los parámetros intrínsecos y extrínsecos de la cámara con el objetivo de mejorar la precisión del procesamiento de imágenes.

b. Python

Python ha jugado un papel fundamental en el ámbito de la detección de objetos gracias a bibliotecas poderosas como TensorFlow, Keras y especialmente OpenCV. Estas herramientas ofrecen funcionalidades robustas para entrenar modelos, implementar algoritmos de detección y realizar análisis de imágenes y vídeos de manera eficiente. La sintaxis simple y legible de Python junto con su extensa comunidad de desarrolladores han contribuido a que sea un lenguaje popular en el campo de la visión por computadora y la detección de objetos [34].

c. Pythorch

En octubre de 2016, se presentó PyTorch como una biblioteca diseñada para modelos de Machine Learning, desarrollada por Adam Paszke, Sam Gross, Soumith Chintala y Gregory Chanan, quienes forman parte del Laboratorio de Investigación de Inteligencia Artificial de Facebook (FAIR). Esta librería, escrita en Python, C++ y CUDA, ofrece dos características principales a un nivel superior [34].

Cálculo con tensores multidimensionales de manera acelerada, similar a los proporcionados por la librería NumPy, gracias al soporte para ejecución en GPU.

Construcción de redes neuronales profundas sobre un sistema que registra las operaciones matemáticas realizadas durante la ejecución del modelo. PyTorch utiliza una "cinta" para este fin, la cual registra todas las operaciones en un tensor durante el entrenamiento. Utilizando la regla de la cadena, el sistema de auto diferenciación puede calcular automáticamente derivadas parciales y gradientes necesarios para ajustar los pesos de la red neuronal durante el proceso de entrenamiento [34].

1.4 Objetivos

1.4.1 Objetivo general

Desarrollar un sistema de localización de personas en catástrofes utilizando UAV's.

1.4.2 Objetivos específicos







- Analizar los sistemas de detección de personas utilizando UAV's.
- Diseñar un sistema de localización de personas en diferentes catástrofes.
- Implementar el sistema de localización de personas utilizando UAV's.




CAPÍTULO II.- METODOLOGÍA

2.1 Materiales

En el transcurso de este proyecto de investigación, se utilizaron diversos dispositivos y materiales con el fin de establecer un sistema de localización de personas en situaciones de emergencia. En la Tabla 3 se detallan todos los materiales empleados para este propósito





Tabla 3. Materiales

Materiales	Descripción	Imagen
Cámara térmica de la empresa Teledyne Flir modelo Tau 2	Utilizada para recolectar imágenes térmicas de un objeto o entorno.	
UAV	Empleado como soporte para la cámara térmica con el propósito de sobrevolar y capturar imágenes térmicas.	
Cables	Para realizar conexiones eléctricas.	
Video transmisor (VTX 801)	Utilizado para transmitir en tiempo real el video emitido por la cámara térmica.	
Video receptor (VRX)	Usado para recibir en tiempo real el video emitido por el transmisor.	
Antenas para el UAV	Permiten la comunicación entre transmisor y receptor.	

Materiales	Descripción	Imagen
Módulo VPC para Flir Tau 2	Utilizado para la alimentación y el control de la cámara térmica.	
Impresiones 3D	Utilizado como soportes para del UAV y para la protección de la cámara.	
Computador	Utilizado para procesar información capturada por el UAV.	

En la Tabla 4 se describen los softwares para la recolección de información como es OBS y también para el entrenamiento de la red neuronal como son Google colab y Python.

Tabla 4. Softwares

Softwares	Descripción	Imagen
OBS	Utilizado para capturar el video y grabar en tiempo real	
Google Colab	Permitió el entrenamiento de la red neuronal.	
Python	Empleado para el desarrollo de la interfaz gráfica para la localización de personas.	
Mission Planner	Este software de código abierto fue empleado para desarrollar vuelos programados mediante trayectorias, utilizando un mapa para ubicar los puntos de altitud y latitud.	
Makesense.ai	Este software en línea permitió etiquetar todas las imágenes para el entrenamiento de la red neuronal	

2.2 Métodos

2.2.1 Modalidad de la investigación

a. Investigación Aplicada

Durante este proyecto de investigación, se emplearon una variedad de dispositivos y materiales para configurar un sistema que permita localizar personas en diferentes tipos de catástrofes o situaciones de emergencia, haciendo uso de tecnologías como la termografía infrarroja, cámaras RGB, dispositivos GPS y los UAV's con el objetivo de localizar e identificar individuos en condiciones críticas.

El proyecto se desarrolló en tres fases: en la primera etapa se llevó a cabo el análisis, la configuración y la conexión de la cámara térmica Teledyne Flir modelo Tau 2 en un vehículo aéreo no tripulado (UAV); en la segunda fase se diseñó un sistema de monitorización para establecer una plataforma de comunicación entre el UAV y la estación terrestre, con el fin de realizar el análisis de datos; y finalmente, se desarrolló una interfaz gráfica utilizando Python que permite la localización y detección de personas.

Para el montaje de la cámara térmica, se utilizaron diversos materiales como cables, un video transmisor (VTX 801), un video receptor (VRX), antenas, la cámara térmica Flir Tau 2 y un módulo VPC para Flir Tau 2. Respecto a la recolección de datos, se montó la cámara Flir Tau 2 en el UAV para la captura de imágenes, utilizando impresiones en 3D para asegurar y montar la cámara de manera precisa y protegerla adecuadamente.

Una vez obtenidas las imágenes, se llevó a cabo el entrenamiento de una red neuronal en Google Colab. Posteriormente, se utilizó Python para desarrollar un sistema de localización que crea una interfaz gráfica, procesando la señal de video para detectar y ubicar a las personas.

b. Investigación bibliográfica

La aplicación de una investigación exhaustiva basada en fuentes bibliográficas ha sido fundamental para recopilar información sobre sistemas de localización de personas en situaciones de catástrofe. Este proceso incluyó una minuciosa exploración en artículos científicos, tesis, libros, revistas, informes y trabajos de investigación. Los recursos empleados provienen de bases de datos reconocidas por su fiabilidad, como Google Académico, eLibro, IEE Xplore y Dyna. Esta recopilación de datos sólidos y respaldados por fuentes confiables ha sido crucial para respaldar de manera sólida el desarrollo del proyecto.

c. Investigación de campo

Se llevó a cabo una investigación de campo con el propósito de recabar información relevante, como el perfil del terreno, la velocidad del viento y los momentos de mayor intensidad ventosa. Esto se realizó con la finalidad de analizar meticulosamente el área y determinar la idoneidad para realizar vuelos con el UAV, llevando a cabo diversas pruebas y capturando imágenes térmicas.

d. Investigación experimental

Esta investigación adoptó la modalidad experimental como enfoque metodológico principal, fundamentado tanto en la construcción como en la validación del sistema de localización de personas en situaciones de catástrofe mediante el uso de UAV's. Durante la fase experimental, se llevó a cabo el sobrevuelo del UAV, generando aproximadamente 1000 imágenes térmicas, distribuidas equitativamente entre el día y la noche, las cuales fueron esenciales para el entrenamiento de la red neuronal. Este proceso reveló de manera efectiva la presencia de un contraste térmico en las imágenes, donde cada color correspondía a una temperatura específica, validando así la capacidad del sistema para identificar la radiación emitida por las personas. Además, se experimentaron variables como la altura de vuelo, la velocidad del UAV y las rutas de vuelo programadas, con el objetivo de determinar los parámetros óptimos que permitieran la detección eficiente de personas en condiciones de catástrofe.

2.2.2 Recolección de información

Para recopilar la información necesaria, se empleó la cámara Flir Tau 2 instalada en el UAV. Se realizaron vuelos para capturar video, considerando factores como la ubicación, la hora, la posición de las personas y la altitud del UAV. En el Anexo A se encuentra un registro detallado que describe el cumplimiento de estas características específicas.

Luego, se procesaron estas grabaciones para obtener las imágenes y organizarlas en una carpeta denominada “data”, para entrenar la red neuronal y lograr una alta eficiencia al momento de realizar la localización e identificación de personas.

2.2.3 Procesamiento y análisis de datos

Posterior a la recolección de la información para el desarrollo del proyecto, es necesario crear una base de datos con las imágenes recolectadas, con la finalidad de entrenar una red neuronal para la detección de personas, en Tabla 5 se detalla el procedimiento que se realizó para el procesamiento de las imágenes recolectadas mediante los sobrevuelos.

Tabla 5. Procesamiento y análisis de datos

Fases	Procedimiento
Crear una carpeta de datos	Para crear la base de datos se siguieron los siguientes pasos. <ul style="list-style-type: none">• Se creó una carpeta denominada “data”, y dentro de ella dos carpetas denominadas images (imágenes) y label (etiquetas).• En la carpeta de “images” y “label” se creó dos subcarpetas denominadas train (entrenamiento) y val (validación).• El almacenamiento de todas las imágenes obtenidas de los sobrevuelos con el UAV se guardó en la carpeta “images/train” y “images/val”• El almacenamiento de las imágenes etiquetadas señalando el objeto a detectar se guardó en la carpeta de etiquetas “label”.• Para etiquetar las imágenes se utilizó makesense.ai• Es esencial separar un porcentaje de imágenes para la validación de la red neuronal en la carpeta de val.
Entrenamiento	Se ha utilizado YOLOv8l para el entrenamiento de la red neuronal, este entrenamiento se realizó con 80 y 100 épocas, con la finalidad de obtener mejor precisión para la detección de personas.
Inferencia	En este apartado se ejecutó el modelo entrenado, mediante la captura del video desde 1 UAV y la detección en tiempo real utilizando el software Python.

CAPÍTULO III.- RESULTADOS Y DISCUSIÓN

3.1 Desarrollo de la propuesta

El presente proyecto se desarrolló con el propósito de abordar los desafíos que surgen cuando los seres humanos se encuentran en situaciones de peligro, como terremotos, inundaciones, deslizamientos de tierra y en nevados. En estos escenarios, la falta de tecnología dificulta la localización eficiente de las personas especialmente en la oscuridad. Es por lo que adecuar los UAV's con sistemas GPS, cámaras térmicas y teledetección es una solución eficiente por la geografía que genera la cordillera de los Andes en Ecuador.

La implementación de estas tecnologías tiene como objetivo prioritario localizar a las personas en áreas específicas, lo que permitirá al equipo de socorro acceder de manera inmediata a las operaciones de rescate correspondientes.

3.1.1 Identificación y análisis de sistemas de localización y detección de personas en situaciones de catástrofe.

Los sistemas de localización de personas utilizando UAV's, en estos últimos años han experimentado un crecimiento exponencial debido a los avances tecnológicos en diferentes tipos de sensores, procesamiento de datos por computador, inteligencia artificial y la propia tecnología de los UAV's. Existen varios tipos de sistemas para la localización de personas mediante UAV's, cada uno de ellos están diseñados para cumplir con aplicaciones específicas. A continuación, se analizan los más comunes utilizados para aplicaciones de detección y localización de personas utilizando UAV's:

a. Sistemas de detección por cámaras RGB

Estos sistemas pueden equiparse con algoritmos de visión por computadora para detectar personas en imágenes y video capturados por el UAV's, como se puede observar en la Figura 33 [37]. Pueden detectar personas tanto en condiciones de luz diurna como en la oscuridad, siempre que se disponga de iluminación idónea. Los

algoritmos de detección y localización son ejemplos de técnicas utilizadas en sistemas RGB, para ello se ocupa de softwares como Python, YOLO y Google Colab [26].

Para que el sistema sea eficiente los UAV's deben estar equipados con cámaras de alta resolución, que puedan capturar imágenes y video de buena calidad desde el aire. La calidad de estas cámaras es esencial para obtener imágenes nítidas y detalladas de personas, esencialmente este factor depende de varias características como: el tiempo de exposición, paso de muestreo, tamaño de viruta, ruido de sensores y la sensibilidad de la luz ISO que establece la Organización Internacional de la estandarización (ISO, por sus siglas en inglés) como un valor numérico [26].

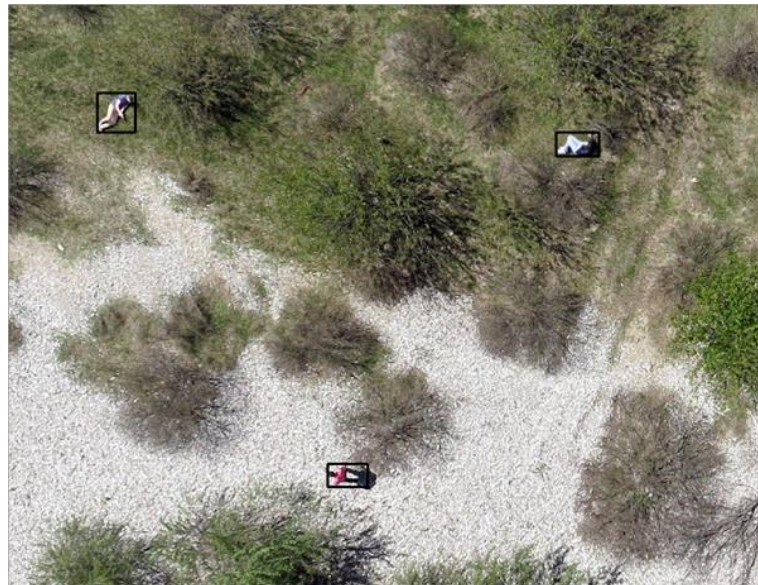


Figura 33. Detección de personas [38]

b. Sistema de posicionamiento y navegación

Los UAV's utilizan sistemas de posicionamiento global (GPS) y sensores de navegación para volar de manera precisa y mantener una posición estable en el aire. Estos sistemas son cruciales para dirigir el UAV hacia el área de interés y mantenerlo en la ubicación adecuada para la identificación de la persona desaparecida en un determinado lugar. Para realizar rutas planificadas se utiliza el software Misión Planner (Figura 34) el cual es diseñado exclusivamente para los UAV's [39].

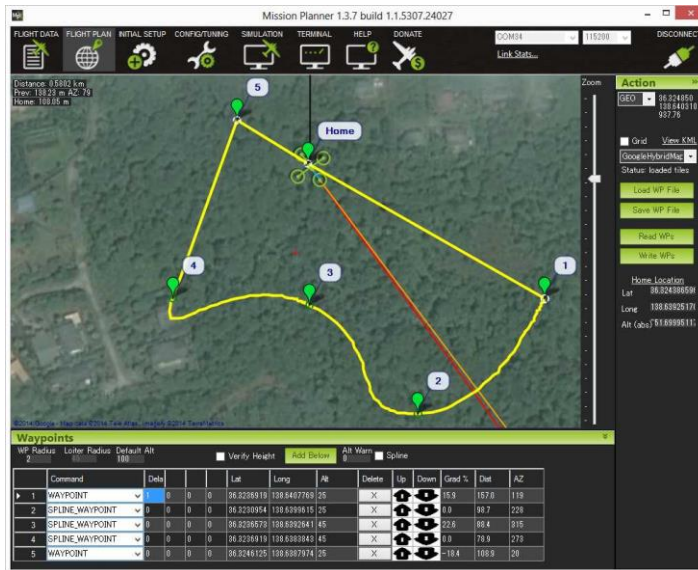


Figura 34. Software Mission Planner [39]

c. *Sistemas de detección térmica*

Los UAV's equipados con cámaras térmicas pueden detectar personas según las diferencias de temperatura entre el cuerpo humano y el entorno, como se puede visualizar en una escala de grises en la Figura 35. Estos sistemas son especialmente útiles para la detección en la oscuridad, en condiciones de niebla o cuando los individuos estén ocultos a simple vista. Pueden utilizarse en búsquedas y rescate, así como en aplicaciones de seguridad [22].



Figura 35. Sistema de detección térmica.

Las cámaras térmicas como la Tau 2 de la empresa Teledyne FLIR, permite detectar la radiación infrarroja que emiten los cuerpos a un intervalo de 0.7 – 1000 μm como

se puede observar en el espectro electromagnético en la Figura 9, siempre y cuando los cuerpos tengan temperatura mayor a 0° Kelvin o -273° Celcius, lo que facilita la detección de personas en la oscuridad o en condiciones de visibilidad reducida. Esto es especialmente útil en operaciones de búsqueda y rescate diurnos y nocturnos o en entornos adversos como campos con maleza [23].

d. Sistemas de detección por LiDAR

LiDAR es una tecnología que utiliza pulsos láser sobre una superficie, mide el tiempo de viaje con el fin de medir distancias y crear mapas en 3D del entorno como se muestra en la Figura 36. Los UAV's equipados con sensores LiDAR pueden detectar personas mediante la identificación de objetos y estructuras tridimensionales en el terreno [40]. Esto es útil en aplicaciones donde la detección precisa y la caracterización del terreno son esenciales.

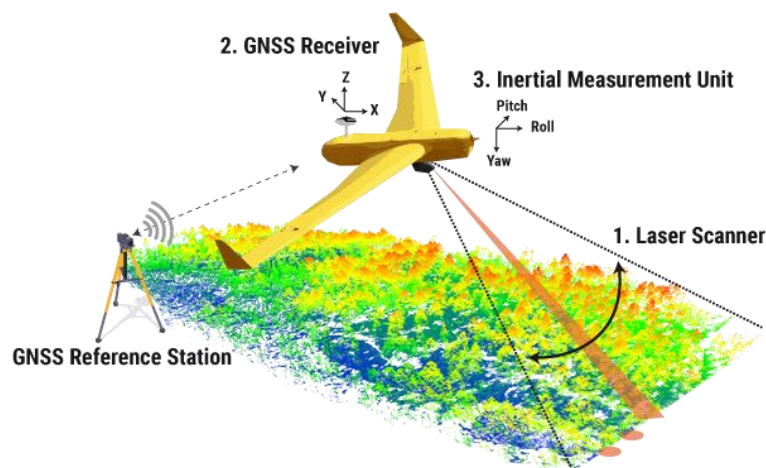


Figura 36. Funcionamiento del sistema LiDAR

e. Sistemas de detección de radio frecuencia (RF)

Los UAV's equipados con sensores de RF pueden detectar la presencia de personas a través de señales de radio. Esto puede utilizarse en la detección de dispositivos de comunicación como teléfonos móviles u otros dispositivos inalámbricos [41]. El detector Awareness de RF, es utilizado para realizar barridos electrónicos veloces y efectivos, puede detectar cámaras inalámbricas de video, dispositivos Wifi/Bluetooth, radios de modulación analógica (AM, FM, PM), teléfonos móviles y transmisores de radio con modulación de banda ancha de hasta 10 MHz [42], lo cual es útil en

situaciones en el que el rescate de personas es imposible mediante cámaras RGB y térmicas, en la Figura 37 se muestra un sistema RF con la utilización de UAV para situaciones de catástrofe.

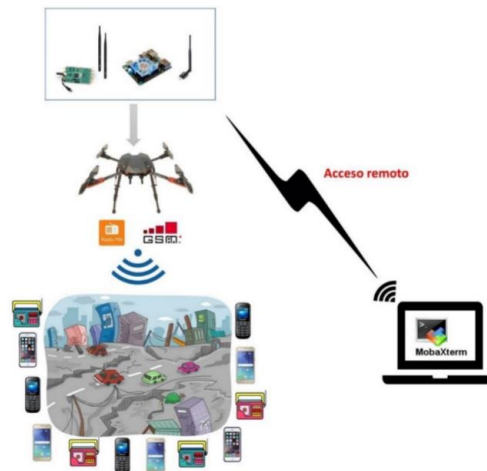


Figura 37. Sistemas de Radio Frecuencia mediante UAV's [41]

f. Sistemas de detección de audio

Algunos UAV's pueden estar equipados con micrófonos para detectar sonidos asociados con la presencia humana, como voces, gritos o ruidos de movimiento en una determinada área. Estos sistemas pueden ser útiles en aplicaciones de búsqueda y rescate, los sistemas de transmisión como el VTX 801 (Figura 38) incorpora un micrófono en su transmisor de forma que no se necesita un micrófono externo.

g. Sistemas de comunicación y transmisión de datos

Los UAV's deben estar equipados con sistemas de comunicación confiables para transmitir datos en tiempo real a los operadores o centros de control, esto es crucial para la toma de decisiones en tiempo real. Las tecnologías que se utiliza son los transmisores de video analógicos como se muestra en la Figura 38, el cual se trasmite mediante ondas de frecuencia a 5.8 GHz, en la que detallan características cruciales de esta tecnología [26].



Figura 38. Video transmisor VTX 801 con micrófono

h. Algoritmos de detección de personas

Los algoritmos de visión por computadora y aprendizaje profundo se utilizan para detectar personas en imágenes y video capturados por el UAV's. Estos algoritmos pueden detectar características faciales, formas corporales y movimientos característicos para identificar personas como se muestra en la Figura 39. Este sistema de detección es bastante importante, debido a que la computadora procesa la información emitida por el UAV y la compara para obtener un resultado [37]. La investigación en esta área se enfoca en mejorar los modelos de aprendizaje automático y la capacidad de adaptación a diversas situaciones.



Figura 39. Algoritmos de detección de personas

3.1.2 Selección del sistema de localización

Para un análisis más profundo sobre los sistemas de detección y localización de personas, en la Tabla 6 se realiza una matriz de enfrentamiento con los criterios como; la detección en la oscuridad, identificación de detalles, detección de temperaturas, costo, identificación en 3D y la compatibilidad con los UAV's, esto con la finalidad de determinar la importancia de cada factor mediante una ponderación.

Tabla 6. Matriz de enfrentamiento

Factores	Detección en la oscuridad	Identificación de detalles	Detección de temperatura	Costo	Identificación en 3D	Compatibilidad con el UAV	Total	Ponderación
Detección en la oscuridad	0	1	1	0	1	1	3	25%
Identificación de detalles	0	0	0	0	1	1	1	8%
Detección de temperatura	0	1	0	1	0	1	2	17%
Costo	1	1	0	0	1	0	3	25%
Identificación en 3D	0	0	1	0	0	0	1	8%
Compatibilidad con el UAV	0	0	0	1	1	0	2	17%
							12	100%

Una vez obtenido la ponderación de los factores, se realiza una escala de calificación para cuantificar las alternativas, para ello se emplea la Tabla 7.

Tabla 7. Escala de calificación

Escala de calificación de análisis de factores	
Calificación	Valor numérico
Excelente	5
Bueno	4
Regular	3
Deficiente	2
Malo	1
Insuficiente	0

En la Tabla 8 se calificó los sistemas de detección visual, térmico y LiDAR, debido a que presentan características que se relacionan con los factores analizados, como detección en la oscuridad, identificación de detalles y la compatibilidad con los UAV.

Tabla 8. Evaluación de alternativas por el método de factores.

