



# Universidad Autónoma de Querétaro

## Facultad de Ingeniería

### Ingeniería en Automatización

Tesis  
**Seguimiento de ruta para un dron.**

Que como parte de los requisitos para obtener el título de  
**Ingeniero en Automatización**

Presenta:  
**Enrique Iván Ramos Juárez**

Dirigido por:  
**Dr. Jesús Carlos Pedraza Ortega**

Dr. Jesús Carlos Pedraza Ortega

Director

\_\_\_\_\_  
Firma

Dr. Marco Antonio Aceves Fernández

Secretario

\_\_\_\_\_  
Firma

Ing. José Luis Avendaño Juárez

Vocal

\_\_\_\_\_  
Firma

Dr. Saul Tovar Arriaga

Sinodal

\_\_\_\_\_  
Firma

\_\_\_\_\_  
Firma

Dr. Manuel Toledano Ayala  
Director de la Facultad de Ingeniería

Centro Universitario, Querétaro, Qro.  
julio del 2023  
México



Dirección General de Bibliotecas y Servicios Digitales  
de Información



Seguimiento de ruta para un drone.

**por**

Enrique Iván Ramos Juárez

se distribuye bajo una [Licencia Creative Commons  
Atribución-NoComercial-SinDerivadas 4.0  
Internacional](https://creativecommons.org/licenses/by-nc-nd/4.0/).

**Clave RI:** IGLIN-244014

## DEDICATORIA

Me gustaría dedicar el resultado de este trabajo a una de las personas más importantes en mi vida, que a pesar de que ya no se encuentra físicamente conmigo, sé que siempre estará cuidándome donde quiera que este, a ti Mamá Sofí, que siempre me quisiste incondicionalmente, gracias por todo, por fin se logró y aunque me hubiera gustado que estuvieses compartiendo este momento tan importante en mi vida conmigo sé que estás orgullosa de mí, sabes que siempre tendrás un lugar en mi corazón, ya que sin ti ni Papá Pon, todo esto no habría sido posible, gracias ambos, esto es por ustedes y para ustedes.

También me gustaría dedicar este trabajo a toda mi familia; tíos y tías, padrinos y madrinas, primos y primas, que siempre me apoyaron, cuidaron, corrigieron, se preocuparon y me protegieron desde mi niñez y a lo largo de toda mi carrera. Pero principalmente me gustaría dedicar este trabajo a mis padres, que me apoyaron y estuvieron en todos los momentos malos y no tan malos. Gracias por enseñarme a afrontar las dificultades de la vida y sin perder nunca de vista lo importante de la vida, ya que sin ustedes no habría sido capaz de lograr lo que estoy logrando. Me han enseñado a ser el hombre que soy hoy, mis principios, mis valores, mi perseverancia y mi empeño y todo esto con una enorme dosis de amor y firme educación, sin pedir nada a cambio, porque para ustedes mi felicidad y éxito profesional es más que suficiente. Gracias a ustedes y a Dios por permitirme lograr esto. Dentro de mi familia también me gustaría mencionar y dedicar este trabajo a mi hermano menor que, ya que ha sido parte importante en mi vida y a pesar de yo ser el hermano mayor, él me ha enseñado muchísimo y más de que en algún momento llegue a imaginar.

## AGRADECIMIENTOS

Agradezco a mis padres, Sofía Mayela Juárez Gaytán y Enrique Ramos Arreguín; ustedes han sido y serán siempre el motor que impulsa mis sueños y esperanzas, quienes estuvieron siempre a mi lado sin importar nada, día y noche, durante todas mis horas de estudio y arduo trabajo. Son los mejores guías de vida que Dios me pudo dar y hoy que concluyo mis estudios, les agradezco profundamente a ustedes amados padres, por acompañarme a conseguir este logro tan importante de mi vida. Me siento orgulloso de ser su hijo y que estén compartiendo este momento tan importante para mí. Las palabras no me alcanzan para agradecerles por tanto padres míos.

Agradezco a mi tutor Juan Manuel Ramos Arreguín porque sin su paciencia, virtudes y constancia no hubiese logrado lo que he logrado hasta el día de hoy. Gracias por formar parte importante de esta historia, con sus aportes profesionales que lo caracterizan, por sus múltiples palabras de aliento, cuando más las necesite; por estar allí cuando mis dudas eran muchas y por sus orientaciones.

Del mismo modo quiero agradecer a mis profesores a lo largo de la carrera, gracias por sus palabras de sabiduría, sus conocimientos rigurosos, ya que gracias a ustedes profesores es que les debo mis conocimientos. A donde quiera que vaya los llevaré conmigo en mí transitar profesional. Me gustaría mencionar que la semilla de conocimiento que plantaron en mí, germino en alma y espíritu. Gracias por su paciencia y por compartir todos sus conocimientos de manera profesional e invaluable, por su dedicación, perseverancia y tolerancia.

También me gustaría agradecer a mis amigos, Dani, Sara, Rei, Gui, Chino, y todos mis compañeros que me acompañaron a lo largo de mi carrera con su amistad y nunca me dejaron solo ya fuese en el ámbito escolar o de amistad; ya que sin ustedes no habría sido posible esto y habría costado muchísimo más. Hoy culmina esta maravillosa aventura y no puedo dejar de recordar todos nuestros momentos increíbles, todas esas horas de desvelo para poder sacar algún proyecto. Hoy nos toca cerrar este capítulo maravilloso en esta historia de vida y no puedo dejar de agradecer por su apoyo incondicional y constancia, al estar en los momentos más difíciles. Gracias por estar siempre.

## RESUMEN

En este trabajo se aborda el tema en el que la tecnología ha venido evolucionando en los últimos años, de la manera en como ha sido aprovechada y empleada para beneficio del quehacer humano (científico); dando pauta a la búsqueda de la innovación y desarrollo de nuevas tecnologías.

Este tipo de tecnología, tal y como lo es el manejo de drones y las diversas aplicaciones que este puede tener para los humanos hoy en día, específicamente hablando de la detección de rutas en vuelo, por medio de una cámara y poder seguir una trayectoria de manera autónoma. En esto se enfoca este trabajo, donde se explica el desarrollo del proyecto el cual consiste en el desarrollo de un programa para un dron, que sea capaz de detectar una ruta establecida para el seguimiento de la misma de manera autónoma y se contiene una amplia explicación y abarca todo lo necesario para poder llevar acabo esta aplicación de la tecnología, desde la explicación y aplicación de los conceptos utilizados, las funciones y herramientas requeridas, bases para su desarrollo, su funcionamiento y aplicación funcional, tanto teórica como física dentro de la sociedad. Es por esto que es enfocado desde un punto de vista funcional y aplicable a la tecnología actual.

**Palabras clave:** Procesamiento de imágenes, autonomía, dron, detección, visión por computadora, seguimiento, control, análisis, filtros, software.

## **ABSTRACT**

This project talks about how technology has evolved over the years, how it is exploited and used for our benefit, comfort, health, and self-well-being, which gives a guideline to what is usually sought with the innovation and development of new technologies.

This type of technology, as is the handling of drones and the various applications that this can have for humans today, specifically talking about the detection of routes in flight, by means of a camera and being able to follow a trajectory autonomously. This project focuses on addressing the development of the project, which consists in the development of a program for a drone, which can detect a path stability for the follow-up of the same in an autonomous way. It contains an applicable explanation and covers everything necessary to be able to carry out this application of the technology, from the explanation and application of the concepts used, the functions and tools required, bases for its development, its operation and functional application, both theoretical and practice within the society. That is why it is focused from a functional point of view and applied to current needs and technologies.

**Keywords:** Image processing, autonomy, drone, detection, computer vision, monitoring, control, analysis, filters, software.

# ÍNDICE DE CONTENIDO

Capítulo 1. Introducción.....	1
1.1. Antecedentes.....	1
1.2. Descripción Del Problema .....	2
1.3. Justificación.....	2
1.4. Objetivos .....	3
1.4.1. Objetivo General.....	3
1.4.2. Objetivos Específicos .....	3
1.5. Estado del arte .....	4
Capítulo 2. Marco Teórico.....	5
2.1. <i>Procesamiento de imágenes.</i> .....	5
2.1.1. Visión por computadora.....	6
2.1.2. Píxel. ....	7
2.1.3. Imagen. ....	7
2.1.4. Iluminación. ....	8
2.1.5. Teledetección. ....	9
2.1.6. Color.....	10
2.1.7. Espacio de color RGB .....	11
2.1.8. Espacio de color HSI. ....	13
2.1.9. HSV.....	14
2.1.10. Histograma. ....	15
2.1.11. Umbralizacion (Thresholding). ....	16
2.2. Definición de una imagen digital. ....	17
2.2.1. Imagen digitalizada.....	18
2.3. Comunicación inalámbrica.....	19
2.4. Redes inalámbricas. ....	20
2.4.1. Según su cobertura. ....	20
2.4.2. Según su rango de frecuencia. ....	22
2.4.3. Tipos de conexiones inalámbricas principales. ....	23
2.5. Python. ....	24
2.5.1. OpenCV con python. ....	25
2.6. Librerías utilizadas en Python. ....	26
2.6.1. The Open Computer Vision Library (Open CV).....	26