Factores	Ponderación	Sistema visual		Sistema térmico		Sistema LiDAR	
		Calificación	Peso	Calificación	Peso	Calificación	Peso
Detección en la oscuridad	25%	2	0,5	5	1,25	4	1
Identificación de detalles	8%	5	0,4	2	0,17	1	0,08
Detección de temperatura	17%	0	0,0	5	0,83	0	0
Costo	25%	3	0,8	2	0,5	1	0,25
Identificación en 3D	8%	0	0,0	0	0	5	0,42
Compatibilidad con el UAV's	17%	4	0,7	4	0,67	2	0,33
Total		2,3		3,4		2,1	
Datos normalizados		17,0%		24,8%		15,2%	

Mediante el análisis de las alternativas de los sistemas de detección se obtuvo un porcentaje del 24,8% para el sistema de detección térmico y 17,0% para el sistema de detección visual. El sistema térmico es esencial para el desarrollo del proyecto, ya que en situaciones de catástrofe la visibilidad es deficiente, y las cámaras térmicas no dependen de la luz visible como las cámaras RGB. Por tal motivo se determinó utilizar la tecnología infrarroja como un sistema principal para la localización de personas.

Además de ello, una vez localizada a la persona en situación de peligro, es necesario saber su ubicación, para lo cual se ha seleccionado el sistema de posicionamiento GPS, el mismo que entrega información de longitud y latitud del lugar donde vuela el UAV's mediante tecnología OSD (On Screen Display), que permite visualizar mediante la cámara RGB.

Con la ayuda de estos sistemas, se puede focalizar a los rescatistas en una determinada área, en donde se encuentra las personas en situación de peligro para que el rescate sea eficiente.

3.1.3 Requerimiento del sistema

Se debe considerar varios factores claves para implementar un sistema de localización de personas utilizando UAV's, cuyos requerimientos se denotan a continuación.

a. Cámara térmica

La cámara térmica debe tener la capacidad de detectar pequeños cambios de temperatura y la precisión en la localización y detección, esto garantiza una identificación precisa de personas o elementos relevantes en entornos desafiantes.

b. Cámara RGB

La conexión directa de la cámara RGB a la placa controladora de vuelo es fundamental para la extracción precisa de datos de longitud y latitud mediante tecnología OSD. Además, permite el acceso a información vital, como la duración de la batería, el número de satélites conectados al UAV y detalles sobre posibles errores que puedan afectar el rendimiento del UAV en general. La información de los datos de localización es obtenida mediante el GPS integrado al UAV, esta integración garantiza una recopilación eficiente y en tiempo real de datos esenciales para el funcionamiento seguro y preciso del sistema de localización.

c. GPS

Es necesario contar con un sistema de posicionamiento global GPS para obtener datos para la localización de personas en situaciones de catástrofes.

d. Altura del UAV

La altura de vuelo determina la capacidad para detectar y localizar personas, esta debe ocupar un área mínima (en píxeles) dentro del campo de visión de la cámara, para calcular la altura es necesario determinar el GSD (del inglés, Ground Sample Distance), que es la distancia en centímetros que cubre cada píxel.

e. Tiempo de vuelo

La capacidad de la batería es un requerimiento importante para el vuelo del UAV, debido a que determina la cantidad de energía disponible para sostener el vehículo en vuelo y llevar a cabo las operaciones programadas.

f. Capacidad de transferir datos en tiempo real

La capacidad de transmitir imágenes y datos térmicos en tiempo real es esencial para la toma de decisiones inmediatas, en la Tabla 9 se detallan las actividades y los requerimientos necesarios para la transmisión de datos en tiempo real del sistema térmico de localización.

Tabla 9. Requerimientos del sistema

Actividad	Requerimiento
Capturar el video de la cámara térmica	Cámara térmica, cables
Enviar el video en tiempo real desde el UAV	Transmisor de video
Adquirir el video en tiempo real en la estación terrestre	Receptor de video
Procesar y visualizar el video capturado	Software Python

g. Interfaz de usuario

Es necesario desarrollar una interfaz intuitiva en el que los operadores puedan utilizarlo eficazmente, en esta interfaz se debe visualizar las detecciones de las personas y parámetros de vuelo como la longitud y latitud del UAV para la localización.

h. Cumplimiento legal y ético

Es importante asegurarse de que el sistema cumpla con los estándares ecuatorianos que dicta la Dirección General de Aviación Civil, tal como se muestra en el Anexo B.

3.1.4 Esquema general del sistema

El esquema general del sistema de localización se compone de tres etapas fundamentales, mediante las cuales se logra capturar, receptar, visualizar y procesar el video obtenido por el UAV's de una determinada área, en la Tabla 10 se ilustra cada una de estas etapas.

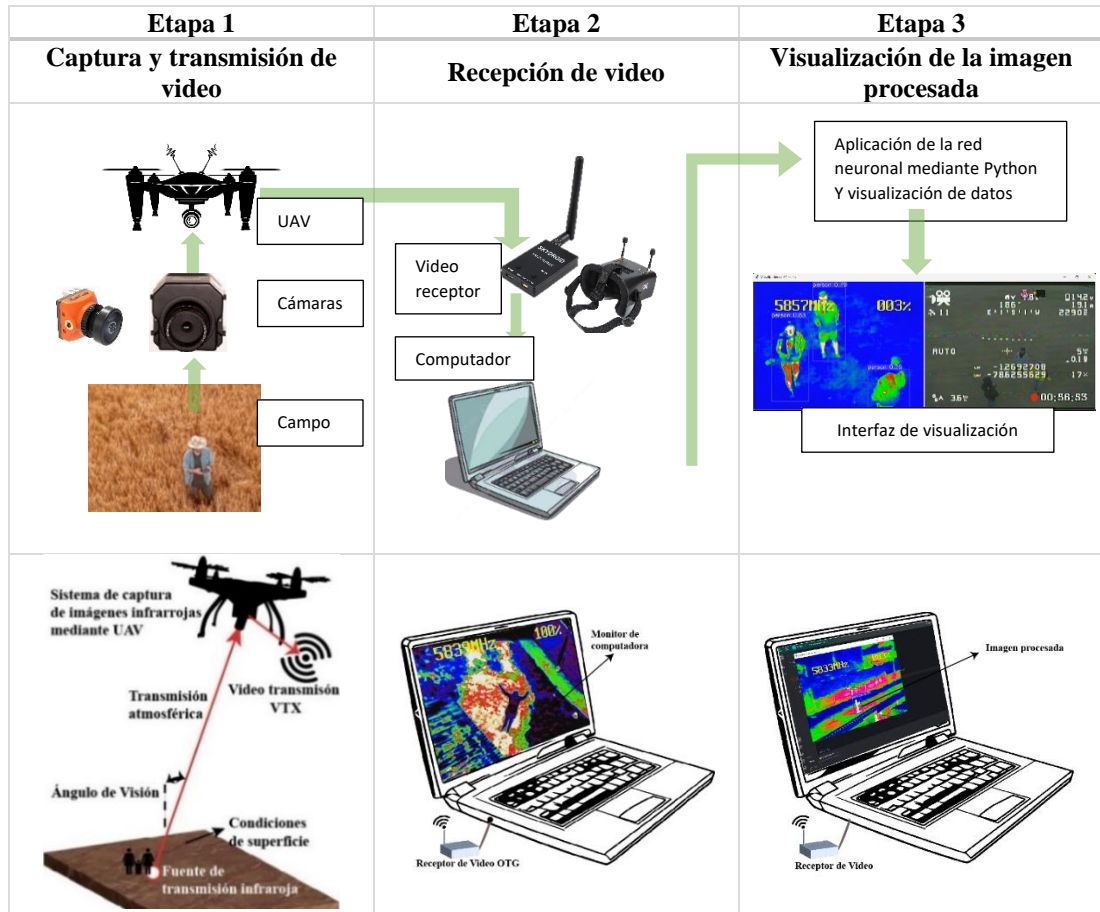
La etapa uno corresponde a la captura del video mediante la cámara térmica de FLIR modelo Tau 2 desde la fuente de transmisión infrarroja, posterior a ello, la señal de video obtenida es enviada mediante una señal analógica hacia el transmisor de video y transmitida mediante señales de radio a 5.8 GHz hacia la estación terrestre. En esta

misma etapa se realiza la captura de video por la cámara RGB para la extracción de datos de localización mediante OSD y es transmitida mediante el mecanismo anterior.

La segunda etapa corresponde a la recepción del video mediante un receptor inalámbrico tipo OTG (On The Go) para el video de la cámara térmica y gafas receptoras FPV para el video de la cámara RGB con los datos de localización. Esto con la finalidad de procesar la señal analógica enviada por el transmisor y convertirla a una señal digital para visualizar la imagen a través del computador.

La tercera etapa implica la identificación de la persona a través del procesamiento del video utilizando el software Python. En este proceso se utilizó el modelo de detección de objetos YOLO (You Only Look Once) y se entrenó la red neuronal en Google Colab con 1000 imágenes tomadas desde el UAV. En esta etapa se puede visualizar la detección de la persona desde el UAV y la visualización de los datos de longitud y latitud mediante la cámara RGB.

Tabla 10. Esquema general del sistema y sus etapas





3.1.5 Selección de los elementos de hardware para el sistema de localización

Se analizaron los componentes de hardware que conforman el sistema de localización de personas, para ello se empleó tablas comparativas para evaluar las características de cada dispositivo, con el objetivo de seleccionar aquellos que ofrezcan especificaciones técnicas óptimas para el desarrollo del sistema.

a. Vehículo aéreo no tripulado (UAV)

El UAV es el elemento central para el desarrollo del proyecto, ya que actúa como plataforma para la cámara térmica. En la Tabla 11, se evalúan diversos UAV's que la Universidad Técnica de Ambato ofrece para distintos propósitos de investigación.

Tabla 11. Características generales de los UAV's

Características		DJI - Phantom 4	MATEK- FPV
			
Aeronave	Peso	1380 g	420 g
	Velocidad máxima	20 m/s	30 m/s
	Tiempo de vuelo máximo	28 minutos	20 min
	Sistema de posicionamiento por satélite	GPS/GLPNASS	GPSS
Sistema de visión	Campo de visión	60 ° H, ±27 ° V	120 °
	Frecuencia de medición	10 Hz, 30 Hz	30 Hz
Control remoto	Frecuencia de operación	2400 – 2283 GHz	2,4 - 2,48 GHz
	Distancia de transmisión	5 km	1 km
	Batería	LiPo 2s 6000 mAh	6 V 1,5 A
	Potencia del transmisor	23 dBm	20 dBm
	Soporte para dispositivo móvil	Teléfono y Tablet	No
Cardán	Estabilización	3 ejes; -90° a + 30°	No
Cámara	Sensor	CMOS 1/2.3''	CMOS
	Lente	FOV 94° 20 mm	120
	Rango ISO	Video: 100-3200	Automático
	Tamaño de imagen máxima	4000x3000	1080 x 1200
	Foto	JPEG, DNG	No
	Video	MP4, MOV	HD Analógico

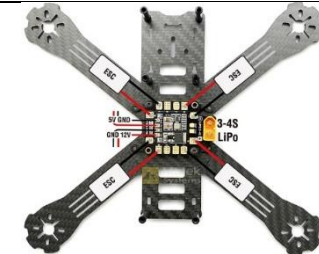
Al analizar las características distintivas de los UAV's disponibles, se destaca la excelencia del modelo DJI. No obstante, la complejidad se manifiesta al integrar elementos suplementarios, como una cámara térmica adicional y un transmisor de video, presentando particular dificultad para acceder a la placa necesaria para alimentar este último componente. En consideración a este desafío técnico, se ha

tomado la decisión de emplear el UAV proporcionado por la empresa MATEK, dado que este tipo de vehículos aéreos se prestan a modificaciones y exhiben una mayor resistencia en comparación con los modelos de DJI.

b. Características específicas del UAV seleccionado


- **Estructura.** El diseño del UAV adopta la configuración de un cuadricóptero, siendo su estructura fabricada en fibra de carbono. Este material se distingue por su excepcional resistencia y ligereza, las dimensiones precisas en milímetros (mm) y el peso específico se encuentran detallados en la Tabla 12.

Tabla 12. Medidas de la estructura del UAV Matek

Estructura	Ilustración
Largo: 135 mm Ancho: 165 mm Alto: 40 mm Peso: 110 g	


- **Motores sin escobillas:** El UAV está equipado con cuatro motores Racerstar, cuyas especificaciones se encuentran detalladas en la Tabla 13.

Tabla 13. Características del motor Racerstar

Motores Racerstar - Br 2205	Ilustración
RPM: 2300 KV Corriente máxima: 27,6 A Potencia máxima: 408 W Pull: 950 g Peso: 28 gramos Batería: Lipo 3-4S	


- **Controlador electrónico de velocidad:** También conocido como ESC (del inglés, Electric Speed Controller), son los encargados de mantener una velocidad estable proporcionando una señal de corriente idónea a cada motor, en la Tabla 14 se describen las características del modelo incorporado en el UAV seleccionado.

Tabla 14. Controlador eléctrico de velocidad

BLHELI-S	Ilustración
Modelo: RS20A Amperios: 20 A Voltaje de entrada: 3S – 4S Peso: 5.4 gramos	


- Tarjeta controladora de vuelo.** La tarjeta integrada en el UAV es la F405-STD de MATEKSYS, una plataforma programable a través de BetaFlight OSD (On Screen Display). Esto posibilita la visualización de datos cruciales como altura, velocidad, voltaje, longitud y altitud de la aeronave [43]. Las características detalladas de este modelo se encuentran descritas en la Tabla 15.

Tabla 15. Tarjeta controladora de vuelo

MATEKSYS F405-STD	Ilustración
<ul style="list-style-type: none"> Procesador <ul style="list-style-type: none"> - STM32F405RGT6 ARM (168MHz) Sensores <ul style="list-style-type: none"> - ICM20602 IMU en versión STD, MPU6000 en versión CTR - Barómetro BMP280 (STD and CTR) - Sensor de tensión y corriente 184^º en versión CRT Interfaces <ul style="list-style-type: none"> - 6x salidas PWM (7 on -STD) - 1x entrada RC PWM/PPM, SBUS - Puerto USB - OSD incorporado Tamaño y dimensiones <ul style="list-style-type: none"> - PCB de 36x36mm con orificios de montaje de 30.5mm - STD: 7g - CTR: 10g 	


- Placa de distribución de energía:** Esta placa, conocida como PBD (del inglés, Power Distribution Board en inglés), tiene la función principal de distribuir la energía a los motores, las luces y la tarjeta controladora de vuelo. Las características específicas de este modelo se encuentran detalladas en la Tabla 16.

Tabla 16. Placa de distribución de energía

PDB-XT60	Ilustración
Rango de voltaje de entrada: 9 – 18 V Corriente continua: 24A*4 o 15A*6 Salidas reguladas de 5 V y 12V	


- **Mini receptor.** Este componente es fundamental al establecer la conexión entre el control y la aeronave, permitiendo un manejo preciso y eficiente diseñado para recibir y emitir señales PWM, opera con el protocolo AFHDS 2A (Sistema Digital de Salto de Frecuencia Automático). Además, es compatible con la salida estándar PPM y brinda soporte i-BUS para hasta 18 canales. Para obtener información detallada sobre sus características, se puede consultar la Tabla 17.

Tabla 17. Mini receptor Fky Sky

Fly Sky 8A VER20	Ilustración
Tipo de salida: PPM i-BUS 2,4 G Alcance: >1km Protocolo: Hormiga Salida: PWM/PPM/i-BUS/S.BUS	


- **Radio control:** El controlador Fly Sky se distingue por su comunicación exclusiva, lo que significa que está diseñado específicamente para el uso con el UAV seleccionado. Es responsable de enviar las señales necesarias para el funcionamiento del sistema. Las características detalladas de este modelo se encuentran descritas en la Tabla 18.

Tabla 18. Transmisor Fly Sky

Fly Sky FS-i6X	Ilustración
Frecuencia: 2.408 GHz a 2.475GHz Comunicación bidireccional: Capaz de enviar y recibir datos de temperatura, altitud y otros sensores y soporte i-BUS Antena de ganancia Omnidireccional: Reduce interferencias	


- **Batería:** La batería que utiliza el UAV seleccionado es de tipo LiPo de 4 celdas, ya que los motores y la placa distribuidora de energía reciben un máximo de 4s, las características se pueden visualizar en la Tabla 19.

Tabla 19. Batería FUN FLY

FUN FLY	Ilustración
Número de celdas: 4s Amperaje: 1550 mAh Descarga: 100C	

- GPS:** En el UAV seleccionado se encuentra un GPS desarrollado por la empresa Taoglas, el cual es utilizado principalmente para estabilizar vuelos programados en UAV's, opera mediante dos tipos de frecuencia: GLONASS, una tecnología creada por Rusia; GALILEO, un sistema desarrollado por Europa; y GPS, una tecnología estadounidense [44]. En la Tabla 20 se describen sus características principales.

Tabla 20. Características del GPS

Taoglas – CGGP.25.4.E.02	Ilustración
Frecuencia de operación: - GLONAS: 1602±5MHz - GPS/GALILEO: 1575,42 ±1,023 MHz Eficiencia: 88,02 %; 88,63% Ganancia máxima; 5,39 dBi; 5,46dBi Polarización: Lineal Impedancia: 50 ohm Peso: 9,5 g	

c. Conexiones de todos los componentes al UAV

Desde la página oficial de ArduPilot se puede obtener información sobre las conexiones de todos los componentes del sistema que integra el UAV seleccionado, es por ello, en la Figura 40 se ilustra el diagrama de cableado de los componentes como el GPS, el receptor del radio control, los motores y la conexión entre el controlador de vuelo y el PBD.

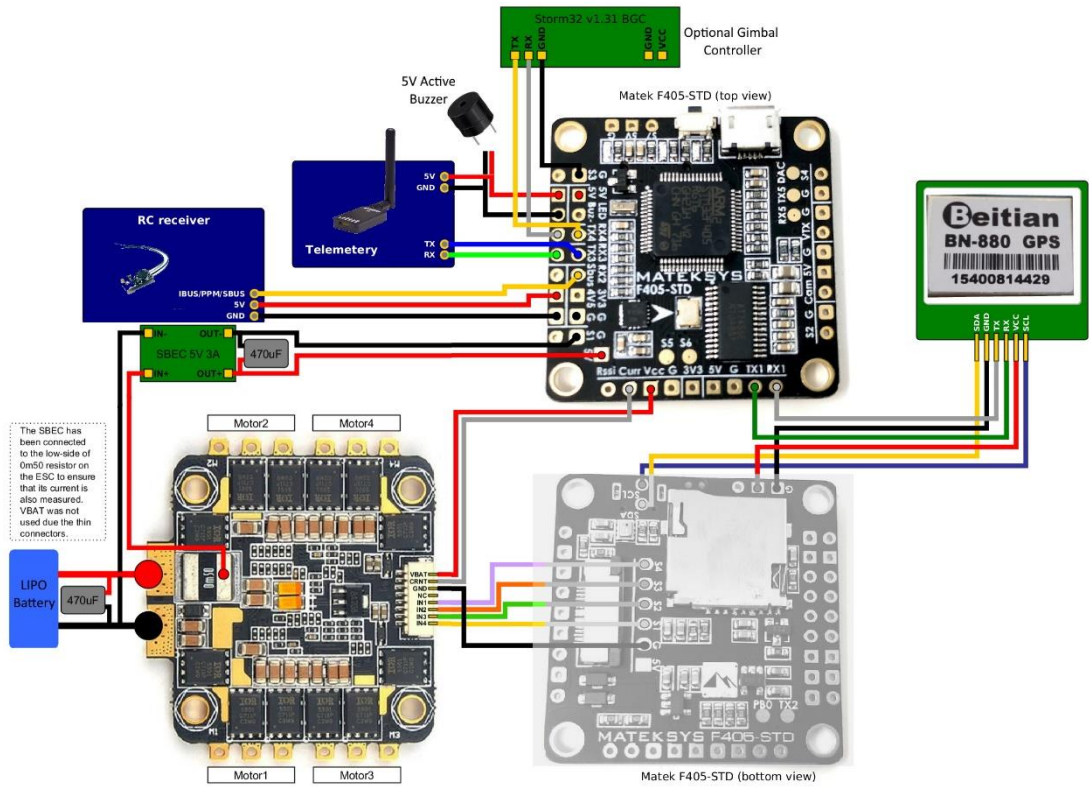


Figura 40. Diagrama de cableado [39].

En la Figura 41 se puede visualizar la conexión eléctrica para conectar la cámara y ESC que proporciona el fabricante de la placa controladora de vuelo.

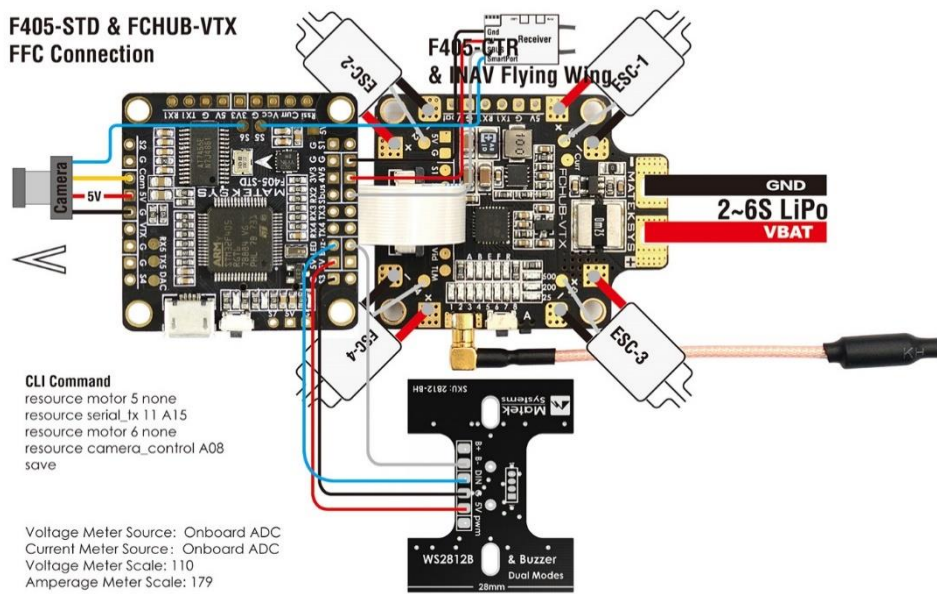


Figura 41. Conexión de la cámara y ESC [43].

d. Dispositivo Captura de video

- **Cámara RGB.** La cámara es el dispositivo encargado de capturar el video necesario bajo parámetros como resolución en píxeles y velocidad de fotogramas (fps), a continuación, en la Tabla 21 se detallan características de modelos de cámaras para UAV's que actualmente están en el mercado, la DJI Zenmuse Xr con salida digital y la CMOS Orange con salida analógica utilizada en UAV's para vuelos en FPV.

Tabla 21. Características de cámaras RGB

Características	DJI Zenmuse Xr	CMOS Orange 1200
Ilustración		
Tamaño	17.3 x 13.0 mm	0.984 in x 0.984 in
Sensor	CMOS	CMOS
Píxeles	16.0 M	1MP
Peso	80 gramos	9.64 gramos
Formato y resolución	RAW: 4K (3840 X2160)	PAL: 1920H×1080V (1MP) NTSC: 1020H×1920V (1MP)
Salida de video	Digital	Analógico
Lente	19mm	2.79 mm
FOV	72 grados	120 grados
Voltaje de funcionamiento	5.00 – 12.00 VCC	5.00 – 12.00 VCC
Precio	USD 1800	USD 20

En el marco del desarrollo del proyecto, se optó la decisión estratégica de integrar la cámara Orange 1200, notoriamente liviana con un peso de 9.64 gramos, lo cual se consideró como una ventaja significativa en comparación al modelo DJI. Esta elección se justificó en función de su complementariedad con la inclusión adicional de una cámara de alrededor de 70 gramos, acompañada por 4 soportes de protección y el respectivo protector diseñado para esta cámara específica. De este modo, se buscó equilibrar el peso total del conjunto de elementos incorporados en el proyecto, teniendo en cuenta tanto la funcionalidad requerida como la optimización del peso general del sistema.

- **Cámara térmica.** Es el dispositivo que ofrece una imagen térmica de objetos sin necesidad de contacto físico, ya que utiliza las emisiones de radiación infrarroja y los presenta en una escala de colores dependiendo de la temperatura. En la Tabla 22 se comparan dos cámaras térmicas con la finalidad de seleccionar una para el desarrollo del proyecto, este análisis se realiza bajo parámetros como el rango de temperatura a medir, rango espectral, sensibilidad, resolución y campo de visión FOV.

Tabla 22. Características de las cámaras térmicas

Características	Flir Tau 2	Flir Hadron
Ilustración		
Tamaño	1.75" x 1.75" x 1.18"	0,94" x 1,77" x 1,42"
Peso	70 gramos	42.8 gramos
Formato y resolución	640 x480 (NTSC) 640X512(PAL)	320x256 píxeles
Salida de video	Analógico digital	Analógico digital
Lente	6.8 mm	6.3 mm
FOV	46°H x 35°V 2.267mr	35 °
Voltaje	4.0 – 6.0 VCC	5VCC
FPS exportable	30 Hz (NTSC)	60 Hz o 30 Hz
Temperatura	-20°C a + 100°C	-20°C a + 60°C
Banda espectral	7.5 μm – 13.5 μm	13 μm
Sensibilidad	<60mK NEDT	<50mK NEDT

Las cámaras térmicas son ideales para detectar variaciones de temperatura y ofrecen visión en condiciones de baja visibilidad, mientras que las cámaras RGB sobresalen en la identificación precisa de detalles visuales y el reconocimiento de objetos en entornos bien iluminados. En contextos de búsqueda y rescate, se combinan sistemas que integran ambas tecnologías (RGB y TIR) para aprovechar las fortalezas de cada una, proporcionando así una información más completa de la escena.

En la Tabla 23 se encuentra un análisis detallado sobre el principio de funcionamiento, ventajas, desventajas y principales aplicaciones de ambas tecnologías. Este análisis proporciona una visión más clara de su utilidad, siendo esencial para elegir la cámara óptima en operaciones de búsqueda y rescate. Destaca las capacidades específicas de cada tecnología, considerando características de escenarios reales donde la localización de personas se dificulta, como entornos de oscuridad.

Tabla 23. Comparativa entre cámaras térmicas y RGB

	Cámaras TIR	Cámaras RGB
Principio de funcionamiento	Detectan la radiación infrarroja emitida por los objetos en función de su temperatura y representan esta información en una imagen térmica en escala de colores o grises, los colores más cálidos representan las temperaturas más altas y viceversa [22].	Las cámaras RGB capturan imágenes mediante la detección de la luz visible en tres canales de color: rojo, verde y azul. Posteriormente, combina estos canales para crear una imagen en color que refleje la apariencia visual de la escena [26].
Ventajas	Detección en la Oscuridad: Las cámaras TIR pueden detectar objetos en completa oscuridad ya que no dependen de la luz visible [23].	Identificación de detalles: Las cámaras RGB son excelentes para la identificación de detalles, lo que la hace ideal para la detección de objetos [26].
	Detección de temperatura: Mediante una escala de colores, representa la temperatura.	Costo Asequible: Debido a su amplia aplicación, el costo es menor.
Desventajas	Limitada Identificación de Detalles: Debido a que se basan en diferencias de temperatura y no en la forma o el color [23].	Dependencia de la Luz Visible: Requieren luz visible para funcionar adecuadamente, la detección en la oscuridad es deficiente [26].
	Costosas: Las cámaras TIR suelen ser más costosas que las cámaras RGB tradicionales.	Limitaciones en condiciones adversas: En condiciones de poca luz o con niebla densa.
Aplicaciones	Se utilizan comúnmente en aplicaciones de seguridad, vigilancia, búsqueda y rescate, control de procesos industriales y detección de personas en la oscuridad [23].	Son ideales para aplicaciones de reconocimiento facial, vigilancia en entornos bien iluminados, fotografía y video convencional, identificación de objetos [26].

La localización de personas en situaciones de búsqueda y rescate puede ser más desafiante en la oscuridad, dado que la falta de luz dificulta la ubicación. Las cámaras térmicas, al funcionar con radiación infrarroja y no depender de la luz visible, resultan más efectivas en estas circunstancias, por esta razón, se optará por la cámara térmica FLIR modelo Tau 2. Esta elección se basa en su superioridad técnica, especialmente en términos de su rango de temperatura de trabajo y su espectro de detección, aspectos fundamentales al utilizar esta tecnología en operaciones de rescate.

e. Modelos en 3D

Para incorporar la cámara térmica al UAV se necesita un soporte adherido al UAV y un protector para la cámara como se puede visualizar en la Figura 42, por tal motivo se ha diseñado un modelo en 3D e impreso en material filamento rojo de poliuretano termoplástico (TPU) para una mayor resistencia ante posibles caídas.

La integración de la cámara térmica requiere un soporte específico para el UAV y un protector que permita regular el ángulo de visión de la cámara, como se detalla en la

Figura 42. Por esta razón, se ha creado un modelo en 3D y se ha impreso utilizando TPU, un material que proporciona mayor resistencia para proteger la cámara térmica en caso de caídas.



Figura 42. Diseños en 3D. a) Soporte, b) protector de para cámara Tau 2

Para la protección de los componentes se ha diseñado dos soportes específicos para resguardar la cámara durante los despegues y aterrizajes del UAV. En un total de cuatro soportes de 9 centímetros: dos derechos y dos izquierdos para cada motor, como se puede visualizar en la Figura 43.





Figura 43. Soportes para aterrizaje y despegue

f. Transmisión de video

En un nivel básico, los sistemas de transmisión y recepción implican el uso de un video transmisor (VTX) que está montado en el UAV y es el encargado de enviar la señal analógica de la cámara a un receptor de video (VRX) ubicado en la estación terrestre [33]. En la Tabla 24 se analiza 3 transmisores de video de diferentes marcas con la finalidad de seleccionar la mejor opción.

Tabla 24. Características de transmisores de video


Características	SpeedyBee-TX 800	Eachine – TX 801	Eachine – TX5258
			
Peso	5,6 g	7,3 g	8.25 g
Tamaño PCB	28 * 28 * 5 mm	28 * 20 * 8 mm	31.88 * 22.90 * 5 mm
Canal de frecuencia	5.8 GHz – 48 canales	5.8 GHz – 72 canales	5.8 GHz – 72 canales
Potencia de salida	25/200/400/800mw	100/200/400/600mw	100/200/400/600mw
Alcance	2 km	3km	2.5 km
Corriente de trabajo	5 V – 250 a 750 Ma	7-24V	7-24V
Salida de voltaje	5 V – 250Ma	5V	5V
Impedancia de video	75 Ω	10 K Ω	10 K Ω
Ancho de banda	18MHz	18MHz	18MHz
Formato de video	NTSC/PAL	NTSC/PAL	NTSC/PAL
Conector de antena	MMCX	BR-SMA	BR-SMA

El transmisor Eachine – TX801 destaca como la mejor elección debido a su conector BR-SMA, evitando así la necesidad de un adaptador. Además, este transmisor está disponible en los mercados ecuatorianos, lo que simplifica su adquisición.

g. Receptor de video

Para visualizar el video transmitido por el VTX, es imprescindible contar con un receptor de video que se conecte a la computadora para procesar la señal en tiempo real. En este contexto, la tecnología OTG se utiliza en estos receptores de video, en la Tabla 25, se examinan dos receptores de video disponibles en el mercado.

Tabla 25. Receptores de video

Características	Eachine – ROTG01	Boscam - RC305
		
Canales	8 canales	8 canales
Rango de frecuencias	5645 – 5945	5725 – 5865
Sensibilidad	-90 dBm	-90 dBm
Potencia	1000mW	350mW
Corriente de trabajo	200 mA / 5 V	200 mA / 5 V
Tensión de alimentación	5V	5V
Conector de antena	Rp-SMA Hembra	Rp-SMA Hembra
Conector de salida	Mini USB	Jack 3.5 mm

El receptor de la marca Eachine modelo ROTG01 técnicamente tiene mejores características como el rango de frecuencias de trabajo y la potencia, lo cual es fundamental para lugares en el que es difícil acceder, además la salida de video es tipo mini USB en comparación al modelo RC305 que tiene salida en jack de 3.5mm, por tal motivo se selecciona el modelo ROTG01.

3.1.6 Altura de vuelo del UAV

La altitud de vuelo de los UAV's resulta crucial para optimizar la búsqueda de posibles víctimas en situaciones de catástrofe. Para lograr una identificación y localización precisa de una persona, esta debe ocupar un área mínima (en píxeles) dentro del campo de visión de la cámara. La proyección del área vista en el suelo y el tamaño relativo de objetos o personas dentro del campo visual de la cámara están directamente ligados a la altura a la que se encuentra el UAV.

Para calcular una altura idónea en el que la persona sea visible en al menos 10 píxeles (px), es necesario calcular el GSD (del inglés, Ground Sample Distance) que es la distancia en centímetros (cm) que cubre cada píxel. Para lograr esto en primer lugar, se debe calcular el ancho del sensor en milímetros (mm).

El ancho del sensor se calcula con el tamaño de la salida de video de la cámara analógica Tau 2, la cual es de 640 x 512 en NTSC y un ancho entre píxel de 17 μm . Es decir, para obtener el ancho del sensor en unidades de longitud se debería multiplicar los 640 px por la distancia entre los píxeles, tal como se describe en la Ecuación (5).

$$\begin{aligned} \text{Ancho del sensor} &= 640 \times 17\mu\text{m} \\ \text{Ancho del sensor} &= 640 \times 1.7 \times 10^{-5} \\ \text{Ancho del sensor} &= 0.01088 \text{ m} * \frac{1000\text{mm}}{1\text{m}} \\ \text{Ancho del sensor} &= 10.88 \text{ mm} \end{aligned}$$

Para calcular el GSD se utiliza la Ecuación (6) y es necesario obtener datos que se describen en la Tabla 26.

Tabla 26. Datos para calcular el GSD

Datos	Unidades	Datos
Ancho del sensor	Milímetros (mm)	10,88 mm
Altura	Metros (m)	16 m
Distancia focal de la cámara	Milímetros (mm)	6,8
Ancho de imagen	Píxeles (px)	640 px

$$GSD = \frac{10,88 \text{ mm} * 16 \text{ m}}{6,8 \text{ mm} * 640 \text{ px}}$$

$$GSD = 0,04 \text{ m/px}$$

$$GSD = 0,04 \frac{\text{m}}{\text{px}} * \frac{100 \text{ cm}}{1 \text{ m}}$$

$$GSD = 4 \text{ cm/px}$$

Tomando como referencia el GSD calculado, para lograr una detección completa y efectiva de una persona es necesario evitar una perspectiva superior, por este motivo en la página 107 se analiza imágenes con diferentes ángulos de la cámara respecto superficie terrestre, obteniendo un rango angular menor a 45 y mayor a 0 grados.

Los 4 cm/px representa linealmente 4 cm en el suelo (4cm*4cm =16 centímetros cuadrados) lo que en 10 px sería 160cm, que equivale a la altura promedio de una persona en Ecuador, esto quiere decir que la altura máxima de vuelo del UAV es de 16 metros, considerando las especificaciones de la cámara FLIR Tau 2 con un lente de 6,8mm.

3.1.7 Diseño eléctrico del sistema de localización térmico

Para elaborar el esquema eléctrico, es fundamental conocer las entradas y salidas (E/S) de la cámara FLIR modelo Tau 2, en la Figura 44 incluye una tabla detallada que define los pines de E/S.

Pin #	Signal Name	Pin #	Signal Name
1	RS232 TX (see 3.1.7)	2	RS232 RX (see 3.1.7)
3	XP D17	4	XP D16
5	DGND	6	DGND
7	Reserved	8	Reserved
9	LVDS CLK P (see 3.1.4.3)	10	LVDS CLK N (see 3.1.4.3)
11	LVDS SYNC P (see 3.1.4.3)	12	LVDS SYNC N (see 3.1.4.3)
13	LVDS DATA1 P (see 3.1.4.3)	14	LVDS DATA1 N (see 3.1.4.3)
15	LVDS DATA2 P (see 3.1.4.3)	16	LVDS DATA2 N (see 3.1.4.3)
17	DGND	18	DGND
19	XP D15	20	XP D14
21	XP D13	22	XP D12
23	XP D11	24	XP D10
25	XP D9	26	XP D8
27	DGND	28	DGND
29	XP D7	30	XPD 6
31	XP D5	32	XPD 4
33	XP D3	34	XPD 2
35	XP D1	36	XPD 0
37	DGND	38	DGND
39	XP CLK OUT	40	Reserved
41	DGND	42	DGND
43	VIDEO H (see 3.1.3)	44	VIDEO L (see 3.1.3)
45	DGND	46	3V3 OUT (see 3.1.2)
47	MAIN PWR RTN (see 3.1.2)	48	MAIN PWR (see 3.1.2)
49	MAIN PWR RTN (see 3.1.2)	50	MAIN PWR (see 3.1.2)

Figura 44. Definición de pines del conector de E/S principal [45]

La Figura 45 proporciona la distribución de pines E/S de la cámara Flir y su orden correspondiente para realizar la conexión.

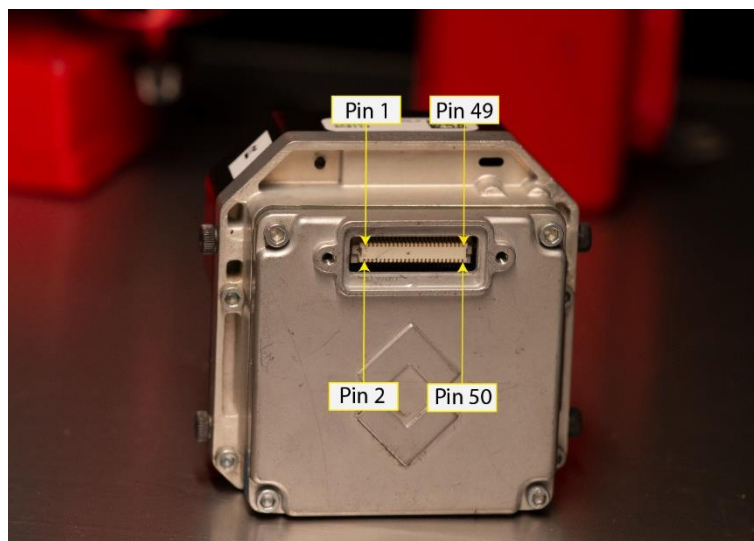


Figura 45. Distribución de pines E/S

Para obtener la señal de video de la cámara Flir modelo Tau 2, es imprescindible contar con un módulo VPC (siglas en inglés de video/power/control) compatible con dicho modelo. Después, se procede a verificar los pines pertinentes para la alimentación (Pin 48, 50), conexión a tierra (pin 47, 49) de la cámara y la salida de vídeo (pin 43), los cuales se conectan directamente al video transmisor TX80.

La fuente de alimentación principal del sistema se deriva de la tarjeta controladora de vuelo a un voltaje de 11 voltios. En la Figura 46, se presenta el esquema detallado de la conexión eléctrica del sistema.

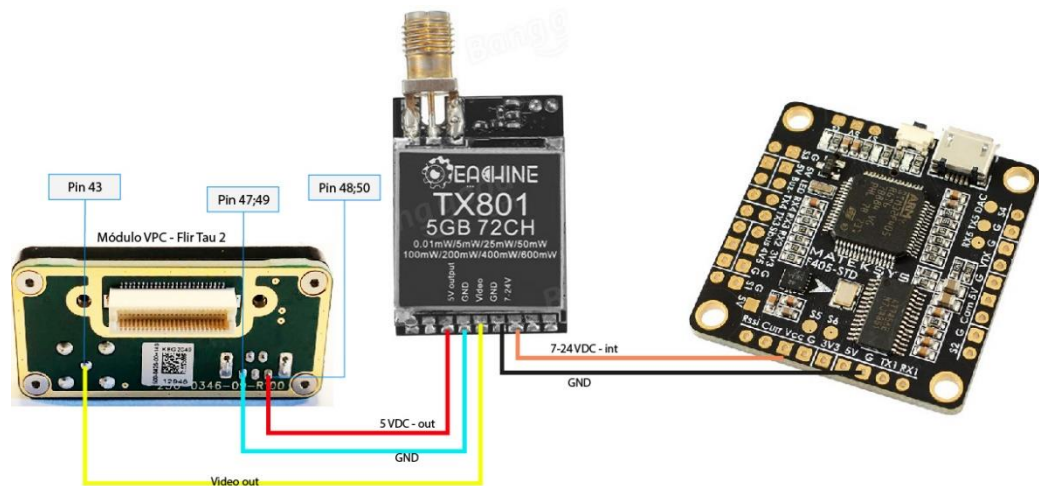


Figura 46. Esquema eléctrico de conexión

3.1.8 Diseño eléctrico del sistema de localización RGB

La placa controladora de vuelo MATEK F405 despliega un papel fundamental en la creación del esquema eléctrico para conectar la cámara FPV y el video transmisor (VTX). Para configurar adecuadamente estas conexiones, es crucial analizar detalladamente las especificaciones y capacidades de esta placa.

En el caso de la cámara, se requieren tres cables específicos: uno amarillo para la transmisión de video, que se enlaza con el pin "cam"; uno rojo para la alimentación de la cámara, conectado al pin "5v"; y un cable negro que se utiliza para establecer la conexión a tierra, marcado como "G".

Para el video transmisor VTX, también se necesitan tres cables: dos para la alimentación y uno amarillo dedicado a la transmisión de video. Estas conexiones deben realizarse a través de la placa MATEK F405, ya que esto permite acceder a datos cruciales almacenados en el OSD (On Screen Display). Esta integración con la placa no solo facilita la conexión física, sino que además permite el acceso a información esencial con la finalidad de utilizar los datos de latitud y longitud para la localización de personas, en la Figura 47 se ilustra el esquema eléctrico de conexión.

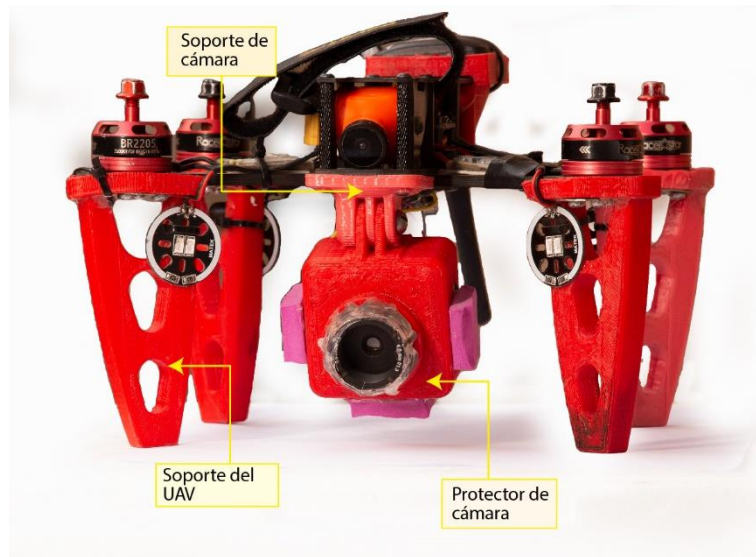


Figura 49. Implementación del diseño en 3D

3.1.11 Emisividad y ángulo de visión de la cámara (FOV)

Las cámaras térmicas no tienen la capacidad de distinguir qué porción de la radiación total que emana del objeto se refleja y qué porción se debe a la auto emisión. A todos los efectos prácticos, se puede minimizar los efectos del cambio en la temperatura aparente en todo el FOV manteniendo el ángulo de visión máximo por debajo de 45°, según menciona en [27].

Los buenos absorbentes son objetos con superficies que tienen emisividades superiores a 0.9, mientras que los buenos reflectores tienen emisividades inferiores que 0.2. La mayoría de los mamíferos tienen emisividades superiores a 0.9, lo que los hace buenos absorbentes y reflectores deficientes. Por ejemplo, la piel humana tiene una emisividad de 0.98 [27], lo cual es beneficioso para poder capturar la firma térmica desde la cámara.

Tomando como referencia la Figura 20 se calculó diferentes parámetros para saber el área que cubre el UAV en el campo. Se inicia por los valores D_C, D_M, D_F , son las distancias más cercana, media y lejana a lo largo del campo de visión de la cámara y R_C, R_M, R_F las distancias entre el UAV y la parte inferior, media y superior del campo de visión. También se calculó el campo de visión proyectado que muestra el área en el

suelo cubierta por la cámara, W_C, W_M, W_F que son los anchos del campo de visión en el eje X de la cámara para las distancias más cercana, media y más lejana a lo largo del eje “y”. Para calcular las distancias es necesario saber el ángulo en vertical de la cámara térmica, en este caso el modelo de FLIR que el campo de visión en horizontal “x” es de 46° y 35° en vertical “y”, están dadas por las Ecuaciones 10, 11, 12.

$$D_c = h \tan\left(\phi - \frac{\theta_y}{2}\right) \quad (10)$$

$$D_c = 16 \text{ m} \tan\left(45^\circ - \frac{35^\circ}{2}\right) = 8.33 \text{ m}$$

$$D_M = D_F - D_c \quad (11)$$

$$D_M = 30.74 \text{ m} - 8.33 \text{ m} = 22.41 \text{ m}$$

$$D_F = h \tan\left(\phi + \frac{\theta_y}{2}\right) \quad (12)$$

$$D_F = 16 \text{ m} \tan\left(45^\circ + \frac{35^\circ}{2}\right) = 30.74 \text{ m}$$

Para obtener los valores R_C, R_M, R_F para cada uno de los valores de D_C, D_M, D_F están dados por la Ecuación 13, a partir de esta ecuación se obtiene todos los parámetros R mediante las Ecuaciones 14, 15 y 16.

$$R = \sqrt{(h^2 + D^2)} \quad (13)$$

$$R_C = \sqrt{(h^2 + D_C^2)} \quad (14)$$

$$R_C = \sqrt{(16^2 + 8.33^2)} = 18.02 \text{ m}$$

$$R_M = \sqrt{(h^2 + D_M^2)} \quad (15)$$

$$R_M = \sqrt{(16^2 + 22.41^2)} = 27.54 \text{ m}$$

$$R_F = \sqrt{(h^2 + D_F^2)} \quad (16)$$

$$R_F = \sqrt{(16^2 + 30.74^2)} = 34.65 \text{ m}$$

En la Figura 50 se ilustra mediante un boceto las distancias calculadas con la Ecuación 13.