2.6.2.	Librería NumPy.....	27
2.7.	Tecnología de drones.....	28
2.7.1.	¿Qué son los drones? .....	28
2.7.2.	¿Cómo funciona un dron? .....	29
2.7.3.	Uso de los drones.....	32
2.8.	Normas para el vuelo de drones.....	33
Capítulo 3.	Metodología.....	36
3.1.	Metodología.....	36
3.1.1.	Actualización de las librerías del Software para el desarrollo del código en Python. ....	36
3.1.2.	Análisis de la programación del dron. ....	37
3.1.3.	Pruebas de manejo y funcionamiento del dron desde la computadora. ....	38
3.1.4.	Identificación de la ruta con Python a través de la cámara del dron.....	39
3.1.5.	Integración y corrección de errores del vuelo junto con la detección de la ruta en el dron.....	39
3.1.6.	Seguimiento de la ruta por medio del dron de manera autónoma. ....	39
3.1.7.	Freno y descenso de manera autónoma.....	40
3.1.8.	Análisis de resultados.....	40
3.2.	Materiales.....	40
3.2.1.	Software y Hardware .....	40
Capítulo 4.	Resultados .....	43
4.1.	Proceso de programación.....	43
4.1.1.	Funciones y movimientos principales.....	43
4.1.2.	Transmisión de video y conexión del dron.....	43
4.2.	Explicación de los códigos más relevantes.....	44
4.2.1.	Desplazar el dron. ....	44
4.2.2.	Suspender el dron en el aire.....	44
4.2.3.	Detección de la ruta.....	45
4.2.4.	Obtención de los contornos de la ruta.....	46
4.2.5.	Descenso del dron al suelo.....	47
4.3.	Explicación de la técnica para identificación de la ruta. ....	48
Capítulo 5.	Conclusiones .....	51
Referencias	.....	53
ANEXOS	.....	58

# ÍNDICE DE FIGURAS

<b>Figura 1.</b> Imagen tomada con la cámara del dron. ....	5
<b>Figura 2.</b> Procesamiento de imágenes y segmentación donde se pretende separar el objeto de análisis del resto de la imagen. ....	6
<b>Figura 3.</b> Imagen representativa donde se pueden apreciar los píxeles (Oltra, G. N., & Mellado, J. R. A., 2008). ....	7
<b>Figura 4.</b> Imagen tomada con ayuda del dron que se usa para este trabajo. ....	7
<b>Figura 5.</b> (a) Representación de la escena con luz directa. (b) Representación de la escena con luz difusa (Ospina, J. C. M., Ortiz, F. A. P., & Bedoya, J. W. B., 2007). ....	8
<b>Figura 6.</b> Diferencia de iluminación en la misma imagen. (a) Imagen con iluminación directa (b) Imagen con iluminación difusa. ....	9
<b>Figura 7.</b> Imagen capturada con el dron que se usa para este trabajo la cual hace referencia a la teledetección. ....	10
<b>Figura 8.</b> Imagen representativa referente al color (Alegria, E. et al. 2016). ....	10
<b>Figura 9.</b> Modelo de color RGB (Marcial, Jorge A., 2011). ....	12
<b>Figura 10.</b> Espacio de colores RGB representado en forma de cubo (Alegria, E. et al. 2016). ....	12
<b>Figura 11.</b> Espacio de color HSI el cual se representa en forma de bipirámide hexagonal (Alegria, E. et al. 2016). ....	13
<b>Figura 12.</b> Modelo de color HSV (Alegria, E. et al. 2016). ....	14
<b>Figura 13.</b> Espacio de color HSV que se representa en forma de cono invertido. (Alegria, E. et al. 2016). ....	15
<b>Figura 14.</b> Imagen donde la luz se reparte uniformemente y se ve un histograma amplio. ....	16
<b>Figura 15.</b> Imagen donde la luz no está bien repartida, es decir falta de exposición. ....	16
<b>Figura 16.</b> Umbralización de una hoja de papel. ....	17
<b>Figura 17.</b> Ejemplo de imágenes digitales generadas por programación. (Oltra, G. N., & Mellado, J. R. A., 2008). ....	18
<b>Figura 18.</b> Imagen digitalizada. (Oltra, G. N., & Mellado, J. R. A., 2008). ....	19
<b>Figura 19.</b> Imagen representativa de la comunicación inalámbrica (Blázquez, J. P., 2015). ....	20
<b>Figura 20.</b> Tipos de redes inalámbricas (Coñapes, S. y Cerro, Y., 2015). ....	22
<b>Figura 21.</b> Representación visual del efecto electromagnético. (Clark, Bryan y Walton, Timothy; 2020). ....	23
<b>Figura 22.</b> Estructura de la librería OpenCV. ....	27
<b>Figura 23.</b> Imagen de cómo debe de girar un dron para mantener un momento angular cero. ....	29
<b>Figura 24.</b> Diferencia de velocidad entre los motores para desplazar el dron hacia la derecha. ....	30
<b>Figura 25.</b> Diferencia de velocidad entre los motores para desplazar el dron hacia la izquierda. ....	30
<b>Figura 26.</b> Diferencia de velocidad entre los motores delanteros y traseros para desplazar el dron hacia delante. ....	31
<b>Figura 27.</b> Diferencia de velocidad entre los motores delanteros y traseros para desplazar el dron hacia atrás. ....	31

<b>Figura 28.</b> Diferencia de velocidad entre los motores que giran en sentido de las manecillas de reloj y sentido contrario rotar el dron en sentido de las manecillas. ....	32
<b>Figura 29.</b> Diferencia de velocidad entre los motores que giran en sentido de las manecillas de reloj y sentido contrario rotar el dron en sentido contrario de las manecillas. ....	32
<b>Figura 30.</b> Diagrama de flujo de la metodología. ....	36
<b>Figura 31.</b> Versiones de las librerías y de Python utilizadas. ....	37
<b>Figura 32.</b> información acerca del paquete “djitelopy”. ....	38
<b>Figura 33.</b> Imagen de la pieza montada en el dron ....	38
<b>Figura 34.</b> Imagen tomada con el dron de la detección de la ruta. ....	40
<b>Figura 35.</b> Imagen de la simulación de la pieza que se fabricó para una visión hacia el piso por parte de la cámara del dron. ....	41
<b>Figura 36.</b> Imagen de la pieza impresa en 3D. ....	41
<b>Figura 37.</b> Jaula protectora para el dron Tello. ....	42
<b>Figura 38.</b> Imagen del dron que se utilizó para este trabajo. ....	42
<b>Figura 39.</b> Estructura para enviar la transmisión de video y conexión con el dron. ....	43
<b>Figura 40.</b> Estructura que se usa para desplazar en dron. ....	44
<b>Figura 41.</b> Código para la obtención de los valores HSV. ....	45
<b>Figura 42.</b> Umbralización la ruta para distinguirla. ....	46
<b>Figura 43.</b> Imagen de la ruta umbralizada capturada por el dron. ....	46
<b>Figura 44.</b> Función para marcar el contorno de la ruta. ....	47
<b>Figura 45.</b> Imagen de la ruta original y los contornos de la detección. ....	47
<b>Figura 46.</b> Estructura que se utiliza para aterrizar el dron. ....	48
<b>Figura 47.</b> Imagen de una detección de un elemento externo a la ruta. ....	49
<b>Figura 48.</b> Imagen del dron cuando la ruta no es recta. ....	49
<b>Figura 49.</b> Comando utilizado para disminuir o aumentar la velocidad del dron en cm/s. ....	50
<b>Figura 50.</b> Imagen del dron detectando y trasladándose sobre la ruta. ....	50
<b>Figura 51.</b> Imagen de la portada del artículo. ....	58

# Capítulo 1. Introducción

## 1.1. Antecedentes

En el año de 1907, los hermanos Jacques y Louis Bréguet crearon se creó el primer dron, este fue un diseño un tanto limitado, porque requería de cuatro hombres para poder estabilizarlo. Más de 100 años después, en el 2010, una compañía de origen francés creó el primer dron controlado por un teléfono inteligente conectado a través de Wi-Fi (Group, B., 2019). Desde ese entonces el auge de los drones se ha incrementado considerablemente, ya que ahora cuentan con muchas más aplicaciones que hace 10 o 15 años en diversos campos. Esto incluye la detección de obstáculos de manera autónoma, mediante técnicas de visión por computadora. Esto nos sirve tanto para el transporte de objetos pequeños, así como para llegar a lugares donde los vehículos terrestres y aéreos no pueden llegar.

De acuerdo a Calderón Jácome, los cuadricópteros, también conocidos como helicóptero, quadrotor o Drones, son tecnologías que están adquiriendo mucha fuerza en distintas aplicaciones, ya que son vehículos aéreos manejados por sistemas remotos de control de forma autónoma siguiendo una actividad pre programada o controlados manualmente desde tierra. Estos pueden llevar realizar tareas o actividades que los vuelos tripulados podrían considerar arriesgadas, debido a su precisión, sigilo y características de cautela. (*Calderón Jácome, M. T., & Maldonado Andrade, D. J., 2014*).

Técnicamente se puede considerar como un helicóptero que es elevado y propulsado por cuatro rotores. Estos se clasifican como helicópteros de cuatro hélices, contrario a las aeronaves de ala fija, debido a que la elevación es generada por perfiles aerodinámicos giratorios de acorde estrechos. (*Calderón, J. et al. 2014*).

En el 2016, (*Vergouw, B. et al. 2016*) realizaron un artículo donde hablan de cómo se pueden diferenciar los drones dependiendo de su aplicación y utilidad. También mencionan las cargas útiles que puede llevar un dron, tipos de sensores y la comunicación inalámbrica para el control del dron. También se analizan las cuestiones legales sobre el uso del espectro de frecuencias y los equipos electrónicos. Se habla del alcance realista de los drones y como es que serán cada vez más autónomos (*Vergouw, B., et al., 2016*).