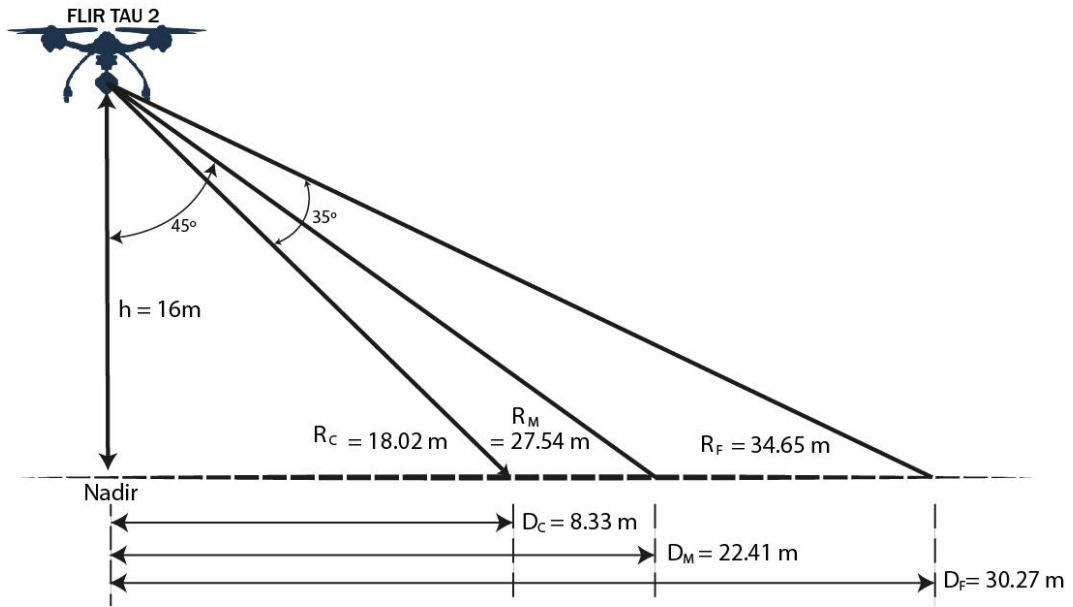


Figura 50. Boceto de una plataforma aérea

El campo de visión proyectado en la dirección “x” (horizontal visto por la cámara) varía en función de la distancia a lo largo de la dirección “y”, está dado por las Ecuaciones 17, 18 y 19:

$$W_C = 2(D_C^2 + h^2)^{\frac{1}{2}} * \tan\left(\frac{\theta_x}{2}\right) \quad (17)$$

$$W_C = 2(8.33^2 + 16^2)^{\frac{1}{2}} * \tan\left(\frac{46}{2}\right) = 11.37 \text{ m}$$

$$W_M = 2\left(\left(\frac{D_C + D_F}{2}\right)^2 + h^2\right)^{\frac{1}{2}} * \tan\left(\frac{\theta_x}{2}\right) \quad (18)$$

$$W_M = 2\left(\left(\frac{8.33 + 30.74}{2}\right)^2 + h^2\right)^{\frac{1}{2}} * \tan\left(\frac{\theta_x}{2}\right) = 15.92 \text{ m m}$$

$$W_F = 2(D_F^2 + h^2)^{\frac{1}{2}} * \tan\left(\frac{\theta_x}{2}\right) \quad (19)$$

$$W_F = 2(30.74^2 + 16^2)^{\frac{1}{2}} * \tan\left(\frac{46}{2}\right) = 21.85 \text{ m}$$

En la Figura 51 se muestra el campo de visión proyectado que muestra el área del suelo cubierta por la cámara FLIR Tau 2.

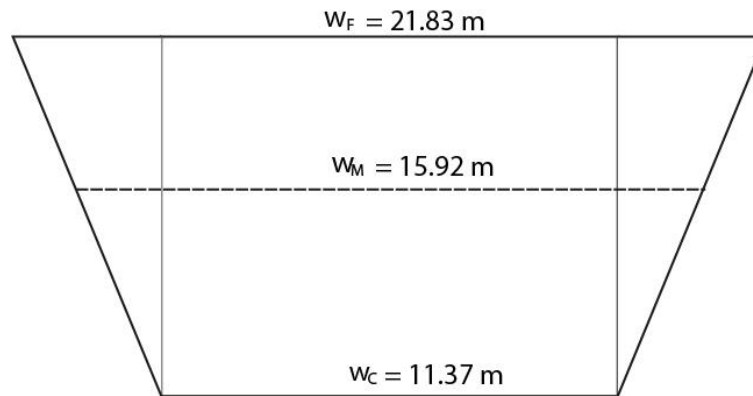


Figura 51. Campo de visión proyectada en el suelo

3.1.12 Verificación del peso de todos los componentes.

Es necesario tener en cuenta el peso total del UAV, para dicho cometido se empleó una balanza digital como se muestra en Figura 52, obteniendo un peso de 790 gramos.



Figura 52. Peso del UAV

3.1.13 Tiempo de vuelo

El tiempo de vuelo del UAV depende directamente de la capacidad de la batería, los componentes que consumen energía y el peso del vehículo, para calcular se utiliza la Ecuación (7).

Para empezar, se calcula la corriente máxima que consumen todos los equipos del UAV, en la Tabla 27 se describen todos los componentes que consumen energía para encontrar la carga máxima en amperios.

Tabla 27. Consumo en amperios de los componentes

Equipo	Número	Consumo (A)	Consumo total (A)
Motores racerstar	4	27,6	110,4
ESC	4	20	80
Mini receptor	1	8	8
Cámara RGB	1	0,075	0,075
Transmisor	2	0,55	1,1
Cámara TIR	1	1	1
GPS	1	1	1
Placa	1	1	1
Total			202,575

Para emplear la Ecuación (9), es necesario la corriente máxima consumida que se calculó en la Tabla 27 y la carga de vuelo en porcentaje. Para obtener este último dato, es necesario saber cuánto es la carga máxima que soporta el UAV y su carga actual, con el fin de obtener un porcentaje utilizado de la capacidad total.

Carga máxima = 3800 gramos

Cargar actual = 790 gramos

$$\frac{3800 \text{ g}}{790 \text{ g}} = \frac{100\%}{X}$$

$$X = \frac{790 \text{ g} * 100\%}{3800 \text{ g}}$$

$$X = 20,79\%$$

La carga de vuelo en porcentaje es de 20,79% y la corriente máxima consumida es de 202,575 A, con estos datos obtenidos se puede utilizar la Ecuación (9).

$$I_{\max full load} = 202,575 \text{ A}$$

$$I_{flying} = 20,79 \%$$

$$I_{flying\ load} = I_{max\ full\ loa} * L_{flying}$$

$$I_{flying\ load} = 202,575\ A * 20,79\%$$

$$I_{flying\ load} = 202,575\ A * \frac{20,79\%}{100\ \%}$$

$$I_{flying\ load} = 42,12\ A$$

Para obtener el tiempo de vuelo se utiliza la Ecuación (7), teniendo en cuenta la capacidad de la batería en miliamperios y la corriente consumida por todos los componentes del UAV.

$$Q = 1550\ mA$$

$$I_{flying\ load} = 42,12\ A$$

$$t_{flight} = \frac{Q}{\frac{1000}{I_{flying\ load}}} * 60$$

$$t_{flight} = \frac{1550\ mAh}{\frac{1000}{42,12\ A}} * 60$$

$$t_{flight} = 2.21\ minutos$$

El tiempo de vuelo es de 2,21 minutos, considerando que el UAV está operando a su máxima potencia, ideal para vuelos de alta exigencia en modo manual de carrera. Sin embargo, para aplicaciones de video y especialmente para operaciones de búsqueda y rescate de personas, resulta fundamental contar con un modo de vuelo estabilizado.

Este modo estabilizado se controla desde las perillas del radio control para garantizar la estabilidad y precisión en estas situaciones críticas. Cabe recalcar que este modo de vuelo requiere de un consumo energético menor por ende el tiempo de vuelo puede variar.

3.1.14 Configuración de vuelos programados autónomos en el software Ardupilot

Para iniciar la configuración de los vuelos autónomos, es fundamental que el UAV cuente con un GPS, en este caso es un Taoglas 3DR GPS+MAG.

Primero, se instaló el software descargándolo desde la página oficial de Ardupilot, conocido como Mission Planner, como se muestra Figura 53. Este software ofrece una interfaz gráfica con un mapa que permite marcar puntos de referencia asignándoles características principales como altitud y latitud en cada punto.



Figura 53. Ejecución del software Mission Planner de Ardupilot

Después de la ejecución, fue necesario conectar el UAV a la computadora. Se empleó un cable de datos micro USB a USB con el objetivo de establecer la conexión a través del puerto designado como COM3, simplemente presionando "CONECTAR". En la Figura 54 se puede observar la conexión exitosa del UAV con el software.

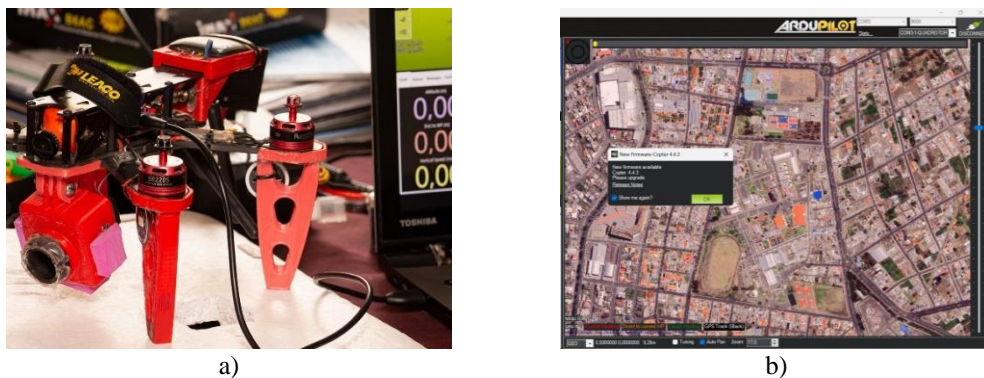


Figura 54. Conexión UAV a Mission Planner. a) Cable de datos; b) Software

Es crucial asignar un modo de vuelo automático a un interruptor del control remoto, para lograrlo, se accedió al menú SETUP y se ubicó la sección de modos de vuelo (flight modes). Se eligió el interruptor SWC, que cuenta con tres posiciones distintas: la primera para vuelos manuales, la segunda para vuelos estabilizados y la tercera para vuelos automáticos. Esta configuración se aprecia en la Figura 55, cada posición del interruptor corresponde a los modos de vuelo 1, 3 y 6, respectivamente.

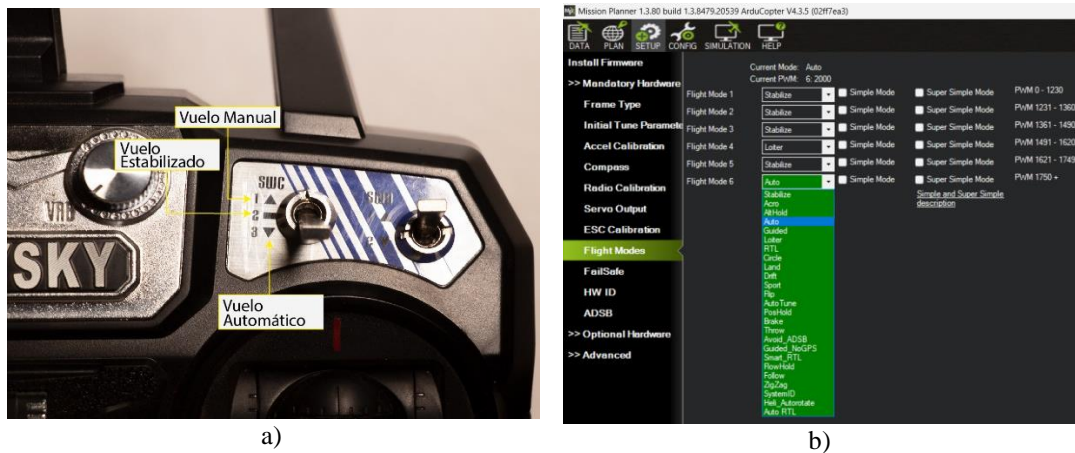


Figura 55. Modos de Vuelo. a) Radio control; b) Software

Dentro de la pestaña de configuración se encuentran los parámetros ajustables según el tipo de vuelo requerido, para realizar grabaciones de video, es crucial configurar vuelos estabilizados con una velocidad reducida, tal como se muestra en la Figura 56. Una velocidad de 70 cm/s permite grabar sin distorsionar el video capturado.

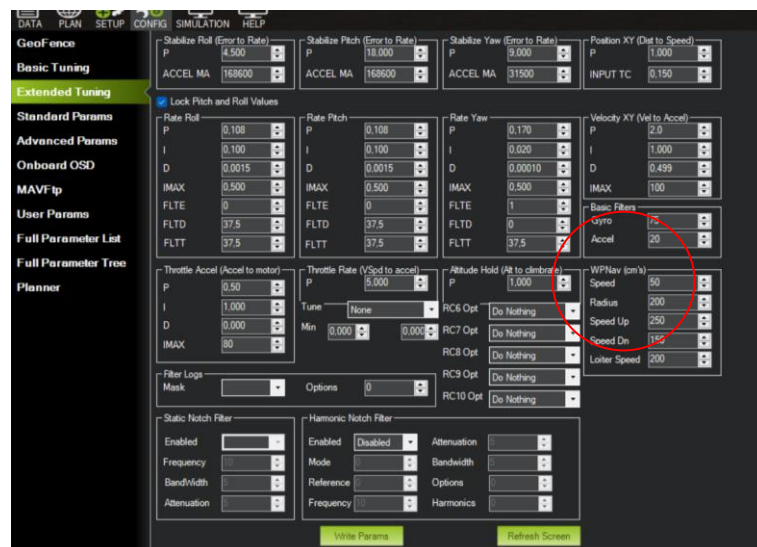


Figura 56. Ajuste de velocidad de vuelo

Una vez se estableció cada uno de los parámetros, la pestaña PLAN permite diseñar el vuelo programado, para las pruebas de vuelo prácticas se estableció en punto “HOME” o “H” en el estadio de la Universidad Técnica de Ambato, como se muestra en la Figura 57.

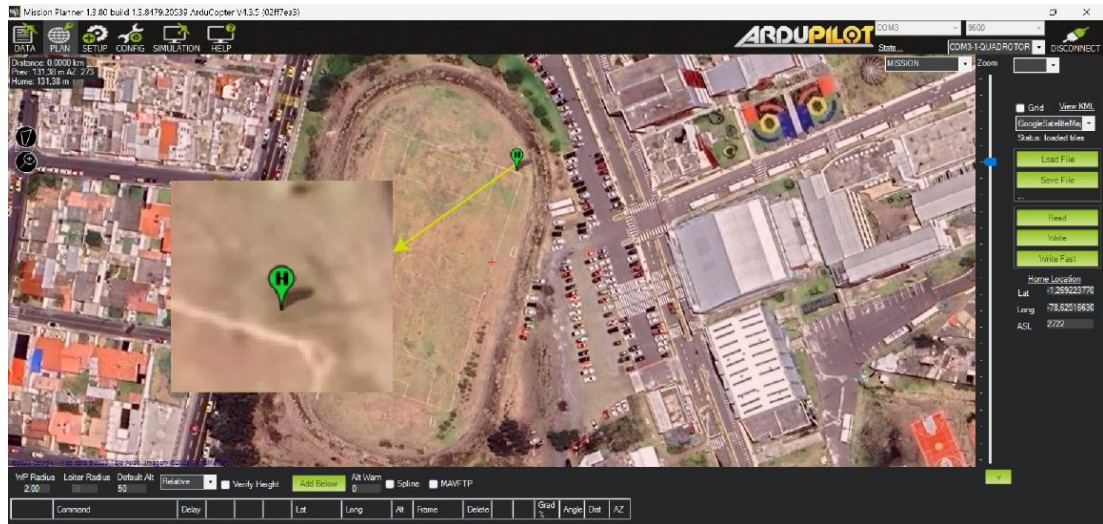


Figura 57. Punto HOME o “H”

Para el vuelo automático existen 2 tipos de comandos, los comandos denominados “DO” o condicionales y los comandos para trazar trayectorias automáticas. En la Tabla 28 se describen los comandos fundamentales para realizar un vuelo exitoso.[39]

Tabla 28. Comandos para el vuelo automático

Comando	Descripción
TAKEOFF	El vehículo ascenderá directamente desde su ubicación actual hasta la altitud especificada (en metros). Este comando debería ser el primero de todas las misiones [39].
WAYPOINT	El vehículo volará en línea recta hasta la ubicación especificada como latitud, longitud y altitud (en metros) [39].
SPLINE_WAYPOINT	Los puntos de ruta “spline” toman los mismos argumentos que los puntos de ruta normales (lat, lon, alt, delay), pero cuando se ejecutan, el vehículo volará por caminos curvos, siempre siguiendo las coordenadas.
DO_SET_ROI	Apunta la cámara hacia una región de interés específica (posición, incluida la altitud)
LOITER_TURNS	El vehículo volará en círculo alrededor de la latitud, longitud y altitud especificadas (en metros).
RETURN_TO_LAUNCH	El vehículo regresará automáticamente al lugar donde fue armado por última vez .
LAND	Utilizado para realizar aterrizajes en las coordenadas de latitud/longitudes proporcionadas [39].
DELAY	El vehículo despegará automáticamente cuando se haya cumplido el tiempo actual [39].

En la Figura 58 se representa una misión que comienza en el punto inicial 'P1', ubicado a 2 metros de altura, el punto 2 'P2' se utiliza para orientar la parte frontal del UAV hacia el punto rojo, mientras se desplaza hacia el punto 3 'P3' a una altitud de 16 metros, permaneciendo allí durante 5 segundos. Luego, se dirige al punto 4, ajustando la orientación frontal de la cámara hacia dicho punto, seguidamente, en el punto 5 el UAV inicia su descenso con el propósito de aterrizar.

Siempre es importante marcar la casilla “Verify Height” para verificar la altura mientras el UAV traza su trayectoria como se puede ver en la Figura 58. Esto es importante cuando existen colinas que pueden impedir que la trayectoria del UAV sea desarrollada con éxito, ya que pueden existir colisiones.

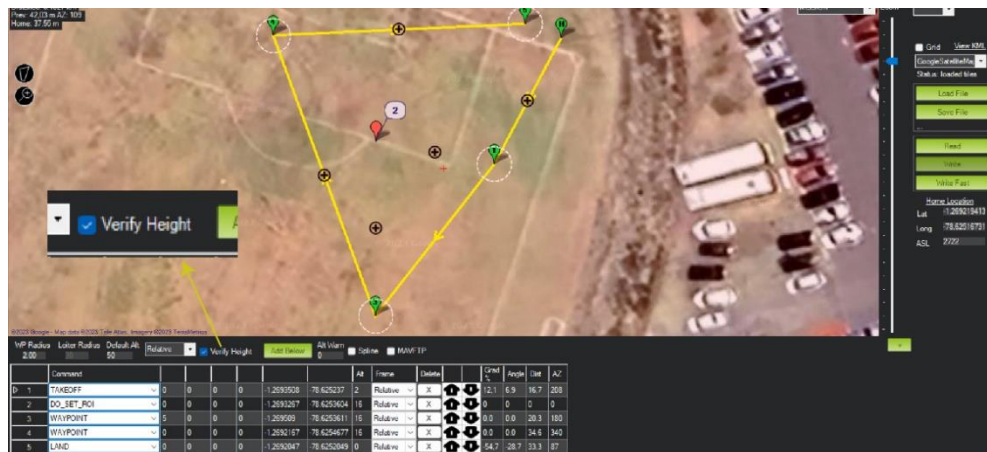


Figura 58. Trayectoria y verificar la altura

Posteriormente, se procede a cargar la trayectoria en el botón “Write” mientras el UAV se encuentra conectado al computador. Para guardar una trayectoria es necesario pulsar en el botón “Save File” para utilizarlo en otra ocasión como indica la Figura 59.