En el trabajo de Guillen, G., se habla de lo que se puede hacer con la librería de código abierto OpenCV para Python, C++ y Java. También se menciona la amplia gama de aplicaciones, como el procesamiento de imágenes médicas, videos de vigilancia, el seguimiento de objetos en movimiento, la detección de objetos, entre otras cosas. (*Guillen, G., 2019*).

En 2019, Guan, Y., Zhou, F., & Zhou, J., mencionan la importancia del procesamiento de una imagen digital, y también se analizan las características de la aplicación de Python y se ponen en práctica las funciones del procesamiento de

imágenes. El procesamiento de imágenes y el procesamiento de gráficos siempre han sido campos importantes en el desarrollo de la tecnología informática (*Guan, Y., Zhou, F., & Zhou, J., 2019*).

En 2017, Santos-Castro, A., presentan una descripción de lo que son los drones para la IEEE, la cual describe a los drones como una de las 11 nuevas tecnologías más importantes de la década y de cómo ha crecido la industria global en los últimos años, debido a la gran cantidad de aplicaciones tanto recreativas como para misiones de rescate en desastres naturales, así como para monitoreo de agricultura y seguridad. (*Santos-Castro, A., 2017*).

En el artículo de Alcón se habla acerca del manejo de drones Tello. Cuenta con tres módulos, los cuales son: simulador, visualizador y calculador de precisión, lo cual es una base importante para el desarrollo de la programación de este modelo de drones (*Alcón Clemente, J., 2019*).

Los drones tienen un amplio campo de desarrollo, así como de aplicaciones que van desde entretenimiento, hasta aplicaciones de investigación. En todo este desarrollo, una característica muy importante es el procesamiento de imágenes y la detección de objetos u obstáculos. Aprovechando estas características, se propone este trabajo de tesis, para programar un dron que tenga la capacidad de seguir una trayectoria, evadiendo obstáculos en el camino.

## **1.2. Descripción Del Problema**

Los drones son considerados actualmente como una parte del desarrollo tecnológico global importante a partir del 2017 (*Santos-Castro, A., 2017*). Uno de los aspectos que se deben trabajar en ese tipo de sistemas, es la autonomía de vuelo.

En este proyecto se va a trabajar con la programación de un dron, para que sea capaz de seguir una ruta específica, y en el caso de detectar un obstáculo, detenga su movimiento. Para esto, se requiere obtener imágenes de la cámara del dron, y se van a analizar usando lenguaje Python, para detectar la ruta, así como un obstáculo en el camino.

## **1.3. Justificación**

En la actualidad los drones son una tecnología que está al alcance de muchas personas y facilita las actividades del día a día. Así mismo, los drones son considerados como una tecnología importante en el mercado global (*Santos-Castro, A., 2017*). Una tendencia que tiene importancia en el desarrollo de la tecnología de drones es el vuelo autónomo. La complejidad que esto implica es importante, debido

a que se debe considerar que el dron se mueve en un espacio de 3 dimensiones. Sin embargo, para los drones no existe tanta información accesible de este tipo de tecnología, ya que la mayoría de los drones son de arquitectura cerrada y esto nos limita a poder programarlos para hacer una actividad de manera autónoma, previamente programada.

Existe la posibilidad de armar un dron con arquitectura abierta, pero puede resultar complejo al momento de escoger los componentes a usar, debido a que las partes se adquieren por separado.

Para que el dron se pueda mover de manera autónoma, es necesario el desarrollo de software para que el dron pueda seguir una ruta específica sin chocar durante su vuelo. Esto implica que el dron tenga la capacidad de reconocer obstáculos, y tomar decisiones evasivas, como puede ser cambio de ruta o detenerse. Sin embargo, se requiere de la programación de sistemas embebidos para realizar el control de este tipo de sistemas. Por lo que se considera pertinente e importante para la línea terminal de electrónica y sistemas embebidos.

Programar un dron para que pueda adquirir imágenes, procesarlas y tomar alguna acción, es la principal tarea que se deberá realizar en este proyecto. Para el desarrollo de este trabajo, se planea usar un dron que facilite la programación de las tareas a realizar, usando una interfaz en Python. Se selecciona la marca de drones TELLO por el costo y las herramientas que proporciona para su programación.

### **1.4. Objetivos**

#### ***1.4.1. Objetivo General***

Mover un dron a lo largo de una trayectoria predefinida, utilizando lenguajes de programación Open Source, así como técnicas de procesamiento de imágenes para la detección de la ruta sobre la cual se desplazará el dron.

#### ***1.4.2. Objetivos Específicos***

- Revisar diferentes tecnologías de drones para seleccionar uno que pueda ser programado por el usuario, usando información de los fabricantes.
- Estudiar la manera de programar el dron, usando la información técnica disponible, para poder comunicarse con el equipo.
- Aprender a programar en el lenguaje requerido por el dron, para desarrollar la aplicación necesaria, usando la información técnica.

- Desarrollar el código en el lenguaje de programación a trabajar, el reconocimiento de la ruta y que siga la misma ruta de manera autónoma.

## 1.5. Estado del arte

En las últimas décadas, el manejo y vuelo de drones se ha convertido en una de las actividades de mayor interés para las muchas industrias tecnológicas, ya que se han encontrado múltiples aplicaciones para su uso. (*Sánchez Otálora, N. A., 2018*). Es por esto que a lo largo de los años se ha trabajado con la tecnología de drones para la detección de objetos, rutas, personas, reconstrucción de diversos objetos, mapeo de ciudades, levantamiento de estructuras e incluso seguimiento de trayectorias de manera autónoma, es decir, sin la necesidad de ser piloteadas por un dron.

En la actualidad, la tecnología de drones no requiere de combustibles para su manejo, tampoco ponen en peligro las vidas de quienes desarrollan las actividades que un dron podría llevar a cabo, además de que ahorran muchísimo tiempo en estas actividades, llevando consigo resultados precisos y eficaces (*Yoza, M. M. B., Zuñiga, K. M., Zambrano, S. M. Z., & Pionce, W. R. C., 2021*).

Gracias a que la tecnología en drones ha tenido un gran auge tanto comercial como profesional e incluso militar (*Pino, E., 2019*), se ha implementado el desarrollo de un programa para la detección autónoma de una ruta por medio de un dron, esto debido a que es una tecnología que tiene una gran aceptación y un amplio desarrollo tanto científico como tecnológico que, si bien ya se ha trabajado, se busca una reducción económica, un acceso y un manejo de la tecnología con drones con mayor facilidad.

Como dice (*Luque López, B., 2016*) en su trabajo final de grado, las aplicaciones de visión por computadora aplicada a los drones, son una tecnología que cada vez ha adquirido más popularidad con el paso de los años. En el trabajo de Luque López, se implementa un sistema que detecta carreteras y vehículos a partir de imágenes adquiridas por drones, partiendo del sistema SegNet (Sistema desarrollado en la Universidad de Cambridge) que, por medio de técnicas de deep learning, logra segmentar semánticamente los píxeles que tiene una imagen en tiempo real desde la perspectiva de un conductor de automóvil. Además de que se proponen aplicaciones para el desarrollo de este tipo de tecnologías, como evaluación del estado de las carreteras, el análisis de tráfico o seguimiento de vehículos.

# Capítulo 2. Marco Teórico

## 2.1. *Procesamiento de imágenes.*

Procesar información es un tema importante dentro de la ciencia y tecnología, y algo que se distingue mucho en las ingenierías. Esto debido a que para el diseño, desarrollo, implementación y realización de procesos se necesita generar información con el propósito de tomar decisiones a partir de una base de datos. Analizar procesos y sistemas se basan en datos obtenidos por medio de sensores que son extremadamente modernos y sofisticados, hablando de las diferentes variables que intervienen en este tipo de procesos. En la actualidad, el procesamiento digital de imágenes se considera que tiene un amplio campo de desarrollo, el cual forma parte de la vida diaria. Al principio su uso se limitaba a una cierta área de investigación básica como las matemáticas, astronomía, ciencias de la computación, entre otras. En la Figura 1 se muestra una imagen tomada con un dron (Cuevas, et al., 2016).



**Figura 1.** Imagen tomada con la cámara del dron.

La imagen puede ser generada de distintas formas, como de manera fotográfica o electrónica, por medio de televisores, entre otras. Por otro lado, las imágenes pueden ser procesadas usando métodos ópticos, o se puede usar una computadora para aplicar métodos digitales de procesamiento de imágenes (Malacara Hernández, D., 1997).

Jean-Baptiste-Joseph Fourier (1768-1830) fue un matemático y físico francés, quien desarrolló trabajos relacionados con la descomposición de funciones periódicas en series trigonométricas convergentes, también conocidas como Series de Fourier. Desde el principio la teoría de Fourier se consideró importante, se decía que el teorema de Fourier no solamente era uno de los mejores resultados del análisis moderno, sino que además proporciona una herramienta fundamental en el tratamiento de la mayoría de los problemas de la física moderna.

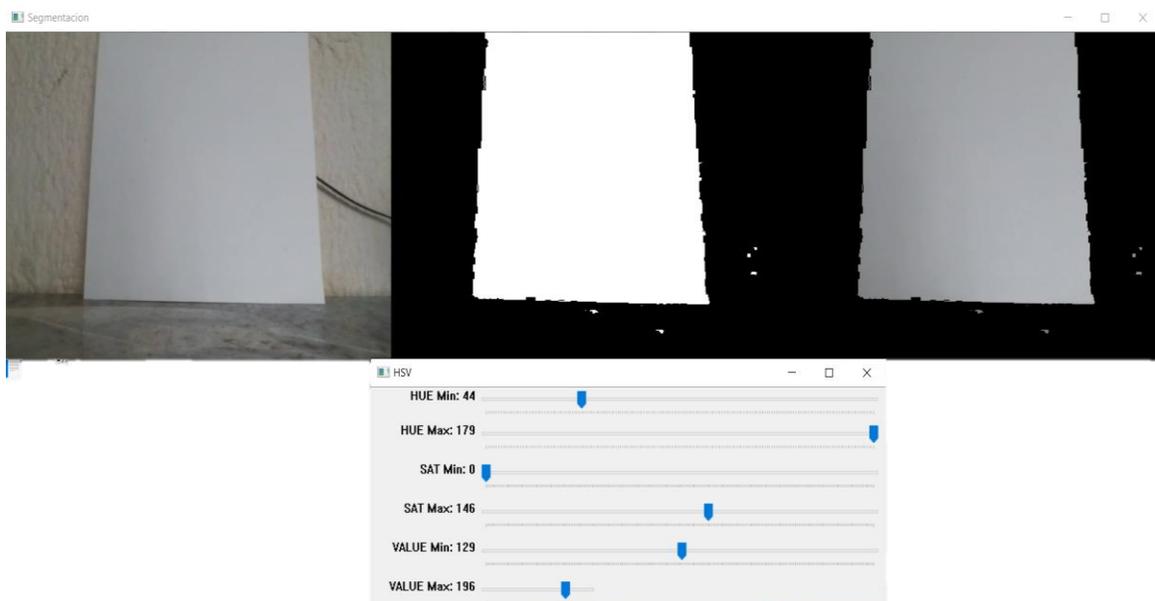
Las principales técnicas de procesamiento digital de imágenes son utilizadas para mejorarlas y resaltar sus características principales. Para esto se requieren de diversas operaciones donde se llevan a cabo la modificación del histograma de la imagen, el manejo y procesamiento del valor de sus píxeles, así como métodos de filtración de la frecuencia para poder tener una buena imagen y tener una manipulación y manejo total de la misma. (Malacara Hernández, D., 1997).

### 2.1.1. Visión por computadora.

“La Visión por Computador, que ha emergido como una disciplina propia basada principalmente en las matemáticas y ciencias de la computación, consiste en hacer que un computador vea” (Mery, D., 2004).

El término visión por computadora está siendo muy usado, ya que engloba una gran disciplina de estudio dentro de la robótica e informática, donde los avances permiten un desarrollo del diseño de soluciones de carácter automático, inteligente y dinámico, que permite a industrias tecnológicas de vanguardia alcanzar objetivos de alto nivel, es por esto que también se considera una disciplina desarrolladora de sistemas los cuales son capaces de interpretar el contenido de una escena natural y procesarlas para su uso y estudio, así como se muestra en la Figura 2.

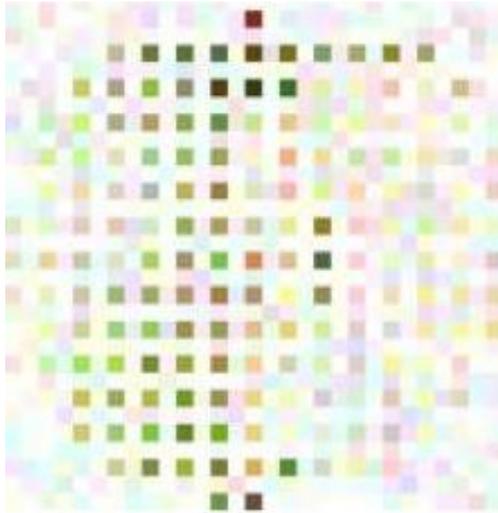
Es importante mencionar que la visión por computadora hace referencia a un conjunto de tecnologías o herramientas que permiten a los equipos tecnológicos captar imágenes reales, procesarlas y generar información, es una propiedad tecnológica que permite a los equipos de cómputo “ver”, poder detectar y analizar lo que existe en el mundo real (Mery, D., 2004).



**Figura 2.** Procesamiento de imágenes y segmentación donde se pretende separar el objeto de análisis del resto de la imagen.

### 2.1.2. *Píxel.*

El píxel es considerado el elemento básico de una imagen (picture element) y es por lo que surge de la mezcla de estas dos palabras, además de que es por esto que es conocida como la unidad homogénea más pequeña en color que compone la misma imagen. En la Figura 3 se muestra una imagen donde se pueden ver los píxeles (*Elizondo, J. E., & Maestre, L. P., 2005*).



**Figura 3.** Imagen representativa donde se pueden apreciar los píxeles (*Oltra, G. N., & Mellado, J. R. A., 2008*).

### 2.1.3. *Imagen.*

Se considera una imagen como un arreglo bidimensional de píxeles con una diferente intensidad lumínica cada uno, es decir, una escala de grises (*Elizondo, J. E., & Maestre, L. P., 2005*) y en términos más generales se puede decir que “es un soporte visual que materializa un fragmento del entorno óptico”.



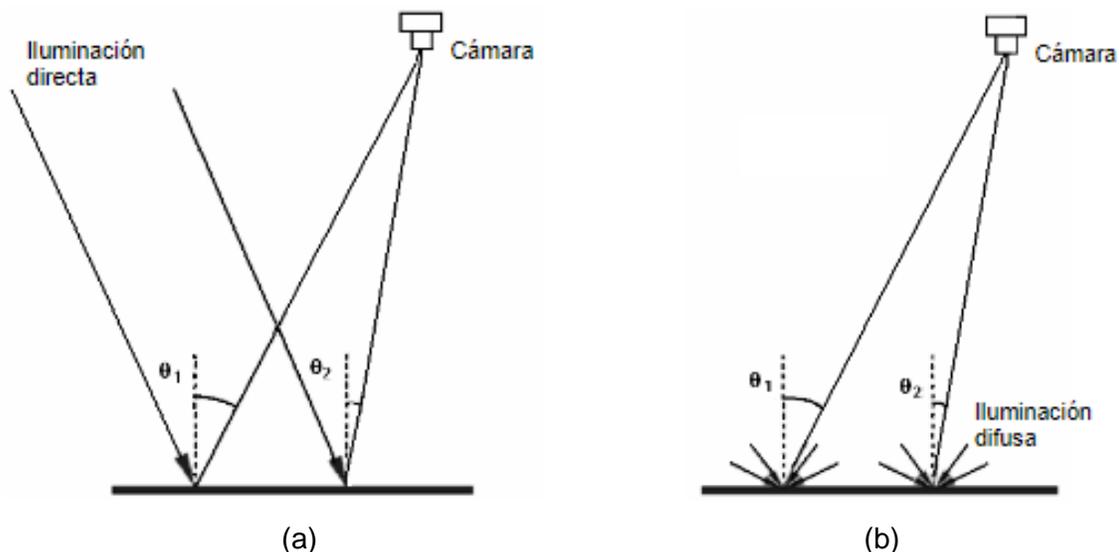
**Figura 4.** Imagen tomada con ayuda del dron que se usa para este trabajo.

### 2.1.4. Iluminación.

Cuando en un entorno o lugar la iluminación no es correctamente controlada, no suele ser aceptable la imagen, es decir, se obtiene una imagen de bajo contraste, ya sea con reflexiones especulares, sombras o destellos que no favorecen la imagen. Por lo que cuando se tiene una buena iluminación, existe una buena luz a la escena de forma a la imagen, manteniendo la información necesaria para su manejo y manipulación de esta (Enrique Alegre, Gonzalo Pajares y Arturo de la Escalera., 2016).

Es importante mencionar la relevancia que tiene la iluminación en una imagen, ya que nos ayuda a distinguir, apreciar e identificar de manera correcta la imagen en cada uno de los detalles que sean relevantes para cualquier tipo de uso que se le dé a la imagen, dependiendo de su iluminación. También es importante mencionar que un problema que se presentan en una imagen es la diferencia que existe entre la iluminación directa y la iluminación difusa y el contraste que existe cuando una predomina más que la otra.

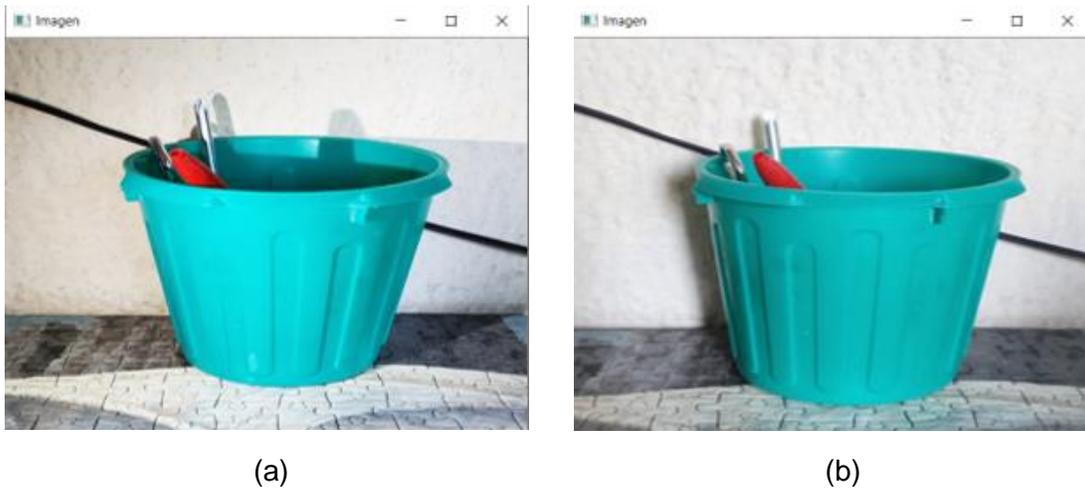
Hay que tener en cuenta que cuando la iluminación en su mayoría es iluminación directa, la manera en que se refleja la luz sobre los objetos no es uniforme y puede variar con el ángulo que es formado entre el punto de vista y la forma de la superficie, lo cual ocasiona que la luz no sea uniforme en toda la imagen Figura 5(a). Por otro lado, cuando predomina la iluminación difusa, las superficies son capaces de reflejar la luz de la misma forma en todas las direcciones, por lo tanto, la iluminación observada en la imagen es uniforme, Figura 5(b). (Ospina, J. C. M., Ortiz, F. A. P., & Bedoya, J. W. B., 2007).