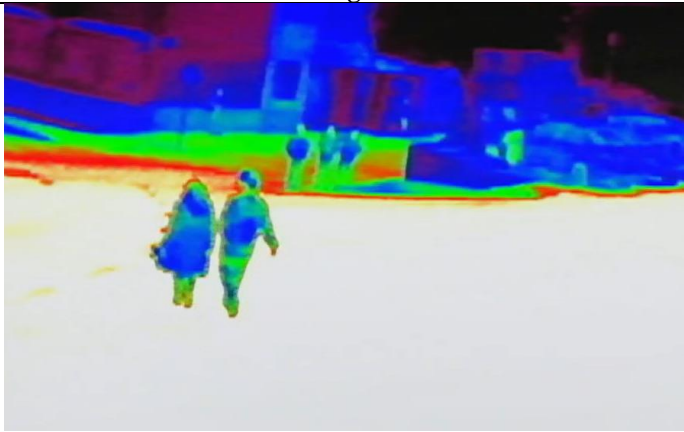
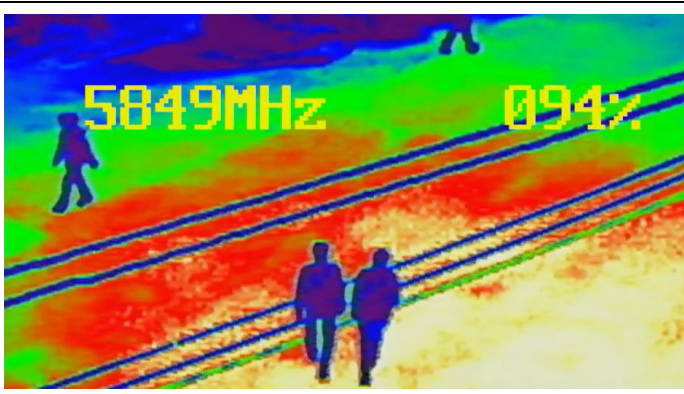



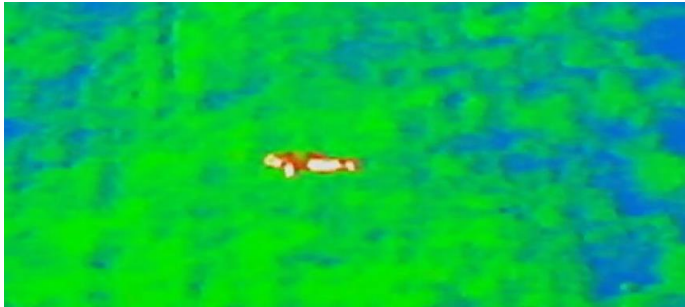
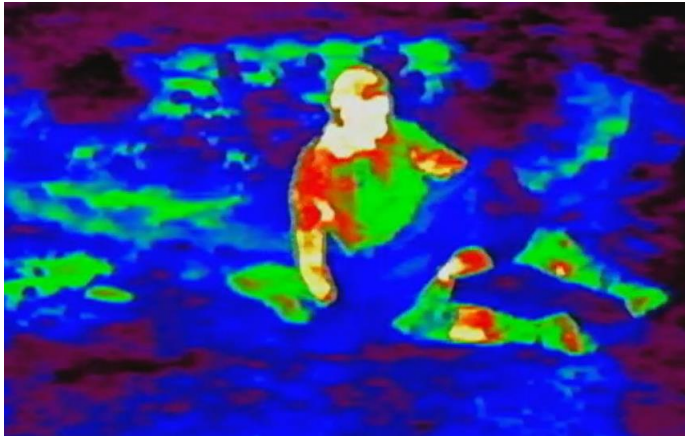
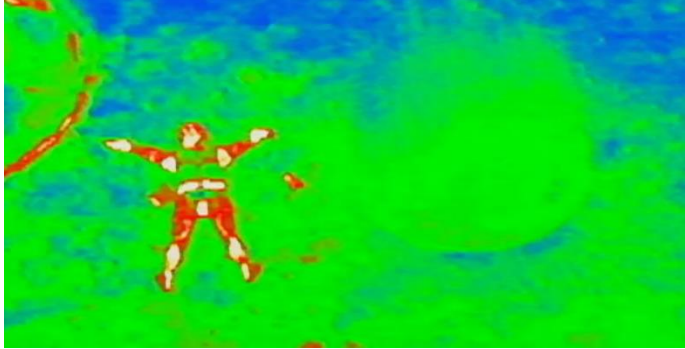
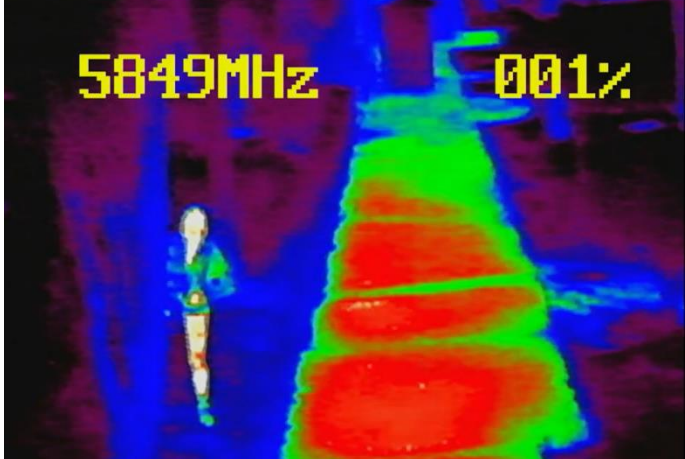
Figura 59. Escribiendo la trayectoria de vuelo en el UAV

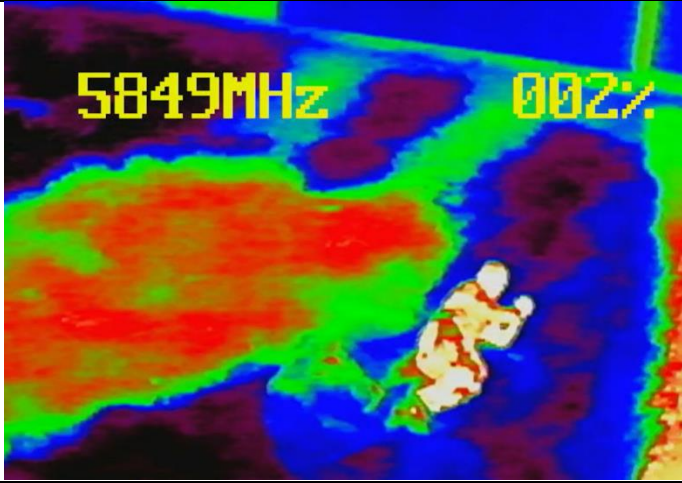
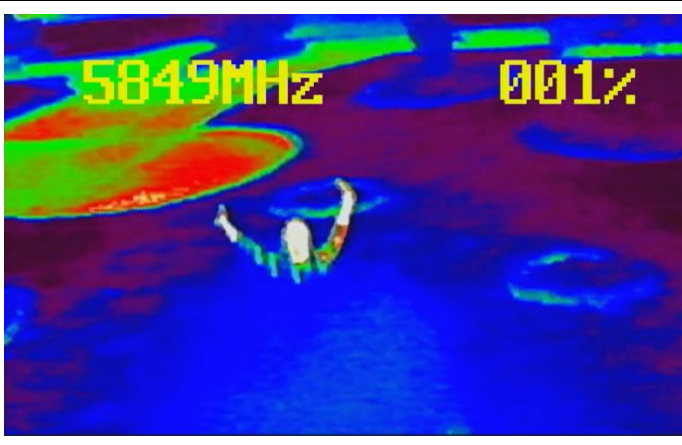

3.1.16 Configuración del sistema de captura de video y detección de personas

Para configurar el sistema de captura y detección de personas es fundamental poner en funcionamiento el sistema para obtener imágenes desde el UAV, para este procedimiento se han obtenido un total de 1000 imágenes en diversos escenarios donde podría ocurrir una catástrofe. En la Tabla 29 se analizan las principales imágenes obtenidas en las pruebas de vuelo.

Tabla 29. Imágenes térmicas en diferentes escenarios

Características del vuelo	Imagen
<p>Altura: 4 m Hora: 15:00</p>	
<p>Análisis</p> <p>Debido a la hora del día y a la superficie de hormigón, se puede ver que el piso se encuentra más caluroso que las personas, sin embargo, si se puede diferenciar a las personas de colores que representan temperaturas más bajas.</p>	
<p>Altura: 8 m Hora: 15:00</p>	
<p>Análisis</p> <p>Debido a la hora del día y a la superficie de hormigón, se puede ver que el piso se encuentra más caluroso que las personas, sin embargo, si se puede diferenciar de colores que representan temperaturas más bajas.</p>	
<p>Altura: 16 m Hora: 19:00</p>	
<p>Análisis</p> <p>Debido a la hora del día y a la superficie de césped, la persona que está en sobre la superficie emite calor, siendo fácilmente detectable.</p>	

Características del vuelo	Imagen
<p>Altura: 16 m Hora: 19:00</p>	
<p>Análisis</p> <p>Como se puede observar en la imagen, la persona se encuentra acostada en el piso, tratando de simular escenarios reales de catástrofes.</p>	
<p>Altura: 5 m Hora: 19:00</p>	
<p>Análisis</p> <p>Debido a la hora del día y a la superficie del césped, la persona que está en sobre la superficie emite calor, siendo fácilmente detectable. Además, se ha tratado de simular escenarios reales en el que la persona se encuentra acostada y cubierta con algún objeto que imposibilita su detección.</p>	
<p>Altura: 8 m Hora: 20:00</p>	
<p>Análisis</p> <p>Como se puede observar en la imagen, la persona se encuentra acostada en el piso, tratando de simular escenarios reales de catástrofes.</p>	
<p>Altura: 7 m Hora: 20:00</p>	
<p>Análisis</p> <p>El objetivo en esta imagen es tratar de obtener medio cuerpo de la persona, con la finalidad de acercarnos a la realidad de una catástrofe.</p>	

Características del vuelo	Imagen
<p>Altura: 5 m Hora: 20:00</p>	
<p>Análisis</p>	
<p>En esta imagen se trata de representar a una persona en el suelo, se puede diferenciar claramente por el calor que genera.</p>	
<p>Altura: 7 m Hora: 20:00</p>	
<p>Análisis</p>	
<p>En esta imagen se trata de representar a una persona tras de un arbusto, en el que se puede observar las extremidades superiores, la cabeza y parte del pecho.</p>	
<p>Altura: 10 m Hora: 11:00</p>	
<p>Análisis</p>	
<p>En esta imagen se visualiza dos personas en un pasto totalmente frío.</p>	

Con las imágenes obtenidas se creó una base de datos sólida que permita localizar personas en tiempo real, para esto es necesario etiquetar las imágenes, entrenar la red neuronal mediante YOLOv8, posterior a esto se desarrolla la inferencia de la red neuronal entrenada, obteniendo así la detección de personas en situaciones de catástrofe utilizando python.

a. Base de datos

Los videos transmitidos por el UAV se grabaron utilizando el software OBS para capturar la secuencia, posteriormente, se extrajeron más 1000 imágenes de los videos mediante capturas de pantalla para ser almacenado en una carpeta de datos.

La carpeta es denominada “data” en el que contiene 2 subcarpetas de imágenes “imagenes” y de etiquetado “labels”. Las dos subcarpetas tienen a su vez dos carpetas para entrenamiento denominado “train” con un total de 806 imágenes (77.35%) y para validación “val” 236 imágenes (22.65%), como se puede visualizar en la Figura 62.

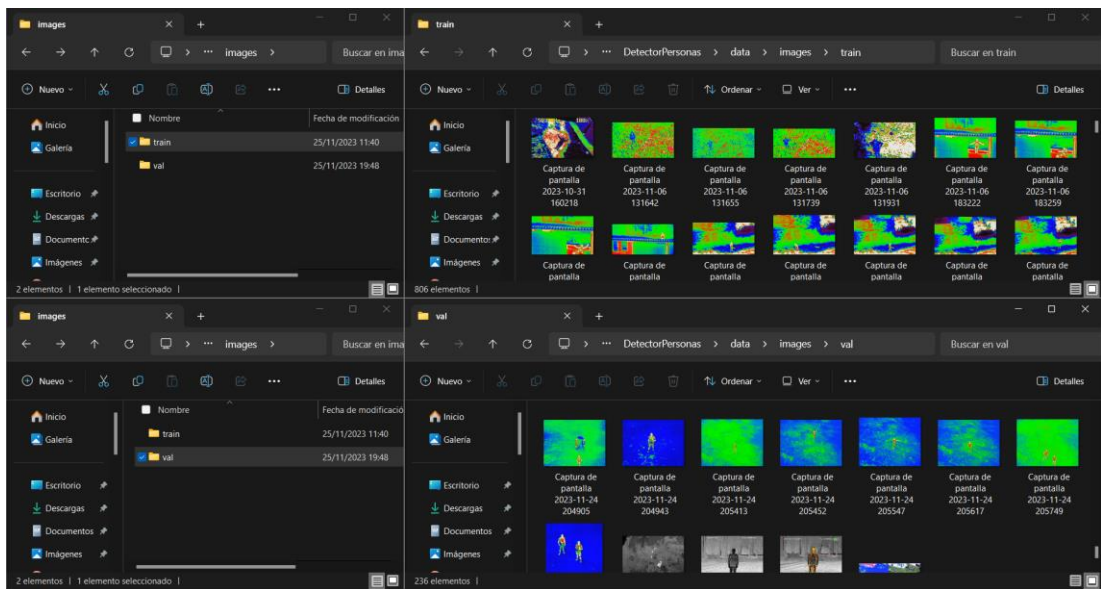


Figura 62. Distribución de la base de datos

b. Etiquetado de las imágenes

Una vez creada la base de datos es necesario etiquetar las imágenes, este procedimiento es crucial para el entrenamiento de la red neuronal. En este contexto, el etiquetado proporciona información detallada sobre la presencia, posición y el tipo de objeto o persona dentro de una imagen. Para ello se ocupó un software en línea denominado MakeSense.ai, para iniciar el etiquetado de las imágenes se pulsa el botón “Empezar”, como se muestra en la Figura 63.

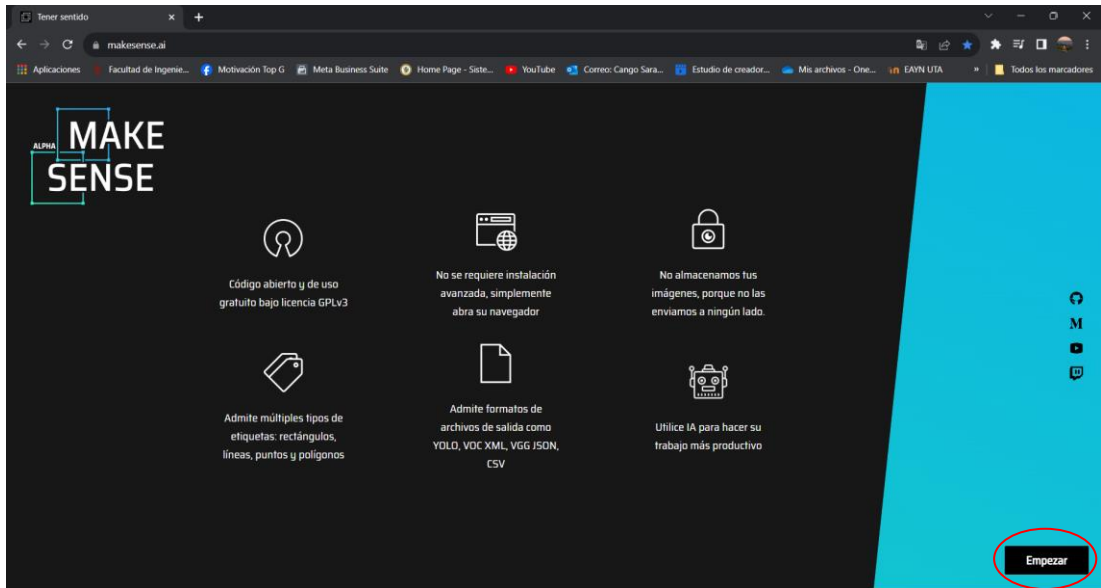


Figura 63. Software Makesense.ai

Una vez ejecutado el programa se procede a cargar todas las imágenes de la carpeta para el entrenamiento denominado “train” y se selecciona el botón de “Object Detection”, como se visualiza en la Figura 64.

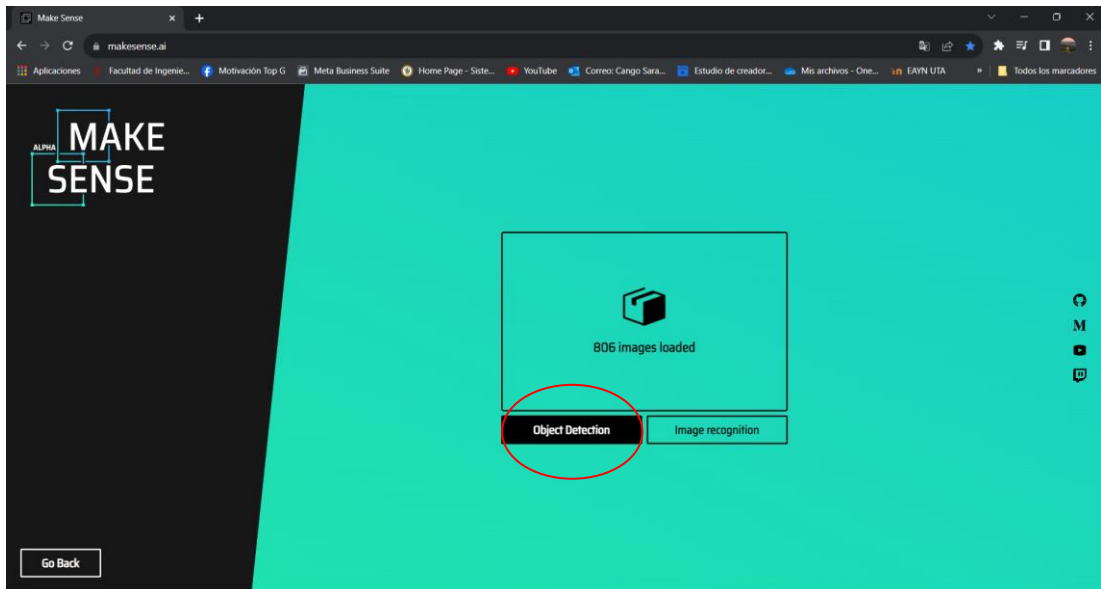


Figura 64. Detección de objetos en el software MakeSense

Posteriormente se creó una nueva etiqueta o label en el botón “+”, en donde se elige el nombre, el color de la etiqueta y se inicia el proyecto en “Start project”, Figura 65.

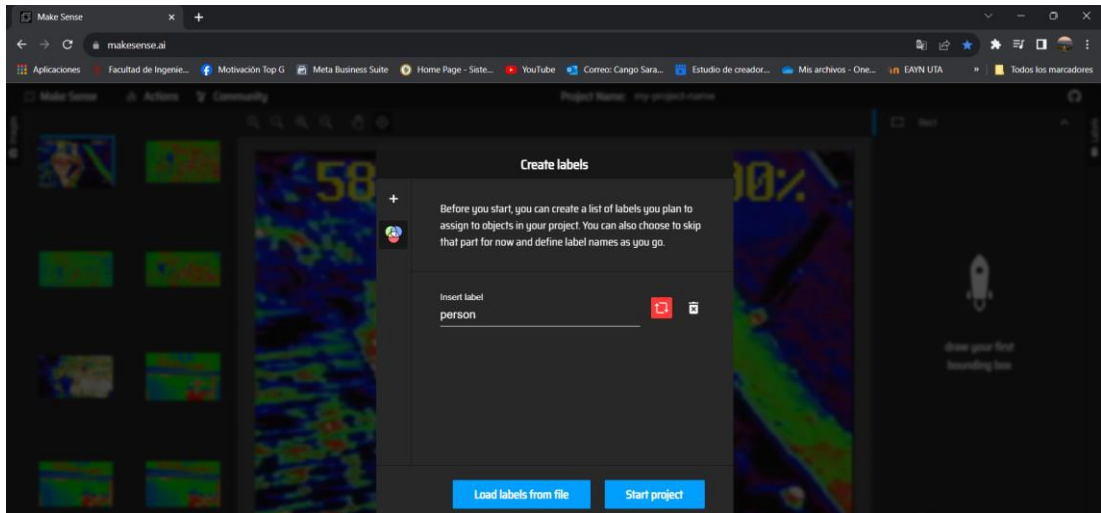


Figura 65. Crear etiquetas

Posteriormente, se busca la posición de la persona en la imagen para seleccionarlo con un cuadro delimitador y se asigna la etiqueta creada como “person”, Figura 66. Este procedimiento se realizó con todas las imágenes.

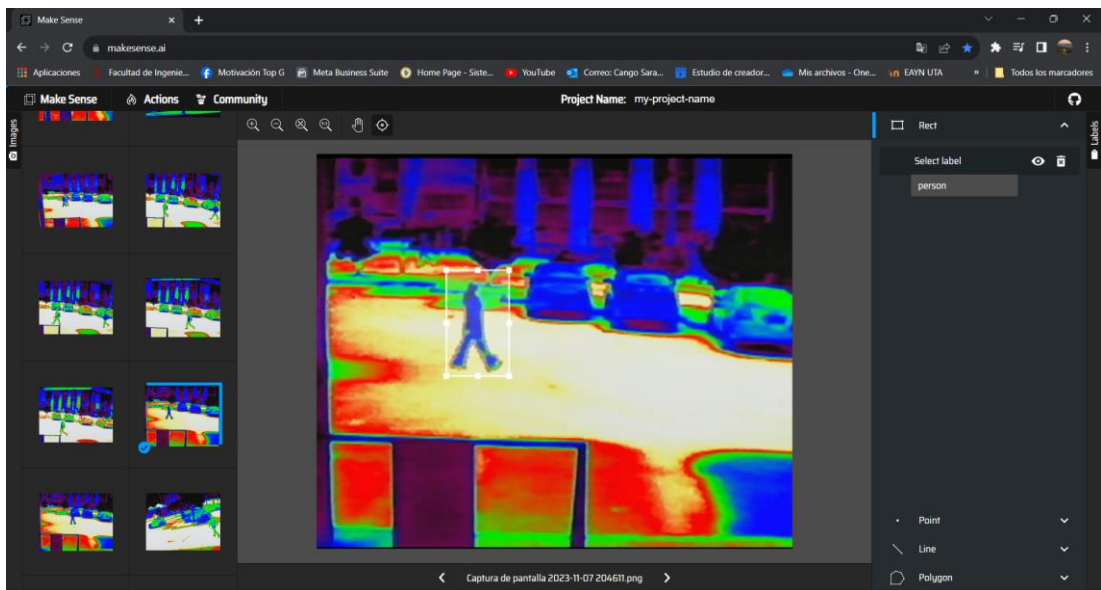


Figura 66. Asignación de etiquetas

Para guardar las etiquetas, en la pestaña de “Actions” se seleccionó “Export Annotations” para utilizar con el formato de YOLO, como se muestra en la Figura 67. Con estas etiquetas creadas la red aprende a reconocer patrones dentro de las regiones de las imágenes delimitadas por las cajas, permitiendo que el modelo pueda detectar y clasificar objetos en imágenes nuevas.

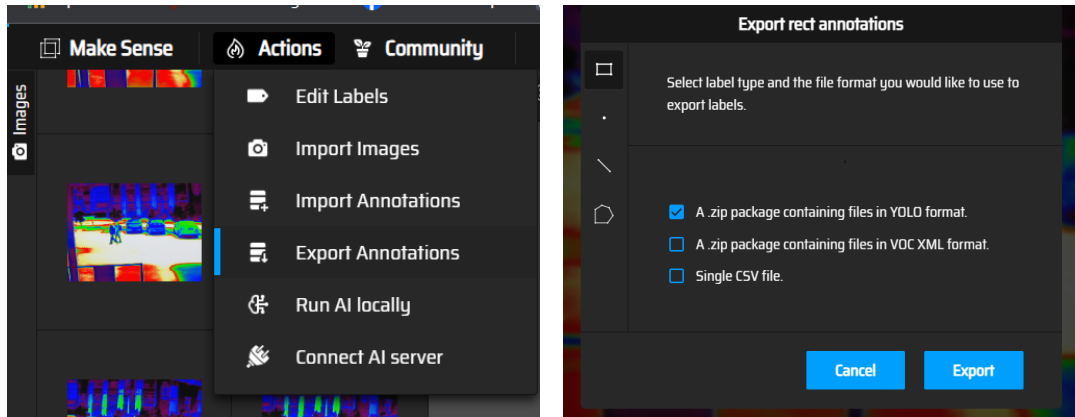


Figura 67. Exportación de las etiquetas.

Los archivos exportados se guardan en la carpeta de labels/train o labels/val, como se muestra en la Figura 68.

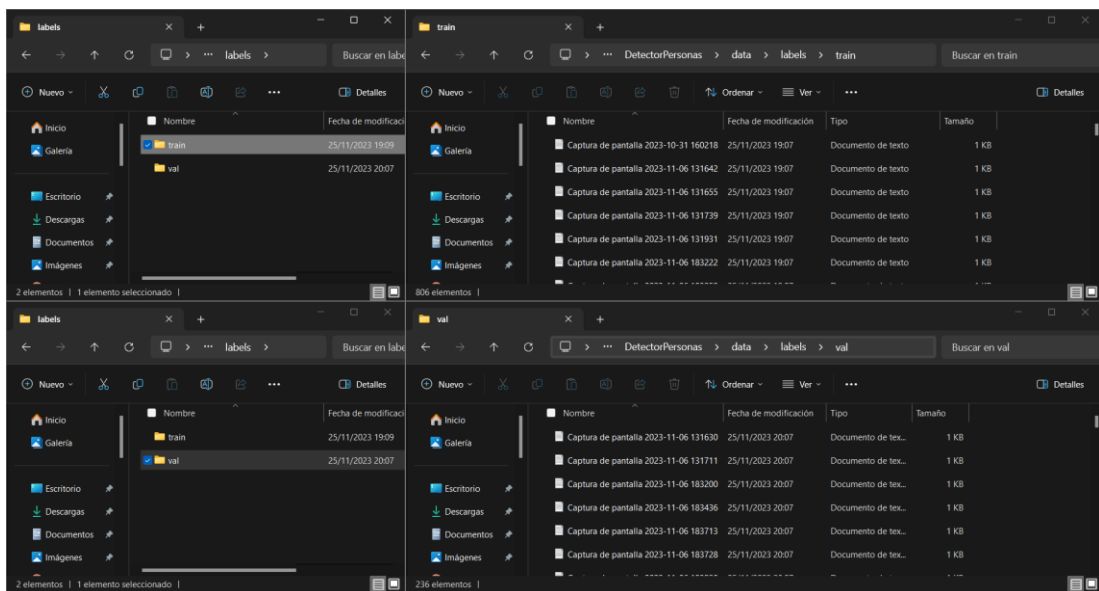


Figura 68. Distribución de las imágenes etiquetadas

c. *Entrenamiento de la red neuronal*

Para el entrenamiento de la red neuronal se utilizó Google Colaboratory o Colab que es un software que permite ejecutar código de Python en el navegador, utilizando la última versión de YOLOv8. Existen distintas versiones de YOLOv8, desde la versión nano hasta la versión grande, que se ilustra en la Figura 69. La versión de YOLOv8X presenta una mejor precisión, pero tiene latencia al momento de realizar las detecciones de hasta 479 milisegundos (ms).

Para este proyecto se optó por utilizar YOLOv8L debido a su velocidad de procesamiento de 375,2 ms, esta elección se basa en buscar un equilibrio entre precisión y tiempo de respuesta. En la Figura 69 se detallan las características de todos los modelos v8 [36].

Modelo	tamaño (píxeles)	mapa ^{valor} 50-95	Velocidad CPU ONNX (ms)	Velocidad A100 TensorRT (ms)	parámetros (M)	FLOP (B)
yolov8n	640	37,3	80,4	0,99	3,2	8,7
Yolov8s	640	44,9	128,4	1,20	11,2	28,6
YOLOv8m	640	50,2	234,7	1,83	25,9	78,9
yolov8l	640	52,9	375,2	2,39	43,7	165,2
yolov8x	640	53,9	479,1	3,53	68,2	257,8

Figura 69. Versiones de YOLOv8 [36].

- **Código para el modelo de entrenamiento.** Para iniciar con el entrenamiento de la red neuronal se ingresó a la página oficial de YOLOv8 y desde el lugar se abrió Google Colab para crear un nuevo cuaderno y ajustar el entorno de ejecución para utilizar la GPU en la pestaña de editar, como se muestra en la Figura 70.

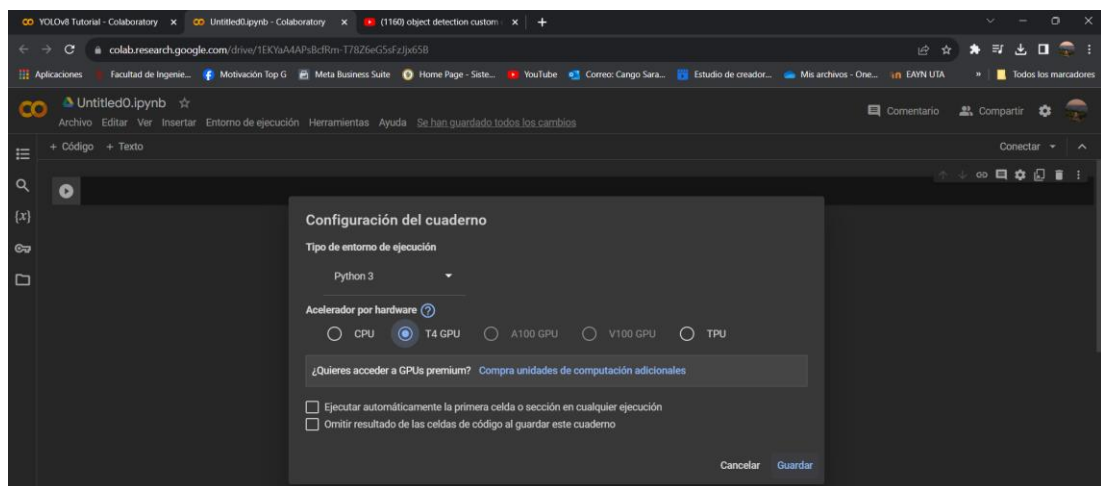


Figura 70. Entorno de ejecución de Google Colab

Posteriormente se cargó la carpeta de datos denominada “data” en formato zip en la sección de contenido denominado “content”, Figura 71.

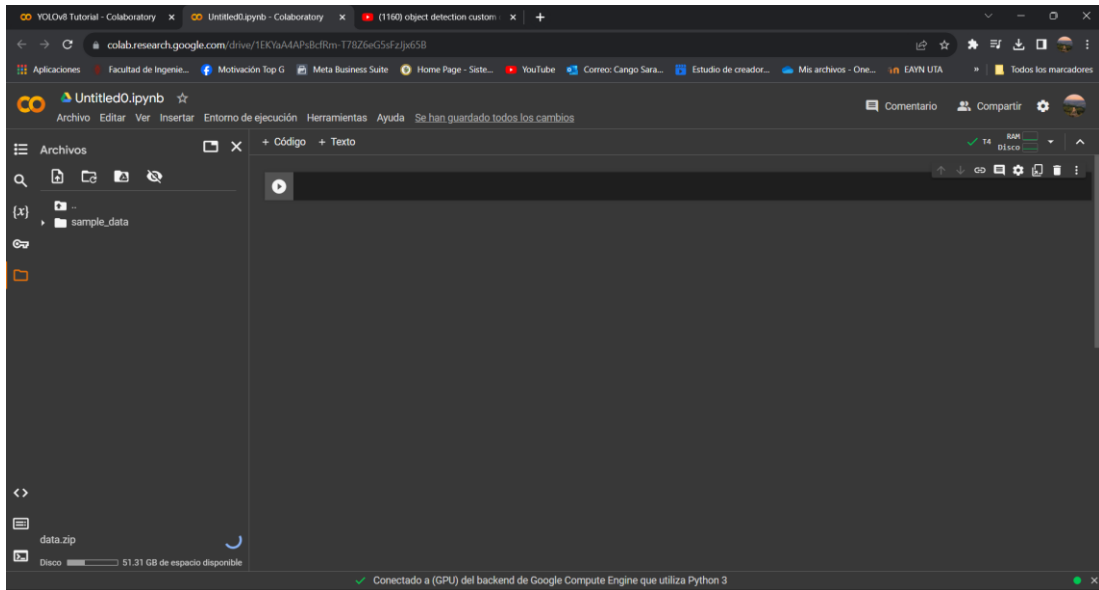


Figura 71. Interfaz gráfica de YOLOv8 y base de datos

Antes de empezar con el entrenamiento es necesario instalar las librerías de ultralytics el cual es una empresa que se centra en el desarrollo de software de código abierto para la investigación de inteligencia artificial, es por lo que se debe instalar todas sus librerías mediante el código presentado en la Figura 72.

```
!pip install ultralytics -q
```

654.0/654.0 kB 8.7 MB/s eta 0:00:00

Figura 72. Descargar librerías

Posteriormente, se crea una instancia de modelo en el que se descarga la versión de YOLOv8L desde ultralytics, Figura 73.

```
[2] from ultralytics import YOLO
```

```
model = YOLO("yolov8l.pt")
```

Downloading <https://github.com/ultralytics/assets/releases/download/v0.0.0/yolov8l.pt> to 'yolov8l.pt'... 100% | 83.7M/83.7M [00:00<00:00, 389MB/s]

Figura 73. Instancia de modelo

Luego, se descomprime la carpeta de data para ser utilizada, Figura 74.

```
!unzip -q /content/data.zip -d /content/
```

Figura 74. Descomprimir archivos.zip

Después se creó un archivo con extensión “.yaml” con la finalidad de seleccionar una ruta de los datos de entrenamiento y crear el número de clases, en este caso solamente para detección de personas, Figura 75.

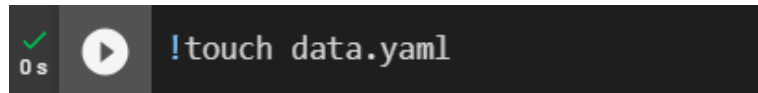


Figura 75. Archivo con extensión .yaml

Posteriormente se editó el archivo creado, seleccionándole las rutas de las carpetas “train” y “val” y también el número de clases que es los objetos que se van a detectar, en nuestro caso solamente para personas lo denominaremos “person” como se puede visualizar en la Figura 76.

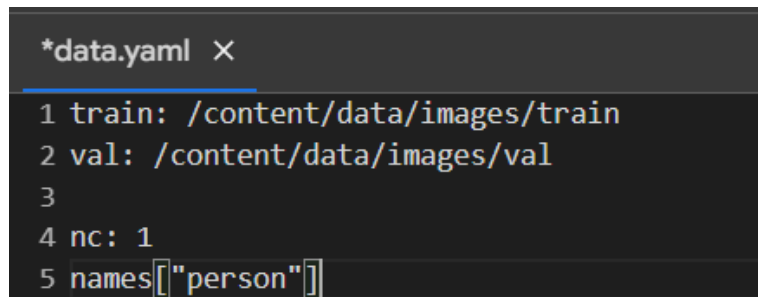


Figura 76. Selección de rutas

En la última línea de código se creó un modelo de entrenamiento con 80 épocas como se muestra en la Figura 77.

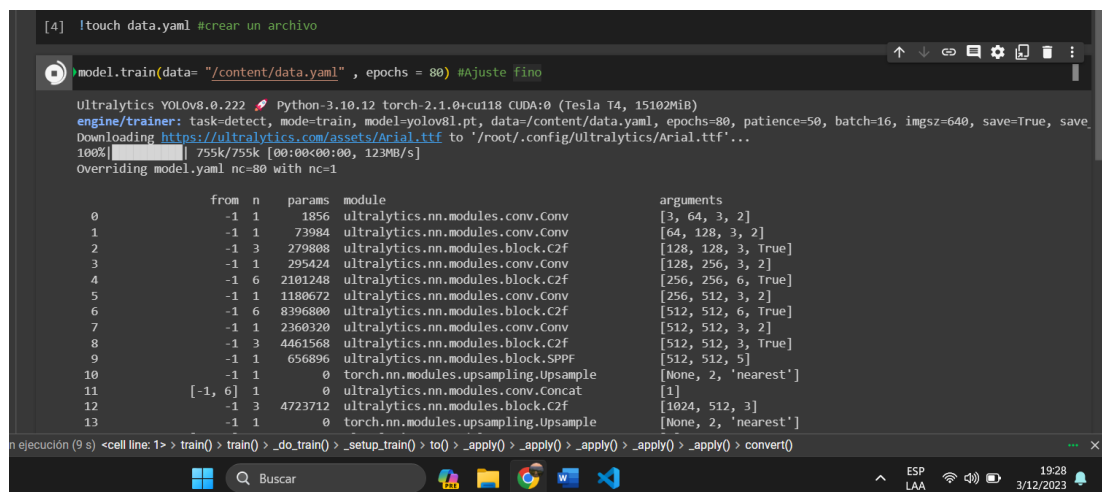


Figura 77. Código para el entrenamiento de la red neuronal

- **Resultados del modelo entrenado.** Después de 90 minutos se puede descargar el modelo entrenado que se encuentra en la pestaña de contenido en la carpeta de “runs” y se denomina “best.pt” ubicando en la ruta “content/runs/detect/train/weights”, como se puede visualizar en la Figura 78.

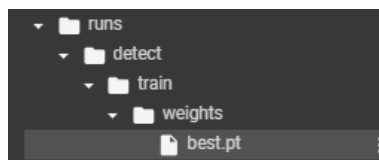


Figura 78. Carpeta de resultados de la red neuronal

En la Figura 79 y Figura 80 se presenta visualmente los resultados obtenidos al entrenar una red neuronal con distintos números de épocas utilizando YOLOv8l. Estas figuras muestran la representación gráfica de los resultados del entrenamiento en Google Colab.

Las métricas utilizadas son fundamentales para evaluar el modelo de detección de objetos, con la finalidad de que la segmentación se realice con precisión, se localice adecuadamente y su análisis sea eficiente, en la Tabla 30 se define la función que cumple cada métrica en un modelo entrenado.

Tabla 30. Funciones de las métricas de modelos entrenados

Métricas	Definición
Train/box_loss	Representa el error entre las coordenadas de la caja delimitadora o etiqueta y las coordenadas reales.
Trains/cls_loss	Mide la discrepancia entre las probabilidades de clase predichas y las etiquetas de clase reales para los objetos presentes en la imagen.
Train/dfl_loss	Está diseñado para optimizar la extracción y el aprendizaje de estas características densas, esto puede lograrse al enfocarse en maximizar la representación de detalles específicos, como la textura de la ropa, la forma de diferentes partes del cuerpo (cabeza, torso, extremidades, etc.).
Precisión	Esta métrica mide la precisión de las predicciones positivas frente a los falsos positivos con la siguiente ecuación, (verdades positivas / (verdades positivas + falsos positivos))
Recall	Evalúa la proporción de observaciones positivas predichas correctamente respecto a los positivos reales.
mPA50	Esta métrica analiza que tan bien se superponen las cajas delimitadoras predichas con las cajas delimitadoras reales
Mpa50-95	Esta métrica considera la precisión media a través de diferentes umbrales de superposición, desde el 50% hasta el 95%.

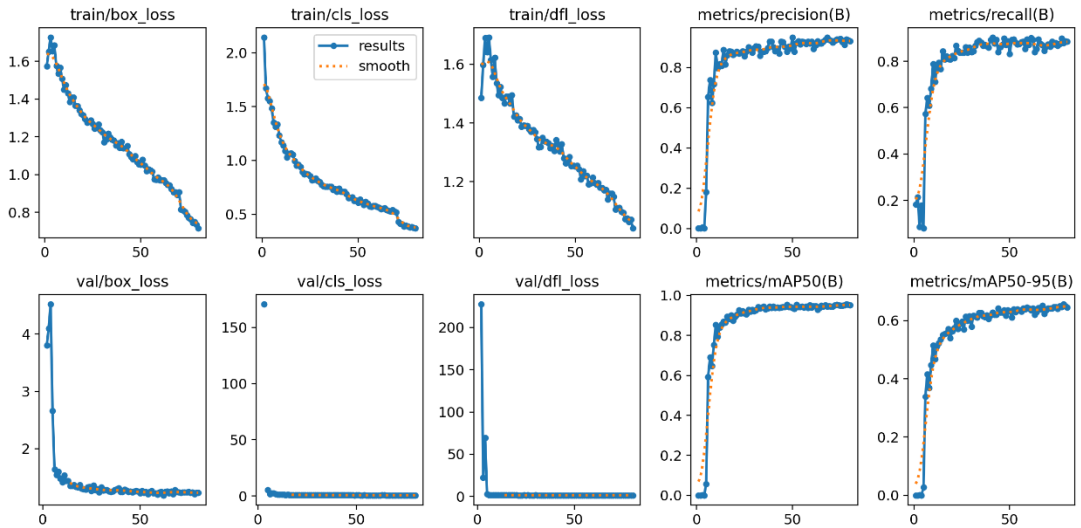


Figura 79. Resultados del entrenamiento de la red neuronal con YOLOv8l utilizando 80 épocas

Con el objetivo de mejorar el rendimiento, se realizó el entrenamiento de una red neuronal con 100 épocas en Google Colab. Esta extensión del entrenamiento tuvo como meta mejorar las métricas, buscando obtener resultados más sólidos y precisos.

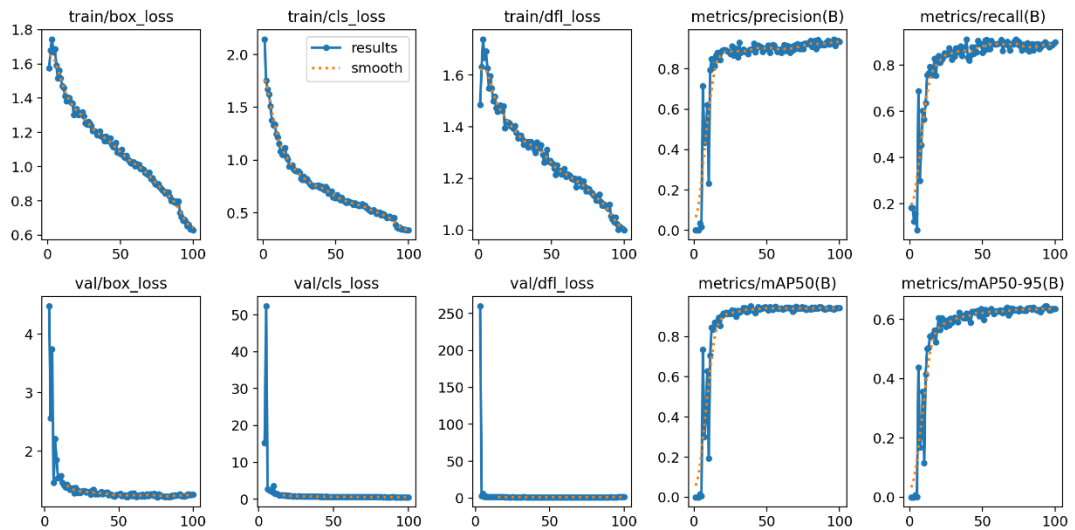


Figura 80. Resultados del entrenamiento de la red neuronal con YOLOv8l utilizando 100 épocas

En la Tabla 31 ofrece un análisis numérico de las redes neuronales entrenadas con 80 y 100 épocas. Los resultados muestran similitudes significativas entre ambas

configuraciones, concluyendo que la red neuronal apenas experimentó variaciones mínimas en las métricas evaluadas.

Tabla 31. Análisis de resultados de dos redes neuronales entrenadas con diferentes épocas

Métricas	YOLOv8l-80 ep	YOLOv8l-100 ep
Train/box_loss	0.71	0.63
Train/cls_loss	0.37	0.34
Train/dfi_loss	1.04	0.99
Precision	0.93	0.93
Recall	0.88	0.90
mPA50	0.95	0.94
mPA50-95	0.60	0.64

Train/box_loss, Train/cls_loss, Train/dfi_loss: Estas métricas representan las pérdidas durante el entrenamiento del modelo. En ambos modelos, existe una disminución en estas pérdidas a medida que se aumenta el número de épocas, indicando una mejora general en el ajuste del modelo a los datos.

Precision, Recall: Ambas métricas están relacionadas con la capacidad del modelo para detectar objetos correctamente. En este caso, ambos modelos tienen una alta precisión (cercana al 0.93) y un recall moderadamente alto (alrededor de 0.88-0.90). Esto sugiere que los modelos son capaces de identificar correctamente la mayoría de los objetos relevantes, pero podrían perder algunos.

mPA50, mPA50-95: Estas métricas miden el promedio de precisión para diferentes umbrales de IoU (Intersección sobre Unión, del inglés Intersection over Union). Un mPA50 de 0.95 en el modelo YOLOv8l-80 épocas indica una alta precisión cuando se acepta una superposición del 50% entre la predicción y la verdad fundamental. Sin embargo, hay una ligera disminución en el modelo YOLOv8l-100 épocas. Por otro lado, mPA50-95 es un promedio de precisión más estricto, considerando IoU de 50% a 95%. Aquí, ambos modelos muestran una mejora a medida que se incrementan las épocas de entrenamiento. En resumen, ambas variantes del modelo YOLOv8l muestran un rendimiento bastante bueno en la detección de objetos, con mejoras en algunas métricas a medida que se aumenta la cantidad de épocas de entrenamiento.

En la Tabla 32 se muestra el resultado de la red neuronal, mostrando las imágenes etiquetadas en el software Makesense.ai y la predicción que genera la red neuronal entrenada en YOLOv8l, cómo se puede visualizar todas las imágenes etiquetadas son predichas mediante la red neuronal ya que se tiene una precisión superior al 80%.

Tabla 32. Resultado del entrenamiento de la red neuronal con YOLOv8l 100 épocas

ep	Etiqueta				Predicción			
100 épocas								

ep	Etiqueta				Predicción			
80 épocas								

Después de entrenar la red neuronal con 100 épocas utilizando 1000 imágenes, se ha observado una ligera diferencia de al menos un 2% entre las etiquetas cargadas y las predichas. Aunque esta diferencia es mínima, es perceptible y sugiere una leve variación en la precisión del modelo entrenado.

d. Codificación de la visualización y detección de personas

Se ha desarrollado una codificación en Python para la visualización de la cámara térmica y los datos proporcionados por el UAV, esta implementación utiliza la biblioteca tkinter de Python para crear una interfaz gráfica básica. Su funcionalidad incluye la detección de personas en un flujo de video proveniente de la cámara 1 y la visualización de datos provenientes de la cámara 2.

- **Importación de librerías:** Para el desarrollo de la codificación es primordial importar todas las librerías necesarias para la captura de video, imágenes y modelos entrenados. En la Figura 81 se ilustra todas las librerías empleadas.

```
import tkinter as tk
import cv2
from PIL import Image, ImageTk
from ultralytics import YOLO
```

Figura 81. Librerías utilizadas

Cada librería tiene una función específica para el funcionamiento del sistema:

import tkinter as tk: Esta librería importa el módulo “tkinter” bajo el alias “tk”, este módulo permite desarrollar interfaces gráficas y botones en Python.

import cv2: Importa el módulo “cv2”, que es la biblioteca de OpenCV para procesamiento de imágenes y video.

from PIL import Image, ImageTk: Importa las clases Image y ImageTk del módulo PIL, que son utilizadas para manejar imágenes en Python.

from ultralytics import YOLO: Importa la clase YOLO del módulo ultralytics, que es un framework de detección de objetos.