**Figura 5.** (a) Representación de la escena con luz directa. (b) Representación de la escena con luz difusa (Ospina, J. C. M., Ortiz, F. A. P., & Bedoya, J. W. B., 2007).

Las imágenes donde existe una fuerte componente de luz directa, es decir, no presentan una iluminación uniforme, tal y como se puede ver en la Figura 6 (a)

donde los detalles de la imagen no se ven tan claros y con sombras, además de brillo excesivo centro de la imagen, por lo que provoca la necesidad de corregir la fuente de iluminación que impacta sobre la imagen para que se presente una iluminación uniforme o difusa sobre toda la imagen, tal y como se puede ver en la Figura 6 (b). En la Figura 6 se puede apreciar la diferencia que existe en la misma imagen con iluminación directa contra iluminación difusa. (Ospina, J. C. M., et. al, 2007).

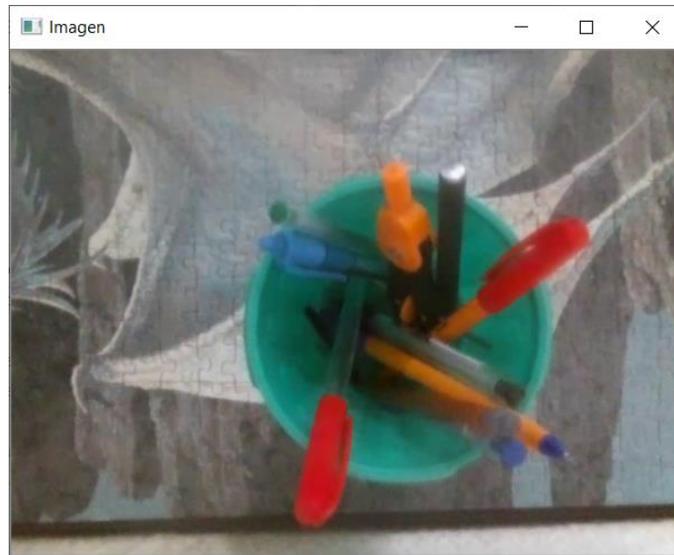


**Figura 6.** Diferencia de iluminación en la misma imagen. (a) Imagen con iluminación directa (b) Imagen con iluminación difusa.

### **2.1.5. Teledetección.**

La teledetección se refiere a la adquisición de información sobre un objeto o fenómeno sin la existencia de un contacto físico con el mismo. Es por eso por lo que se relaciona con la observación directa y captura de imágenes. La percepción remota (Remote Sensing en inglés) es una disciplina científica que conlleva una extensa variedad de conocimientos y tecnologías que se implementan en la observación, análisis e interpretación de fenómenos terrestres y planetarios. Actualmente, el uso de esta palabra se refiere a la utilización de tecnologías aéreas para la captura, análisis y toma de áreas específicas (Di Leo, Néstor C., 2015).

El rango de aplicación de esta disciplina es muy amplio, además de que comprende campos como lo son la cartografía, la agricultura, la geología, entre otras. También es común que después de la adquisición de este tipo de imágenes se requiera hacer correcciones o mejoras. Esto por diversos factores como la distancia, la iluminación, condiciones atmosféricas, entre otras que afectan directamente a los píxeles de la imagen o en otras palabras a la calidad de esta. En la Figura 7 se muestra una imagen tomada por el dron que hace referencia a la teledirección (Di Leo, Néstor C., 2015).



**Figura 7.** Imagen capturada con el dron que se usa para este trabajo la cual hace referencia a la teledetección.

### **2.1.6. Color.**

“El color es un efecto óptico en el que intervienen procesos fisiológicos, físicos y químicos” (Castañeda, W., 2005, Pag 9). Es por esto que la amplia gama de colores que se conocen es gracias a la combinación de los tres colores básicos o bien conocidos como colores primos, rojo, azul y verde en inglés RGB.

Cualquier aspecto visual o que podamos observar y sea cuantificable en un objeto, se debe al color y a la luminosidad. Esto debido a que los límites que definen la forma de los objetos o figuras son derivados de la capacidad que tiene del ojo para distinguir entre las zonas de luminosidad y entre cada color previsible, tal y como se puede ver la diferencia de colores en la Figura 8 (Castañeda, W., 2005).



**Figura 8.** Imagen representativa referente al color (Alegria, E. et al. 2016).

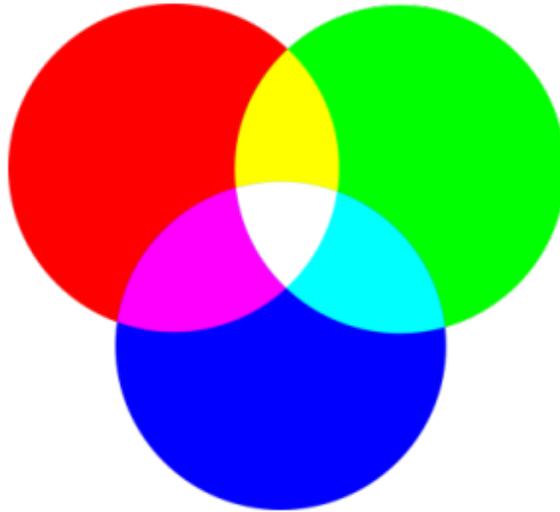
Pese a que el color es considerado una de las características más importante que nos ayudan a definir a los objetos anteriormente, no se le daba importancia a la visión por computadora, esto gracias a que las computadoras no podían diferenciar de manera correcta los colores y la memoria necesaria para su procesamiento era insuficiente (*Alegre, E. et al. 2016*).

Para poder comprender mejor que es el color, es necesario definir algunos conceptos indispensables para el mismo, tal y como lo son (*Alegre, E. et al. 2016*):

- Brillo: Luz emitida por la imagen y la claridad con la que se puede distinguir, indicando que área es más o menos luminosa, también se puede definir como la sensación, la cual indica si un área está más o menos iluminada.
- Colorida: Grado de intensidad y disposición de los distintos colores.
- Tono: Indica si un área se asemeja o tiene similitud al rojo, amarillo, verde o azul, es decir la escala de grises que la imagen puede tener, en una escala general de tonos en blanco y negro va del negro más intenso al blanco.
- Saturación: es la relación existente entre lo que se conoce como colorido y el brillo que se percibe en una imagen, es un atributo que diferencia un color intenso de uno pálido.
- Luminosidad: Brillo que se aprecia de una zona respecto a otra en una imagen, indica el aspecto luminoso del color, es decir, la percepción no lineal que se tiene de la cantidad de luz recibida.
- Croma: Muestra la coloración que existe en un área contra al brillo de un blanco de referencia, es decir, que indica la pureza e intensidad de un color.
- Intensidad: Es la representación de la iluminación percibida, así como de la sensación de que algún objeto refleje más o menos luz.

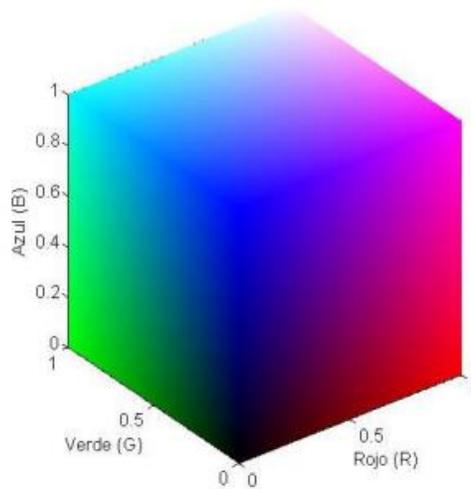
### **2.1.7. Espacio de color RGB**

Este concepto se basa en la mezcla de tres colores primarios o señales de luminancia cromática distinta, es decir, el color, rojo, verde y azul, en inglés “Red, Green, Blue” por esto sus siglas, para poder obtener un color es necesario obtener la cantidad de rojo, verde y azul que se necesita mezclar. En la figura 9 se aprecia una representación gráfica de la mezcla de colores mediante una suma aritmética de los canales R, G, B (*Elizondo, J. E., & Maestre, L. P., 2005*).



**Figura 9.** Modelo de color RGB (Marcial, Jorge A., 2011).

En la Figura 10 se puede apreciar un espacio RGB que es representado por un espacio tridimensional, donde surge la necesidad de normalizar el espacio de color RGB a causa de la necesidad que existe a los cambios uniformes de los niveles de luz. También aquí se puede mencionar que cuando una cámara obtiene una imagen a color, a cada píxel se le asigna el valor de tres componentes, los cuales son cada uno de los colores primarios; por lo tanto, la ganancia máxima para cada componente es correspondiente a la longitud de onda de tres colores primarios. (Elizondo, J. E., & Maestre, L. P., 2005).



**Figura 10.** Espacio de colores RGB representado en forma de cubo (Alegria, E. et al. 2016).

Es importante mencionar que la segmentación de imágenes debe de ser desarrollada en un espacio en tercera dimensión, tal y como se ve anteriormente en la Figura 9, por lo tanto, al tener un rango de valores positivos y negativos existen las variaciones de los niveles de luz, es decir, del brillo. Esto suele ser un problema

que se presenta en el espacio RGB y se soluciona normalizando el espacio de color RGB (Alegria, E. et al. 2016).

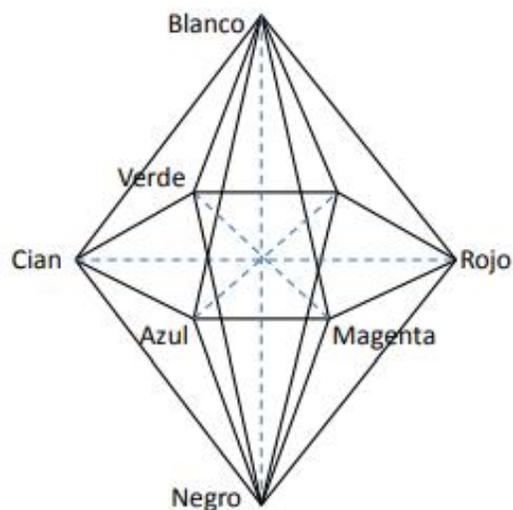
Con esto podemos decir que mediante la modulación y manejos de las señales roja y azul nos permite apreciar dos nuevos parámetros; la fase de la subportadora, la cual transmite el tono, y la amplitud que es la que transmite la saturación. Desde el punto de vista de la colorimetría, estos parámetros definen el aspecto cualitativo del color, lo cual nos dice que el aspecto cuantitativo está vinculado a la luminancia (Alegria, E. et al. 2016).

### 2.1.8. Espacio de color HSI.

El espacio de color HSI se basa en colores que percibimos los humanos, y como se diferencian unos de otros por el tono, la intensidad y la saturación.

Es por lo que se distingue el color en términos de tono (Hue), saturación (Saturation) y brillo (Intensity); donde estos factores son importantes y facilitan la segmentación de la imagen en relación con el tono del color, con mayor invariancia a la iluminación, lo cual nos permite tener una mejor separación de colores.

Si ponemos un ejemplo donde se tienen dos fuentes de luz que tienen el mismo espectro entre la comparación del espacio HSI y el espacio RGB de colores se puede decir que a través del espacio RGB, la que presenta una mayor intensidad es la que aparece con un color más brillante y, por otro lado, en el espacio HSI se pueden segmentar colores basados en el tono, una luz verde, y otra roja se diferencian los diferentes tonos que representan la longitud de onda del color específico, todo esto sin la necesidad de recurrir a la intensidad lumínica como lo es el caso del espacio de color RGB (Alegria, E. et al. 2016).



**Figura 11.** Espacio de color HSI el cual se representa en forma de bipyramide hexagonal (Alegria, E. et al. 2016).

La Figura 11 representa un espacio de color HSI que tiene forma de bipyramide hexagonal. En este caso, el vértice inferior indica el color negro y el vértice superior

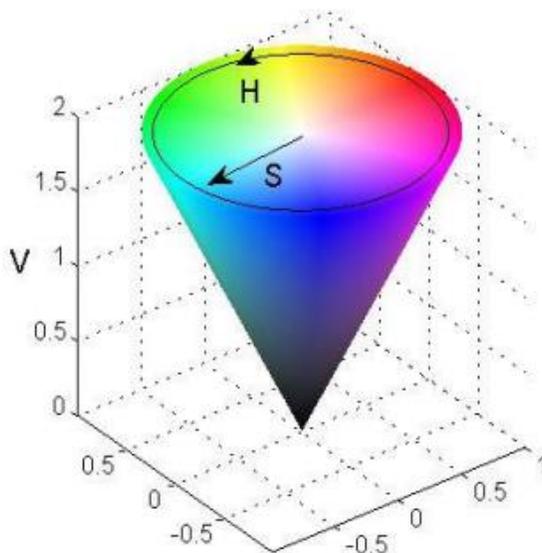
indica el color blanco. También es importante mencionar que el tono indica el ángulo que se forma entre el eje de referencia (al color rojo) y el punto que ocupa el color obtenido en el espacio RGB (Enrique Alegre, Gonzalo Pajares y Arturo de la Escalera., 2016).

Existen variaciones de este espacio de color, las cuales se mencionan a continuación:

- HSL (Hue, Saturation and Lightness)
- HSV (Hue, Saturation and Value)
- HCl (Hue, chroma / colourfulness and Intensity)
- HVC (Hue, Value and Chroma)
- HSD (Hue, Saturation and Darkness)

### 2.1.9. HSV.

El modelo HSV da como resultado un modelo de color en términos de sus componentes, este modelo es considerado más realista que el modelo RGB de cómo es que las personas describen o representan las sensaciones del color. Este modelo obtiene el nombre HSV, por sus siglas en inglés (Hue, Saturation, Value), que significa Tonalidad, Saturación y Valor. (Jorge Alberto Marcial., 2011). Este es una transformación no lineal del modelo de color RGB, donde los colores terminan siendo una mezcla de los tres valores: tonalidad, saturación y su valor, estos valores se puede representar, así como se ve en la Figura 12.

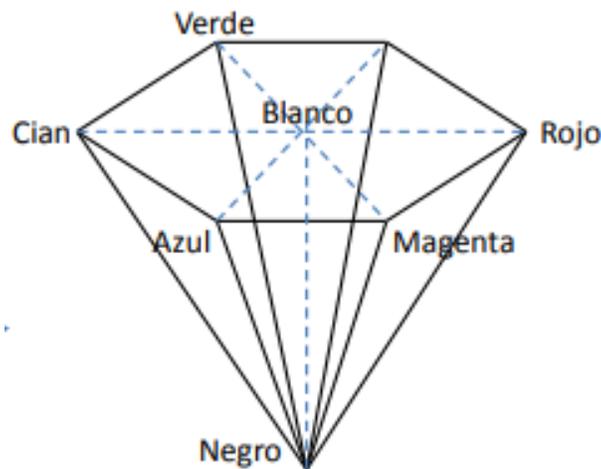


**Figura 12.** Modelo de color HSV (Alegria, E. et al. 2016).

Para poder elegir un color en este modelo es necesario seleccionar el tono en todo lo que compone la región circular con valores entre  $0^{\circ}$  y  $360^{\circ}$ , donde cada valor corresponde a un color y posteriormente seleccionar la saturación del color que se encuentra en el eje horizontal del cono (S), finalizando con el color en el eje vertical. En la Figura 13 también se puede ver el modelo HSV(V) (, E. et al. 2016).

Es decir que cada magnitud tiene los siguientes valores (Marcial, Jorge A., 2011):

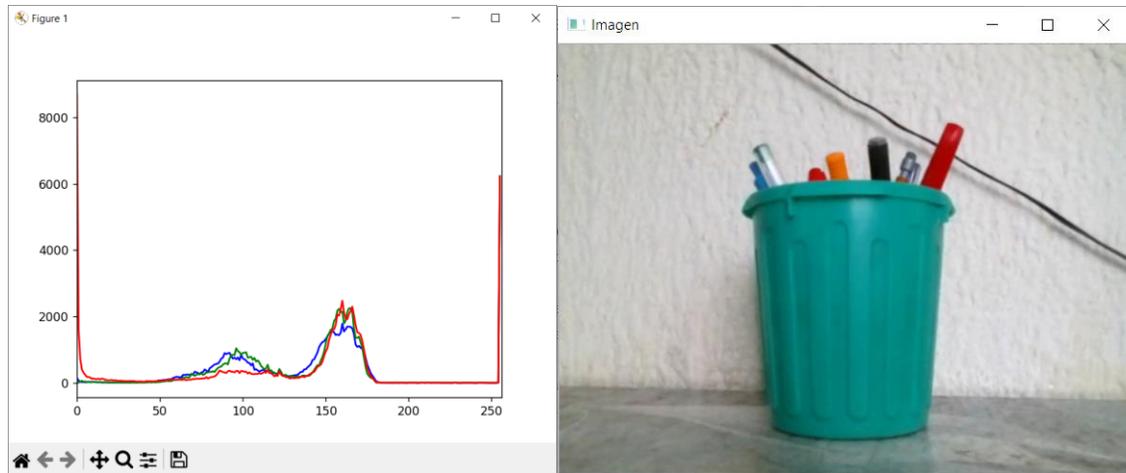
- Tonalidad (Hue): Este es el tipo de color (verde, azul, amarillo) que se puede representar con el valor de un ángulo cuyos valores están dentro del rango de  $0^\circ$  a  $360^\circ$  o bien en un valor normalizado de 0% a 100%.
- Saturación (Saturation): Este se representa es como la distancia del eje del brillo entre negro y blanco, donde se maneja un rango estándar de 0% a 100%.
- Valor (Value): Finamente, el valor se representa con la altura en el eje negro al blanco, donde de igual manera los valores del rango están estandarizados entre 0% y 100%, el cual nos dice que o es negro, todo depende de la saturación, y 100, blancos o un color más o menos saturado.