- **Importación del modelo YOLO:** En Figura 82 se desarrolla la lectura del modelo entrenado en YOLOv8l, este código crea una instancia de la clase YOLO, cargando un modelo pre entrenado denominado “besto80.pt” o “best100.pt”.

```
model = YOLO("best80.pt")
```

Figura 82. Lectura del modelo entrenado

- **Portada:** El bloque de código de la Figura 83 es el encargado de mostrar una ventana de portada utilizando la biblioteca tkinter, esta interfaz consta de una imagen y un botón para iniciar una ventana en el que se visualiza la captura de las dos cámaras.

```
class Portada:
    def __init__(self, root):
        self.root = root
        self.root.title("Portada")

        # Cargamos la imagen denominada interfaz
        self.image = Image.open("Interfaz.jpg")
        self.image = self.image.resize((640, 520))

        self.photo = ImageTk.PhotoImage(self.image)
        self.label = tk.Label(self.root, image=self.photo)
        self.label.pack()

        # Botón para abrir la ventana principal
        self.open_button = tk.Button(self.root, text="Iniciar
localización", command=self.abrir_ventana_principal)
        self.open_button.pack()
```

Figura 83. Portada

def __init__ (self, root): En esta línea inicial se define una clase denominada portada que posee un método especial denominado “__init__” que se ejecuta de manera automática cuando se crea un objeto de esta clase. Toma un argumento “root” que será la ventana principal de la interfaz gráfica.

self.root = root: Se establece el atributo “root” de la instancia de la clase como la ventana principal.

self.root.title(“Portada”): Se asigna un título denominado portada a la ventana principal.

Self.image = Image.open("Interfaz.jpg"): Se carga una imagen desde donde se encuentra guardado el proyecto, en el caso de que la imagen se encuentre en otro lugar es necesario copiar su ruta.

Self.image = self.image.resize((640, 520)): Se ajusta la imagen al tamaño que se requiera.

Self.photo = ImageTk.PhotoImage(self.image): Se convierte la imagen cargada en un objeto "photoImage" de "tkinter" para que pueda ser utilizada en la interfaz gráfica.

self.label = tk.Label(self.root, image=self.photo): Ese crea un widget "Label" que muestra la imagen en la ventana principal "root" utilizando el objeto "PhotoImage" creado anteriormente.

self.label.pack(): Se empaqueta el widget para mostrarlo en la interfaz.

self.open_button = tk.Button(self.root, text="Iniciar localización" command=self.abrir_ventana_principal): Se crea un botón con el texto "Iniciar localización" que, al hacer clic, llamará al método `abrir_ventana_principal`. Este botón se empaqueta también para mostrarlo en la interfaz que se muestra en la Figura 84.



Figura 84. Interfaz con portada

El bloque de código de la Figura 85 muestra la segunda ventana denominada ventana principal que es donde se muestra el video capturados.

```
def abrir_ventana_principal(self):
    self.root.destroy() # Cierra la ventana de portada
    ventana_principal = tk.Tk()
    app = Videos(ventana_principal)
    ventana_principal.mainloop()
```

Figura 85. Cerrar la ventana

def abrir_ventana_principal(self): Se define una función llamada `abrir_ventana_principal` que pertenece a la clase `Portada`. Esta función se encargará de cerrar la ventana actual de la portada y abrir una nueva ventana principal.

self.root.destroy(): Esta línea cierra la ventana actual “`self.root`” de la portada, “`destroy()`” es un método de tkinter que se usa para cerrar ventanas o widgets.

ventana_principal = tk.Tk(): Se crea una ventana denominada “`ventana_principal`” utilizando tkinter.

app = Videos(ventana_principal): Se crea un objeto de la clase `videos`

ventana_principal.mainloop(): Finalmente, se inicia el bucle principal “`mainloop()`” de la ventana principal, lo que permite que la interfaz gráfica se ejecute y responda a las interacciones del usuario hasta que se cierre la ventana principal.

- **Captura del video:** En este bloque de código se realiza la captura de las dos cámaras que están colocadas en el UAV, Figura 86.

```
class Videos:
    def __init__(self, ventana_principal):
        self.ventana_principal = ventana_principal
        self.ventana_principal.title("Visualización de Cámaras")

        self.cap1 = cv2.VideoCapture(1)
        self.cap2 = cv2.VideoCapture(2)

        self.label_videoTIR = tk.Label(self.ventana_principal)
        self.label_videoTIR.pack(side=tk.LEFT)

        self.label_videoRGB = tk.Label(self.ventana_principal)
        self.label_videoRGB.pack(side=tk.RIGHT)
        self.mostrar_video()
```

Figura 86. Captura del video

Class Videos: Se define una clase denominada videos, ya que se utilizará para la captura de video de las dos cámaras.

def __init__(self, ventana_principal): Este es el método de inicialización de la clase, el cual toma dos argumentos: self (que hace referencia a la instancia de la clase) y ventana_principal que es la ventana en el que se visualizará las dos capturas de los videos.

self.ventana_principal = ventana_principal: Guarda la ventana principal pasada como argumento en un atributo llamado ventana_principal.

self.ventana_principal.title("Visualización de Cámaras"): Establece el título de la ventana principal como "Visualización de Cámaras".

self.cap1 = cv2.VideoCapture(1): Inicializa la captura de video de la cámara 1 que es la cámara térmica y guarda el objeto de captura en el atributo cap1.

self.cap2 = cv2.VideoCapture(2): Inicializa la captura de video de la cámara 2 que es la cámara RGB y guarda el objeto de captura en el atributo cap2.

self.label_videoTIR = tk.Label(self.ventana_principal): Crea un widget de etiqueta para mostrar el video de la cámara TIR y lo coloca en la ventana principal en el lado izquierdo.

self.label_videoRGB = tk.Label(self.ventana_principal): Crea otro widget de etiqueta para mostrar el video de la cámara RGB y lo coloca en la ventana principal en el lado derecho.

self.mostrar_video(): Llama al método "mostrar_video()" definido más adelante en la clase. Este método contiene la lógica para capturar imágenes de las cámaras y mostrarlas en las etiquetas de la interfaz gráfica.

- **Lectura del modelo:** El bloque de código presentado en la Figura 87, define una función denominada “deteccion_personas” que toma un fotograma “fps” como entrada y realiza la detección de personas utilizando el modelo YOLO cargado, los resultados se guardan en anotaciones.

```
def deteccion_personas(fps):
    resultados = model.predict(fps, imgsz=640)
    anotaciones = resultados[0].plot()
    return anotaciones
```

Figura 87. Segmentación del modelo

- **Visualización de los videos:** Se utiliza un método mostrar videos, el cual toma fotogramas de dos fuentes de video, en el bloque de código de la Figura 88 se realiza detecciones de personas utilizando la cámara 1 que posee imágenes térmicas y convierte los fotogramas para convertirlos en objetos de imagen Tkinter, todo ello en un bucle continuo con la finalidad de mostrar en tiempo real las cámaras.

```
def mostrar_video(self):
    ret1, fps1 = self.cap1.read()
    ret2, fps2 = self.cap2.read()

    if ret1 and ret2:
        fps1_rgb = cv2.cvtColor(fps1, cv2.COLOR_BGR2RGB)
        fps2_rgb = cv2.cvtColor(fps2, cv2.COLOR_BGR2RGB)

        anotaciones = self.deteccion_personas(fps1_rgb)
        img1 = Image.fromarray(anotaciones)
        img2 = Image.fromarray(fps2_rgb)

        img_tk1 = ImageTk.PhotoImage(img1)
        img_tk2 = ImageTk.PhotoImage(img2)

        self.label_videoTIR.imgtk = img_tk1
        self.label_videoRGB.config(image=img_tk1)

        self.label_videoTIR.imgtk = img_tk2
        self.label_videoRGB.config(image=img_tk2)

    self.ventana_principal.after(10, self.mostrar_video)
```

Figura 88. Visualización del video

def mostrar_video(self): Define el método mostrar_video dentro de la clase actual, toma self como su primer parámetro, lo que significa que opera en una instancia de la clase.

ret1, fps1 = self.cap1.read(): Lee un fotograma de la fuente de video representada por self.cap1 (asumiendo que cap1 es un objeto de captura de video de OpenCV). Devuelve un booleano “ret1” que indica si la lectura fue exitosa y el fotograma en sí “fps1”.

if ret1 and ret2: Verifica si ambos fotogramas de las fuentes de video fueron capturados exitosamente.

fps1_rgb = cv2.cvtColor(fps1, cv2.COLOR_BGR2RGB): Convierte el fotograma “fps1” del formato BGR al formato RGB utilizando la función cv2.cvtColor() de OpenCV.

anotaciones = self.deteccion_personas(fps1_rgb): Llama al método deteccion_personas() con el fotograma convertido fps1_rgb. Se realiza la detección de personas en este fotograma y devuelve una imagen con anotaciones.

img1 = Image.fromarray(anotaciones): Crea un objeto de imagen a partir de la matriz de píxeles anotaciones. Se asume que anotaciones es una matriz de píxeles que representa una imagen.

img2 = Image.fromarray(fps2_rgb): Crea un objeto de imagen a partir del fotograma RGB original “fps2_rgb”.

img_tk1 = ImageTk.PhotoImage(img1): Convierte el objeto de imagen img1 “anotaciones” en un formato compatible con Tkinter utilizando ImageTk.PhotoImage.

img_tk2 = ImageTk.PhotoImage(img2): Convierte el objeto de imagen img2 (fotograma RGB) en un formato compatible con Tkinter utilizando ImageTk.PhotoImage.

self.label_videoTIR.imgtk = img_tk1: Actualiza el atributo imgtk de la etiqueta self.label_videoTIR con la nueva imagen img_tk1.

self.label_videoRGB.config(image=img_tk1): Configura la imagen mostrada en la etiqueta “self.label_videoRGB” como “img_tk1”.

self.label_videoTIR.imgtk = img_tk2: Se actualiza el atributo imgtk de la etiqueta “self.label_videoTIR” con la nueva imagen img_tk2.

self.label_videoRGB.config(image=img_tk2): Configura la imagen mostrada en la etiqueta self.label_videoRGB como img_tk2.

self.ventana_principal.after(10, self.mostrar_video): Programa la ejecución del método “mostrar_video” después de 10 milisegundos utilizando after de Tkinter en la ventana principal “self.ventana_principal”. Esto crea un bucle continuo que actualiza la visualización de video cada 10 milisegundos, en la Figura 89 se puede visualizar la interfaz con las dos cámaras montadas en el UAV y en tiempo real.

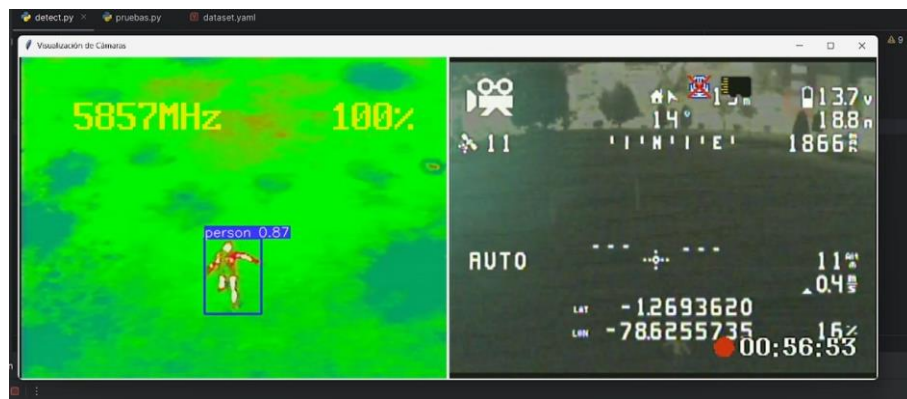


Figura 89. Ventana principal de visualización de videos

- **Ciclo infinito:** Este código crea una ventana denominada “ventana principal” usando Tkinter, crea una instancia de una clase de interfaz gráfica (Portada) pasándole esa ventana principal y luego inicia el bucle principal de eventos para mantener la interfaz gráfica abierta y funcional hasta que se cierre la ventana, Figura 90.

```
def main():
    root = tk.Tk()
    portada = Portada(root)
    root.mainloop()

if __name__ == "__main__":
    main()
```

Figura 90. Bucle de eventos

def main(): Define una función llamada main.

root = tk.Tk(): Crea una instancia de la clase Tk de Tkinter, que representa la ventana principal de la interfaz gráfica. Esta ventana es el contenedor principal para todos los elementos gráficos.

portada = Portada(root): Crea una instancia de la clase Portada, que contiene elementos gráficos (como botones, etiquetas, etc.).

root.mainloop(): Inicia el bucle principal de eventos de la interfaz gráfica de Tkinter. Esta línea es crucial ya que ejecuta un bucle infinito que está atento a los eventos de la interfaz gráfica (como clics de ratón, presiones de teclas, etc.).

if __name__ == "__main__": Esta línea verifica si el archivo Python se está ejecutando directamente, si es así, ejecuta la función “main()”. Esto es útil cuando se está desarrollando un programa y se quiere probar o ejecutar la lógica principal del programa cuando se ejecuta directamente como un script.



En el Anexo C se puede visualizar todo el código desarrollado en Python.





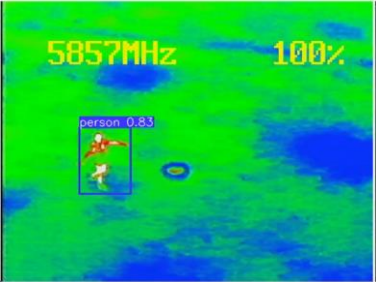

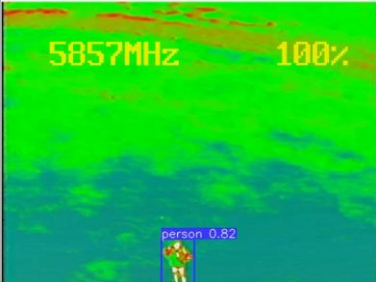

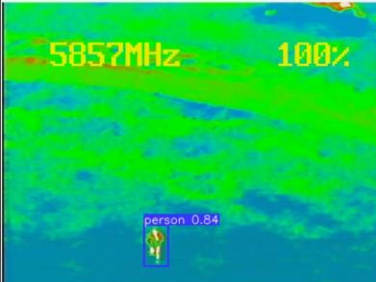

3.1.17 Resultados de los vuelos

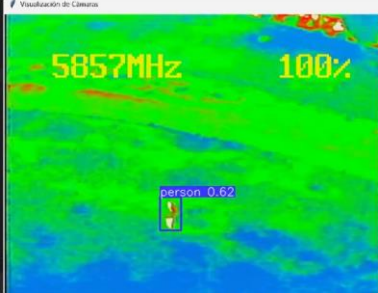

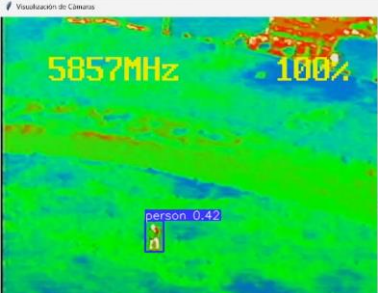

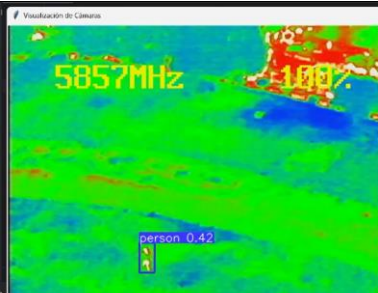

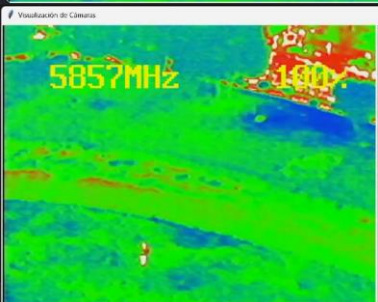

a. *Vuelo con una sola persona en el FOV*

En la Tabla 33 se ilustra sobrevuelos a diferentes alturas desde 4 metros hasta 20 metros con una sola persona en el FOV de la cámara, esta ruta de vuelo se realizó en vuelo automático a las 19:00.

Tabla 33. Sobrevuelo a diferentes alturas

Detección	Datos de localización
Altura: 4 metros	
	

Detección	Datos de localización
Altura: 6 metros	
	
Altura: 8 metros	
	
Altura: 10 metros	
	
Altura: 12 metros	
	
Altura: 14 metros	
	

Detección	Datos de localización
Altura: 16 metros	
	
Altura: 18 metros	
	
Altura: 20 metros	
	
	

El vuelo programado siguió una ruta predefinida y logró capturar detecciones de personas dentro del campo de visión (FOV) utilizando cámaras térmicas, las imágenes recolectadas durante el vuelo son similares a las empleadas para entrenar la red neuronal. En promedio, se obtuvo una precisión superior a 70 %, como se muestra en la Tabla 34. Sin embargo, se observó una disminución en la precisión a medida que el UAV aumentaba su altitud. Por ejemplo, a alturas de 14, 16, 18 y 20 metros, se registraron precisiones del 84 %, 62 %, 42 % y 42 % respectivamente.

Tabla 34. Precisión en la detección de personas

Altura	Precisión (%)
4	84
6	89
8	84
10	83
12	82
14	84
16	62
18	42
20	42
Promedio	72.44 %

Es importante destacar que, a una altura de 20 metros el UAV experimentó dificultades para detectar con precisión, puesto que en ocasiones el cuadro delimitador del objeto desaparecía en la imagen. Esta situación resultó en una reducción significativa de la precisión, comprometiendo la capacidad de detección del UAV a altitudes más elevadas, como se puede visualizar en la Figura 91 que desde los 16 metros la precisión empieza disminuir gradualmente.

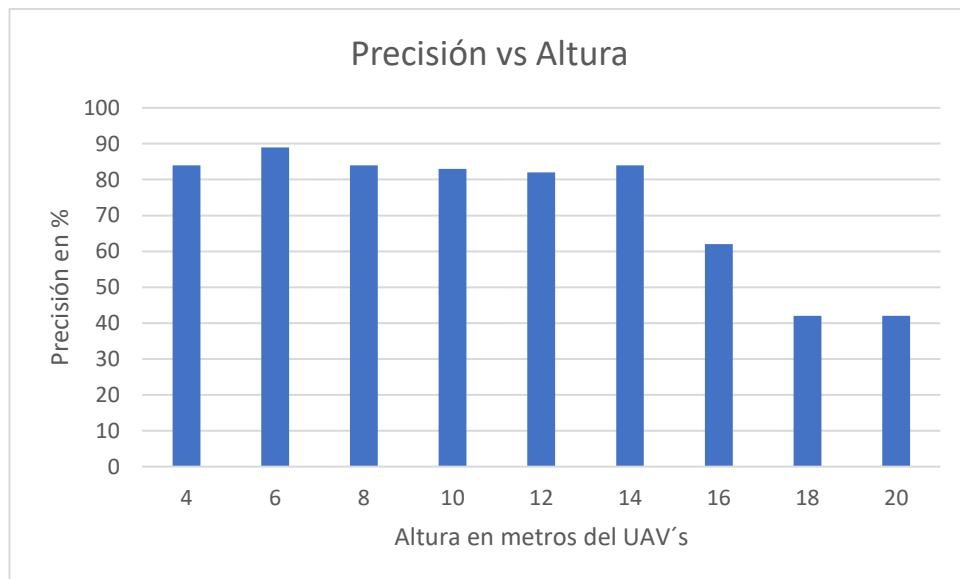


Figura 91. Precisión en la detección vs altura del UAV

b. Vuelo con varias personas en el FOV

En la Figura 92 se presenta un vuelo a una altura de 19 metros y en el campo de visión de la cámara se encuentra una multitud densa de estudiantes de la Universidad Técnica de Ambato, la cámara logra captar la radiación de cada estudiante con un color de

temperatura frío perteneciente al color verde, por el contrario, la detección utilizando la red neuronal entrenada no detecta a todas las personas debido a la altura.



Figura 92. Vuelo con varias personas en el FOV

Para realizar un análisis de la red neuronal se ocupará la métrica recall, el cual encuentra la proporción de observaciones positivas predichas correctamente respecto al total de predicciones positivas, para ello se utiliza la Ecuación 20. En la imagen térmica se encuentran 10 personas a las 18:00.

$$recall = \frac{\text{verdaderos positivos}}{\text{Verdaderos positivos} + \text{falsos negativos}} \quad (20)$$

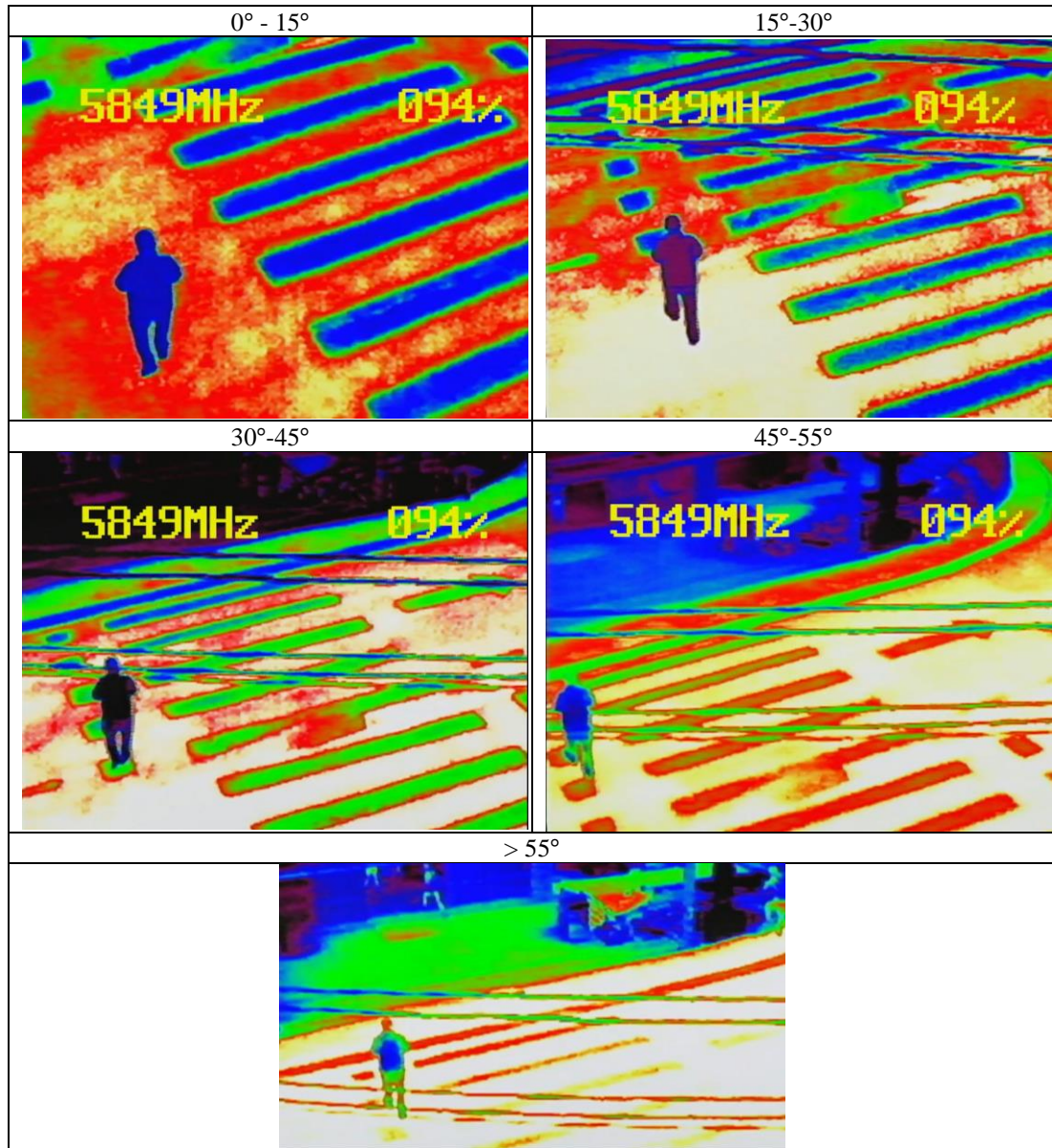
$$recall = \frac{5}{5 + 5} = 0.5 \%$$

Mediante el análisis de esta métrica se obtiene un porcentaje del 0.5%, es un porcentaje bajo debido a la altura del UAV, ya que para una detección eficiente se debe sobrevolar al vehículo a 16 metros de altura como se concluyó en la sección 3.1.13.

c. *Vuelo con diferentes ángulos*

Con el propósito de comprobar del ángulo de visión para capturar la firma térmica, se llevó a cabo la grabación de un vídeo en el que una persona se encuentra dentro del campo de visión de la cámara, variando los ángulos desde 0 hasta 45 grados, durante esta prueba, se observó que tanto la emisión como la reflexión muestran un impacto moderado, a pesar de la presencia del objeto sobre una superficie reflectante como el concreto. Al superar los 45 grados de ángulo de visión, se evidencia un aumento en la reflectividad y una disminución en la emisividad, como se visualiza en la Tabla 35.