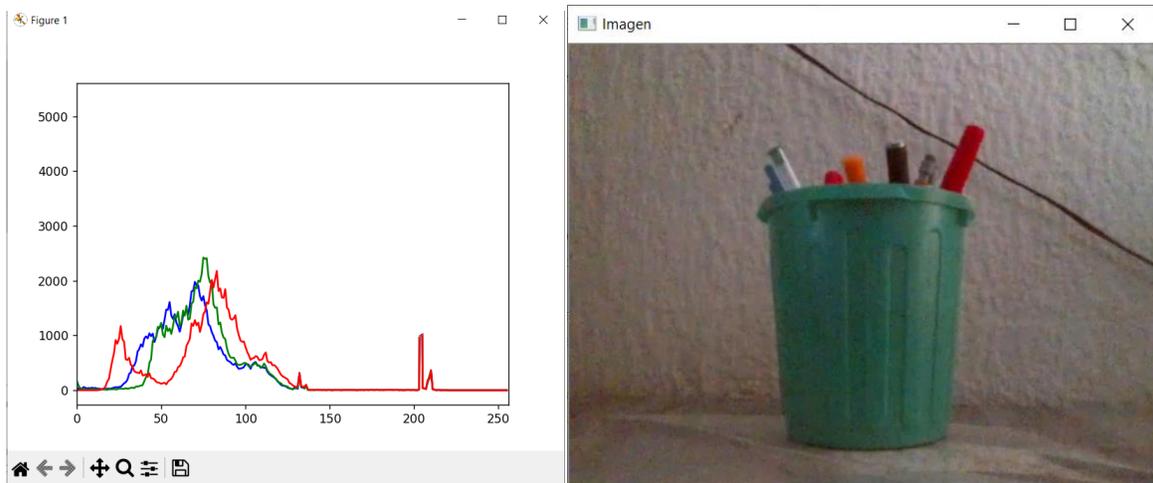
**Figura 13.** Espacio de color HSV que se representa en forma de cono invertido. (Alegria, E. et al. 2016).

#### **2.1.10. Histograma.**

Un histograma es capaz de entregar una representación del cómo se distribuye el brillo de una imagen, así como también una información especificar sobre las características del brillo de la imagen que se analiza y poder trabajar con los resultados. Es importante mencionar que cuando se captura una imagen con cualquier tipo de cámara digital es importante la cantidad de luz y el ángulo en la que se toma la imagen, debido a que esto involucra directamente a las características de su luminosidad y contraste de la imagen que se toma. Idealmente, un histograma ocupa prácticamente todo el rango de tonos. En las Figuras 14 y 15 se puede ver la un ejemplos de histogramas con la luz se da sobre la imagen de manera uniforme y donde no, respectivamente (Atienza Vanacloig, V. L., 2011).



**Figura 14.** Imagen donde la luz se reparte uniformemente y se ve un histograma amplio.



**Figura 15.** Imagen donde la luz no está bien repartida, es decir falta de exposición.

### 2.1.11. *Umbralizacion (Thresholding).*

La umbralización de una imagen es considerada una técnica de separación o segmentación por la cual se busca un umbral adecuado, el cual permita diferenciar dentro de una imagen objetos en específico. Este proceso es capaz de separar los píxeles que contiene una imagen en una escala de grises, así como en dos categorías (blanco o negro), o también conocido como puntos de objeto y puntos de fondo, todo esto a partir de un umbral de intensidad. El umbral también se puede definir como el punto en donde el histograma de una imagen se divide en dos picos. (Cattaneo, C. A., et al., 2011). En la Figura 16 se puede ver la umbralización de un objeto.



**Figura 16.** Umbralización de una hoja de papel.

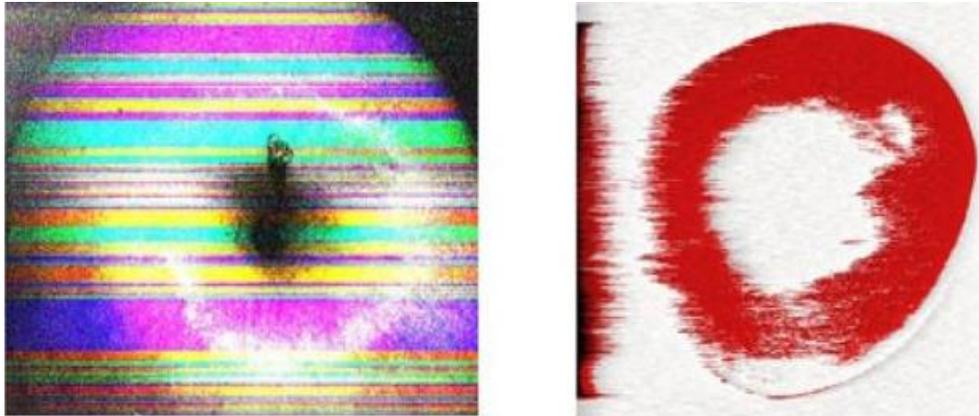
## **2.2. Definición de una imagen digital.**

Según el trabajo de Aguirre Dobernack, N., “Una imagen digital está compuesta de un número finito de elementos y cada uno tiene una localidad y un valor particular”. Se les dice píxeles a estos puntos finitos, el término más utilizado para expresar la unidad de medida de una imagen digital. (Aguirre Dobernack, N., 2013).

La resolución espacial es uno de los parámetros que más importan en una imagen, es el número de píxeles que contiene una imagen, la cual es utilizada para la clasificación de la mayoría de los dispositivos que tiene relación con las imágenes digitales, tanto como de pantallas de computadoras, escáneres, cámaras digitales, televisores, impresoras, entre otras. La cantidad de píxeles que conforman una imagen de mapa de bits es considerada como la resolución total, esto significa que la calidad que tiene una imagen depende estrictamente de su resolución. La resolución de una imagen es expresada por dos valores numéricos, el primero es la cantidad de píxeles que contiene una imagen de lo ancho y el segundo valor es la cantidad de píxeles que contiene una imagen en lo alto. (Aguirre Dobernack, N., 2013).

Es lógico y muy coherente pensar que una característica principal de una imagen digital podría ser toda la que se ha convertido en información alfanumérica y que se construye mediante algún algoritmo matemático, es decir, binaria, la cual es representada visualmente por píxeles (ya sea de un determinado tamaño y resolución), es importante mencionar que es indiferente si se habla de una imagen, sonido, números, ecuaciones matemáticas, palabras, etc. Todo acaba dentro de una computadora en código binario programado. También se considera a una imagen digital una imagen que posee naturalmente una estructura formal, está sometida a las condiciones que son impuestas por el lenguaje binario que los programadores o en el ámbito informático se utiliza para procesar la información,

sin importar cuál sea su origen o tipo como por ejemplo, algebraica, matemática, textual, visual, sonora, etc. A continuación en la Figura 17 se agrega un ejemplo de dos imágenes que fueron generadas por programación. (Oltra, G. N., & Mellado, J. R. A., 2008).



**Figura 17.** Ejemplo de imágenes digitales generadas por programación. (Oltra, G. N., & Mellado, J. R. A., 2008).

### **2.2.1. Imagen digitalizada.**

Cuando se habla de una imagen digital se refiere a la condición que es posible alcanzar mediante el poder abstracto de la mente, es decir, que es dependiente de la naturaleza, de la realidad a la que se está enfrentando esta imagen, para poder ser digital.

Una imagen digital se puede considerar así por ser creada desde la computadora, ya sea porque la adquisición de la imagen se haga trasvés de cámaras que adquieren la imagen, por otro lado, la imagen digitalizada tiene como punto principal un soporte gráfico, una imagen analógica que después mediante un escáner se digitaliza y después de ser incorporada a la computadora poder ser sometida a cualquier tratamiento de dicha imagen, es decir, poder manipular sus propiedades.

Existe una diferencia entre una imagen digitalizada y una imagen digital, aunque la diferencia es muy pequeña, se puede decir que una imagen digital es de origen noble dentro del mundo digital, mientras que la imagen digitalizada es una especie de “hija” dentro del mismo, por lo que la imagen digital sería más electrónica que la digitalizada. Se puede apreciar una imagen digitalizada en la Figura 18.

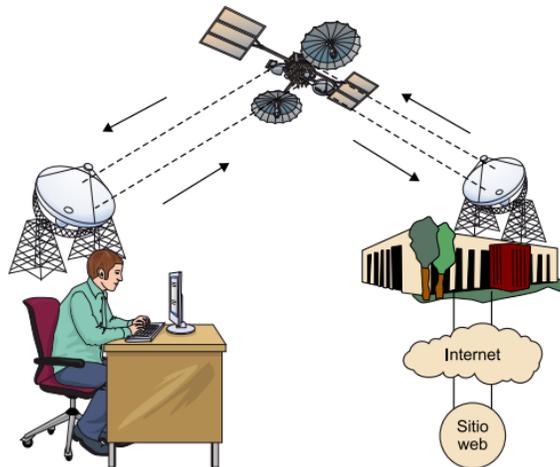


**Figura 18.** *Imagen digitalizada.* (Oltra, G. N., & Mellado, J. R. A., 2008).

### **2.3. Comunicación inalámbrica.**

Una comunicación inalámbrica es considerada una comunicación sin la necesidad de cables, es decir, es una comunicación donde no se requiere utilizar un medio de propagación físico. Por el contrario, en esta comunicación se utiliza la modulación de ondas electromagnéticas, las cuales son propagadas por el espacio sin la necesidad de algún medio físico que requiera comunicar cada uno de los extremos de la transmisión. Es por lo que los dispositivos físicos se presentan con los emisores y receptores de la señal. Por ejemplo: antenas, computadoras, teléfonos móviles, entre otras. Se sabe que el ser humano siempre ha ido buscando establecer comunicación entre dos puntos específicos y para ello se necesita establecer algún tipo de conexión para que esto se pueda llevar a cabo, es aquí donde entra la comunicación inalámbrica (Morando, L., 2004).