Tabla 35. Imágenes capturadas a diferentes ángulos de visión



d. Localización utilizando Google Maps

Para la ubicación precisa de las personas detectadas, es fundamental obtener sus coordenadas de latitud y longitud. Con este fin, se llevó a cabo un vuelo a una altitud de 9 metros a las 18:00 en la Universidad Técnica de Ambato. Una vez se detectó la presencia de personas, se procedió a registrar los datos de ubicación: la latitud y longitud correspondientes a "-1.2694457, -78.6255818" respectivamente. Estos datos se extrajeron de la Figura 93.

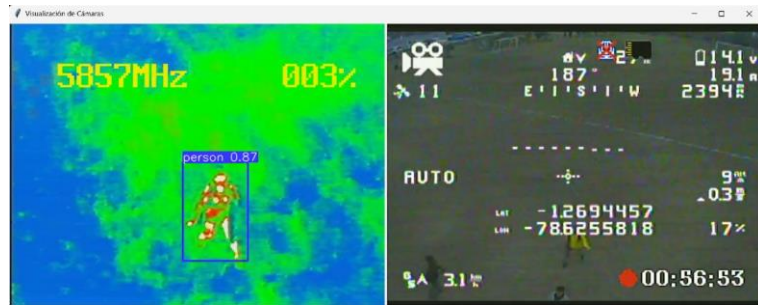


Figura 93. Detección y localización de personas

Para determinar la ubicación de la persona detectada dentro del FOV de la cámara, se requiere copiar las coordenadas y pegar en el buscador de Google, separadas por una coma que permite obtener la ubicación exacta, tal como se muestra en la Figura 94.

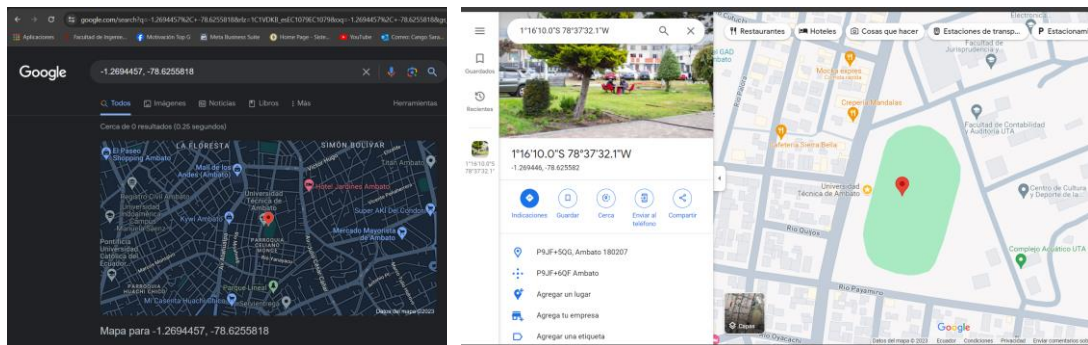


Figura 94. Localización en Google Maps

Una vez implementado el sistema de localización de personas en situaciones de catástrofe utilizando UAV's y cámaras térmicas, se establece que la altura óptima para la detección de personas es de 16 metros, a partir de esta altura, se observa una disminución en la precisión de la detección utilizando el modelo entrenado en YOLOv8ol. Asimismo, se determina que un ángulo de visión inferior a 45 grados es preferible para mantener un equilibrio entre la emisividad del cuerpo humano y la reflectividad del entorno.

Con los resultados obtenidos, es posible ampliar la aplicabilidad de este sistema en colaboración con entidades clave como los bomberos, la Fuerza Aérea Ecuatoriana (FAE) y la Policía Nacional. Esta colaboración podría involucrar la adaptación de estas tecnologías a las necesidades específicas de cada entidad, con el objetivo de crear un sistema robusto que tenga una amplia gama de aplicaciones, como seguridad, búsqueda y rescate.

CAPÍTULO IV.- CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones

- El sistema de localización de personas en situaciones de catástrofes mediante UAV's con cámaras térmicas destaca por su capacidad para detectar con precisión el calor humano en emergencias, utilizando una estación terrestre con una interfaz en Python que muestra imágenes térmicas, RGB y coordenadas, estas tecnologías permite la rápida identificación de individuos, facilitando su localización a través de herramientas como Google Maps, Earth o equipos de precisión como GPS, con la finalidad de mejorar la coordinación y eficiencia en operaciones de rescate.
- El prototipo resultó funcional para el desarrollo del sistema de localización de personas utilizando UAV's y cámaras térmicas. Con la puesta a punto de la cámara en tiempo real se sobrevoló el UAV en diferentes lugares, alturas y horario, buscando simular escenarios reales en el que se obtuvo más de 1000 imágenes y se entrenó dos redes neuronales en YOLOv8l, una con 80 épocas y otra con 100 épocas para la detección de personas, las cuales tuvieron una precisión superior al 90% con las imágenes recopiladas de los vuelos.
- Una estrategia clave para reducir la reflectividad de la radiación y mejorar la precisión en la captura de la firma térmica, fue mantener el ángulo de visión de la cámara térmica por debajo de los 45 grados. Esto asegura una captura más precisa del calor, minimizando los falsos positivos generados por materiales como piedras, asfalto y concreto, los cuales tienen una emisividad similar a 0.98, la misma que la del cuerpo humano.
- Para una detección exitosa se determinó una altura óptima de 16 metros, este parámetro se basa en el GSD de 4 cm por píxel lo que equivale a una resolución de 160 cm^2 en un área de 10 px, siendo este el mínimo para poder identificar una persona tanto en cámaras RGB como en térmicas, a medida que el UAV eleva su distancia la detección se vuelve deficiente, sin embargo, se ha

sobrevolado al vehículo a una altura de 20 metros en el que se obtuvo detecciones con un 50% de efectividad según la métrica recall.

4.2 Recomendaciones

- Para posteriores estudios se recomienda utilizar una batería de mayor capacidad a 1.5 mA con un índice de descarga menor a 100C, con la finalidad de prolongar la autonomía del vehículo en vuelo.
- Para vuelos más estabilizados se recomienda utilizar un hexacóptero o drones con 6 motores, integrados con un estabilizador de cámara con la finalidad de que la estabilización sea la más adecuada para las detecciones y localizaciones de las personas en situaciones de catástrofe.
- Debido a que se ha trabajado con equipos básicos y económicos, se recomienda utilizar antenas de al menos 7 dBi y transmisores de video con mayor alcance para evitar que la imagen se distorsione al realizar la transmisión en tiempo real desde el UAV en vuelo hasta la estación terrestre.
- Para que el sistema de captura de video y segmentación de objetos utilizando el modelo entrenado sea eficiente es necesario utilizar una computadora que disponga de una memoria dedicada de video con la finalidad de que no se ralentice el video en el computador.
- Para mejorar la versatilidad del UAV en diversas condiciones climáticas en catástrofes, se recomienda ajustar su diseño incluyendo mejoras como protecciones contra la humedad o tecnologías que aseguren la eficacia de vuelos estables en condiciones de lluvia intensa.
- Para llevar a cabo la geolocalización, se recomienda registrar la información de longitud y latitud en una base de datos, con la finalidad de establecer un sistema único de localización y evitar la necesidad de ingresar manualmente los datos previamente mencionados a una aplicación de mapas.

REFERENCIAS BIBLIOGRÁFICAS

- [1] E. Roldán, B. Elena, y F. Farbiarz, «El terremoto y posterior Tsunami del 26 de diciembre de 2004 en Indonesia», *Dyna (Medellin)*, 2004, [En línea]. Disponible en: <http://www.redalyc.org/articulo.oa?id=49614508>
- [2] Secretaría de Gestión de Riesgos, «Terremoto 7.8 - Pedernales», Pedernales, abr. 2016. Accedido: 4 de enero de 2024. [En línea]. Disponible en: <https://www.gestionderiesgos.gob.ec/wp-content/uploads/downloads/2016/05/Informe-de-situaci%C3%B3n-n%C2%B065-especial-16-05-20161.pdf>
- [3] F. Naranjo, «Agenda_Tungurahua», Ambato, 2019.
- [4] Y. Loaiza, «al menos 46.000 personas fueron afectadas por las fuertes lluvias», *INFOBAE*, 3 de abril de 2023. Accedido: 4 de junio de 2023. [En línea]. Disponible en: <https://www.infobae.com/america/america-latina/2023/04/03/ecuador-al-menos-46000-personas-fueron-afectadas-por-las-fuertes-lluvias/>
- [5] PRIMICIAS, «Deslave en Alausí: asciende a 53 la cifra de muertos», *Primicias*, Quito, 6 de mayo de 2023.
- [6] C. Burke *et al.*, «Requirements and limitations of thermal drones for effective search and rescue in marine and coastal areas», *Drones*, vol. 3, n.º 4. MDPI AG, pp. 1-14, 1 de diciembre de 2019. doi: 10.3390/drones3040078.
- [7] C. Burke, M. Rashman, S. Wich, A. Symons, C. Theron, y S. Longmore, «Optimizing observing strategies for monitoring animals using drone-mounted thermal infrared cameras», *Int J Remote Sens*, vol. 40, n.º 2, pp. 439-467, ene. 2019, doi: 10.1080/01431161.2018.1558372.
- [8] C. Rottondi, F. Malandrino, A. Bianco, C. F. Chiasserini, y I. Stavrakakis, «Scheduling of emergency tasks for multiservice UAVs in post-disaster scenarios», *Computer Networks*, vol. 184, ene. 2021, doi: 10.1016/j.comnet.2020.107644.
- [9] D. C. Schedl, I. Kurmi, y O. Bimber, «Search and Rescue with Airborne Optical Sectioning», *Cornell University*, sep. 2020, doi: 10.1038/s42256-020-00261-3.
- [10] I. Loaeza Rodríguez, «Robot aéreo seguidor de líneas usando visión artificial», Universidad Autónoma del Estado de Morelos, 2019. Accedido: 20 de mayo de 2023. [En línea]. Disponible en: <http://riaa.uaem.mx/handle/20.500.12055/3521>
- [11] B. Pazmiño, «Desarrollo de un sistema de identificación de personas con un drone mediante el uso de redes neuronales artificiales y visión artificial», Universidad Politécnica Salesiana, Quito, 2023. Accedido: 5 de junio de 2023. [En línea]. Disponible en: <http://dspace.ups.edu.ec/handle/123456789/24197>

- [12] C. Cuerno-Rejado, L. García-Hernández, A. Sánchez-Carmona, A. Carrio, J. L. Sanchez-Lopez, y P. Campoy, «Evolución histórica de los vehículos aéreos no tripulados hasta la actualidad», *Dyna (Spain)*, vol. 91, n.º 3, p. 332, may 2016, doi: 10.6036/7781.
- [13] VICOR, *A better way to power today's advanced UAV platforms*, vol. 1. 2010.
- [14] C. Sánchez, «Vehículos aéreos no tripulados: descripciones generales y aplicaciones», *ResearchGate*, 2017.
- [15] A. Roberto, «Drones de ala rotatoria: Tipos, características y precios».
- [16] P. Hu, W. Guo, S. C. Chapman, Y. Guo, y B. Zheng, «Pixel size of aerial imagery constrains the applications of unmanned aerial vehicle in crop breeding», *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 154, pp. 1-9, ago. 2019, doi: 10.1016/j.isprsjprs.2019.05.008.
- [17] M. López Escorial, «Drones de sangre», *Planeta futuro*. Accedido: 22 de mayo de 2023. [En línea]. Disponible en: https://elpais.com/elpais/2019/02/21/planeta_futuro/1550766718_183227.html
- [18] B. D. Song, K. Park, y J. Kim, «Persistent UAV delivery logistics: MILP formulation and efficient heuristic», *Comput Ind Eng*, vol. 120, pp. 418-428, jun. 2018, doi: 10.1016/j.cie.2018.05.013.
- [19] M. Zhang, H. Li, y G. Xia, «Research on the Application of Deep Learning Target Detection of Engineering Vehicles in the Patrol and Inspection for Military Optical Cable Lines by UAV», *IEEE Xplore*, abr. 2019, Accedido: 22 de mayo de 2023. [En línea]. Disponible en: <https://doi.org/10.1109/ISCID.2018.00029>
- [20] M. A. Boon, A. P. Drijfhout, y S. Tesfamichael, «Comparison of a fixed-wing and multi-rotor UAV for environmental mapping applications: A case study», en *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, International Society for Photogrammetry and Remote Sensing, ago. 2017, pp. 47-54. doi: 10.5194/isprs-archives-XLII-2-W6-47-2017.
- [21] K. Nizam, A. Ahmad, y W. Naim, «Aerial mapping using autonomous fixed-wing unmanned aerial vehicle», *IEEE Xplore*, 2012.
- [22] F. González, «Posicionamiento de un UAV mediante visión artificial infraroja», *Industriales*, sep. 2019.
- [23] M. Cañada y R. Royo, *Termografía infraroja Nivel II*, II., vol. II. Madrid: Fundación CONFEMETAL, 2016.
- [24] TECSA, «Cámaras térmicas».
- [25] TELEDYNE FLIR, «Your perfect palette».

- [26] S. Richard, *Computer Vision: Algorithms and Applications*, 2da ed. Washington: Springer, 2021. Accedido: 22 de octubre de 2023. [En línea]. Disponible en: <https://szeliski.org/Book/>
- [27] K. Havens y E. Sharp, «Thermal Imaging Techniques to Survey and Monitor Animals in The Wild: A Methodology», *ResearchGate*, vol. 1, p. 105, ene. 2016, Accedido: 22 de mayo de 2023. [En línea]. Disponible en: https://www.researchgate.net/publication/282341852_Thermal_Imaging_Techniques_to_Survey_and_Monitor_Animals_in_the_Wild_A_Methodology
- [28] V. E. Zuev, «Information-calculating system Spectroscopy of Atmospheric Gases. The structure and main functions», *ResearchGate*, 2005, Accedido: 4 de enero de 2024. [En línea]. Disponible en: <http://spectra.iao.ru>
- [29] Vision Aerial, «Distancia de muestreo del suelo, GSD», Montana, abr. 2021.
- [30] TranslatorsCafe, «Calculadora de batería LiPo para dron», may 2018.
- [31] G. Dean, «What are FPV drones?», SPACE. Accedido: 9 de junio de 2023. [En línea]. Disponible en: <https://www.space.com/what-are-fpv-drones>
- [32] A. Usama, M. Dora, M. Elhadidy, H. Khater, y O. Alkelany, «First Person View Drone-FPV», *IUGRC*, 2021, doi: 10.21608/IUGRC.2021.246400.
- [33] Prometec, «Guía FPV para multirrotores», Bogotá, 2019.
- [34] A. Arregui, «Detección de cascos de seguridad en tiempo real mediante modelos de Deep Learning», Universidad politécnica de Cartagena, Cartagena, 2023.
- [35] M. Sánchez Martínez y M. Sánchez, «Detección de personas mediante técnicas de aprendizaje automático: SVM Y CNN», 2018.
- [36] Ultralytics, «YOLOv8», Ultralytics. Accedido: 27 de diciembre de 2023. [En línea]. Disponible en: <https://docs.ultralytics.com/es/models/yolov8/>
- [37] M. Sánchez Martínez y M. Sánchez, «Detección de personas mediante aprendizaje automático: SVM Y CNN», Madrid, 2018.
- [38] Environmental Systems Research Institute, «Detección de personas», ARCGIS.
- [39] Jani Hirvinen, «ArduPilot Documentation». Accedido: 13 de junio de 2023. [En línea]. Disponible en: <https://ardupilot.org/ardupilot/index.html#>
- [40] F. L. Cabrero y J. García Herrero, «Análisis de datos LiDAR y navegación para altimetría mediante UAVS», 2019.
- [41] P. Chamba y C. Imaicela, «Sistema de comunicación para escenarios de emergencia y desastre usando un UAV y tecnologías de radio definido mediante software», Universidad Técnica Particular de Loja, Loja, 2018.
- [42] Tactial Security, «Detector Awareness de Radio Frecuencia», Quito, 2023.

- [43] MATEKSYS, «FLIGHT CONTROLLER F405-STD». Accedido: 18 de diciembre de 2023. [En línea]. Disponible en: <http://www.mateksys.com/?portfolio=f405-std#tab-id-3>
- [44] Taoglas, «Hoja de datos Taoglas», 2020. Accedido: 1 de diciembre de 2023. [En línea]. Disponible en: <https://www.mouser.ec/ProductDetail/Taoglas/CGGP.25.4.E.02?qs=GmNq1OH0n5S3R7U0xHuQag%3D%3D>
- [45] FLIR, «Electrical interface description document (IDD)», nov. 2011. Accedido: 4 de enero de 2024. [En línea]. Disponible en: <https://www.thermalvideo.com/46336006h-fpnlx.htm?search=tau&page=2>
- [46] Dirección general de aviación civil, «Operación de Aeronaves Pilotadas a Distancia (RPAs)», Quito, nov. 2020.

ANEXOS

Anexo A. Ficha de Cumplimiento

Ficha de cumplimiento				
Lugar:	Campo abierto, bosque con maleza, Personas entre escombros	Características		Observación
			# Personas:	4
Fotografías de respaldo		Hora:	9:00 a. m.	SC
			15:00 pm	SC
			19 pm	SC
		Posición de la persona	Caminando	SC
			Acostado	SC
			De rodillas	SC
			Alzando las manos	SC
		Altura del UAV	4 metros	SC
			8 metros	SC
			12 metros	SC
			16 metros	SC
20 metros	SC			

SC = Si cumple
 NC= No cumple



DIRECCIÓN GENERAL DE AVIACIÓN CIVIL

Operación de Aeronaves Pilotadas a Distancia (RPAs)

Anexo C. Código desarrollado en Python

```
import tkinter as tk
import cv2
from PIL import Image, ImageTk
from ultralytics import YOLO

# Lectura del modelo entrenado en YOLOv9l
model = YOLO("best80.pt")

class Portada:
    def __init__(self, root):
        self.root = root
        self.root.title("Portada")

        # En esta sección se carga la imagen denominada interfaz
        self.image = Image.open("Interfaz.jpg")
        self.image = self.image.resize((640, 520))

        self.photo = ImageTk.PhotoImage(self.image)
        self.label = tk.Label(self.root, image=self.photo)
        self.label.pack()

        # Botón para abrir la ventana principal
        self.open_button = tk.Button(self.root, text="Iniciar
localización"
, command=self.abrir_ventana_principal)
```

```

        self.open_button.pack()

    def abrir_ventana_principal(self):
        self.root.destroy() # Cierra la ventana de portada
        ventana_principal = tk.Tk()
        app = Videos(ventana_principal)
        ventana_principal.mainloop()

class Videos:
    def __init__(self, ventana_principal):
        self.ventana_principal = ventana_principal
        self.ventana_principal.title("Visualización de Cámaras")

        self.cap1 = cv2.VideoCapture(1)
        self.cap2 = cv2.VideoCapture(2)

        self.label_videoTIR = tk.Label(self.ventana_principal)
        self.label_videoTIR.pack(side=tk.LEFT)

        self.label_videoRGB = tk.Label(self.ventana_principal)
        self.label_videoRGB.pack(side=tk.RIGHT)

        self.mostrar_video()

    def deteccion_personas(fps):
        resultados = model.predict(fps, imgsz=640)
        anotaciones = resultados[0].plot()
        return anotaciones

    def mostrar_video(self):
        ret1, fps1 = self.cap1.read()
        ret2, fps2 = self.cap2.read()

        if ret1 and ret2:
            fps1_rgb = cv2.cvtColor(fps1, cv2.COLOR_BGR2RGB)
            fps2_rgb = cv2.cvtColor(fps2, cv2.COLOR_BGR2RGB)

            anotaciones = self.deteccion_personas(fps1_rgb)
            img1 = Image.fromarray(anotaciones)
            img2 = Image.fromarray(fps2_rgb)

            img_tk1 = ImageTk.PhotoImage(img1)
            img_tk2 = ImageTk.PhotoImage(img2)

            self.label_videoTIR.imgtk = img_tk1
            self.label_videoRGB.config(image=img_tk1)

            self.label_videoTIR.imgtk = img_tk2
            self.label_videoRGB.config(image=img_tk2)

        self.ventana_principal.after(10, self.mostrar_video)
def main():
    root = tk.Tk()
    portada = Portada(root)
    root.mainloop()

if __name__ == "__main__":
    main()

```