En términos generales, se puede definir por comunicaciones inalámbricas a las comunicaciones que existen entre dispositivos, ya sean o no móviles, o intercambio de información entre personas donde se utiliza el espectro electromagnético. Es importante mencionar que las comunicaciones inalámbricas abarcan desde la comunicación Bluetooth que existe entre un teléfono y una computadora, hasta una comunicación entre dos teléfonos móviles, estrictamente hablando la comunicación inalámbrica puede incluir la comunicación verbal entre dos personas que se encuentran en el mismo lugar, ya que se utiliza el aire como un medio para el intercambio de información, tal y como se ve en la Figura 19 (Blázquez, J. P., 2015).



**Figura 19.** Imagen representativa de la comunicación inalámbrica (Blázquez, J. P., 2015).

## 2.4. Redes inalámbricas.

De acuerdo con lo que se dice (Morando, L., 2004), “Una red inalámbrica es como cualquier otra red de computadores, su función es la de conectar ordenadores a la red, pero en este caso sin necesidad de cables. Estas se pueden clasificar según su cobertura y según su rango de frecuencia.”. (Morando, L., 2004).

Las redes se clasifican de acuerdo a su cobertura y frecuencia, como se muestra a continuación.

### 2.4.1. Según su cobertura.

- Wireless Personal Área Network (WPAN).

La red WPAN se considera como una red de cobertura personal, donde las tecnologías están basadas en HomeRF, es decir, para conectar computadoras y teléfonos móviles al interior del hogar usando un aparato central que puede estar basado en Bluetooth, ZigBee; las cuales son comunicaciones consideradas como seguras con una tasa baja de transmisión de datos, pero con una mayor vida útil de sus baterías; RFID el cual es un sistema remoto de almacenamiento y recuperación de datos (Morando, L., 2004).

- Wireless Local Área Network (WLAN).

Dentro de las redes que son de área local se encuentran tecnologías inalámbricas basadas en HiperLAN o Wi-Fi. (Morando, L., 2004).

Esta red nos permite establecer conexiones inalámbricas exclusivamente en el área que cubre la red, es decir, hasta donde la señal alcance a llegar. Las WLAN se pueden manipular de dos diferentes maneras. Las WLAN de infraestructura, son las estaciones inalámbricas, es decir, son dispositivos que cuentan con radio, módem externo o tarjetas de red, los cuales son conectados a puntos de acceso inalámbrico que trabajan como enlaces entre las estaciones y la red primaria existente. (Coñapes, S. y Cerro, Y., 2015).

- Wireless Metropolitan Área Network (WMAN).

Esta red es utilizada para áreas metropolitanas las cuales se basan en tecnologías WiMAX. Este protocolo es parecido al Wi-Fi; sin embargo, tiene una cobertura mayor y su ancho de banda es mejor. También es posible usar otro tipo de redes como el LMDS por sus siglas en inglés (Local Multipoint Distribution Service). (Morando, L., 2004).

Las redes WMAN son capaces de hacer que los usuarios establezcan conexiones inalámbricas entre distintas ubicaciones que se encuentran de un área metropolitana, pero sin el precio tan elevado que representa la instalación de cableado especial, como es cobre o fibra óptica. La red WMAN basa su funcionamiento en ondas de radio o en el uso de luz infrarroja para realizar la transmisión de datos, además de que cuentan con un rango de transmisión y alcance mucho mayor que el de las WLAN, en el rango de decenas de kilómetros, por lo que alcanza para cubrir a una población completa. (Coñapes, S. y Cerro, Y., 2015).

- Wireless Wide Área Network (WWAN).

La UMTS por sus siglas en inglés (Universal Mobile Telecommunications System) es ejemplo de las redes WWAN, las cuales se utilizan en los teléfonos móviles de tercera generación (3G) y es sucesor de la tecnología GSM desarrollada para dispositivos móviles de segunda generación (2G). También es posible que se conecten a diferentes localidades, ya sea usando conexiones con satélites o usando antenas de radio de microondas, lo que la hace de fácil instalación, bajo costo económico y que mucho más flexible. (Morando, L., 2004).

En la Figura 20 se puede ver de manera más resumida el acomodo que se le da a los tipos de redes según su cobertura.

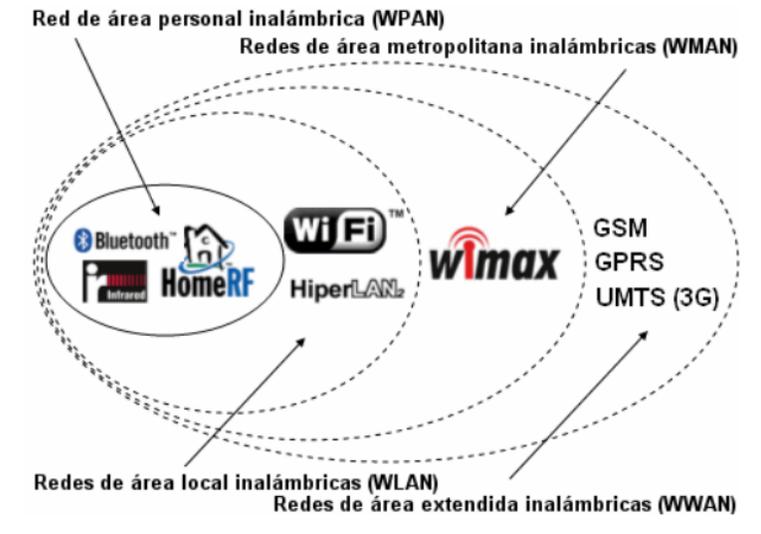


Figura 20. Tipos de redes inalámbricas (Coñapes, S. y Cerro, Y., 2015).

#### 2.4.2. Según su rango de frecuencia.

- Ondas de radio.

Tienen la ventaja de que la transmisión es insensible a las atenuaciones provocadas por la lluvia. Esto debido a que opera en frecuencias no tan altas. Dentro de este rango se encuentran las bandas desde la ELF (Extremely Low Frequency) hasta la banda UHF (Ultra High Frequency), es decir, que abarca el espectro de radiofrecuencia de 30 Hz – 3 GHz (Morando, L., 2004).

- Microondas terrestres.

Para poder usar este tipo de red se requiere usar antenas parabólicas, las cuales tienen un diámetro de tres metros aproximadamente, y tiene una cobertura de varios kilómetros, sin embargo, el emisor y el receptor deben estar en la misma posición y alineados a la perfectamente. Por lo anterior, este tipo de tecnologías se usan en enlaces punto a punto con distancias cortas. Una desventaja es que la atenuación que se produce por la lluvia es muy importante, porque se opera a una frecuencia mucho más grandes que la de la lluvia. Las microondas comprenden las frecuencias aproximadas de 1 hasta 300 GHz (Morando, L., 2004).

- Microondas por satélite.

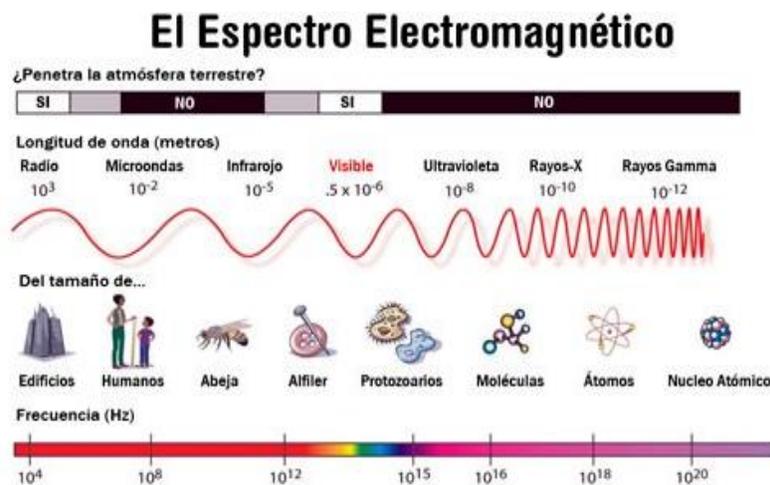
Para las microondas por satélite es necesario hacer enlaces entre dos o más estaciones que se encuentren en la tierra, también conocidas como estaciones base o terrestres. Funciona debido a que el satélite recibe la señal, la cual se denomina señal ascendente en una banda de frecuencia, la cual es amplificada y se retransmite en otra banda, es decir, es una señal descendente, donde cada satélite opera en una banda concreta. Es muy común que los límites de las frecuencias de comunicaciones por microondas, tanto terrestres como por

ejemplo de satélite, con los infrarrojos y las ondas de radio de alta frecuencia, se mezclen bastante, por lo que ocasiona interferencia con las comunicaciones en frecuencias específicas (Morando, L., 2004).

- Infrarrojos.

Existe un enlazamiento entre transmisor y receptor, el cual controla la luz infrarroja no coherente, por lo que es necesario que estén alineados directamente. Por lo que no le es posible atravesar las paredes a esta red. Los infrarrojos van desde 300 GHz hasta 384 THz (Morando, L., 2004).

En la Figura 21 se representa de manera grafica el especto electromagnético.



**Figura 21.** Representación visual del efecto electromagnético. (Clark, Bryan y Walton, Timothy; 2020).

### 2.4.3. Tipos de conexiones inalámbricas principales.

- Tecnología Inalámbrica Bluetooth.

Las primeras tecnologías de Bluetooth iniciaron a principios de 1998. Sin embargo, no fue público hasta mayo del mismo año. Al transcurrir 2 meses desde su publicación, la primera especificación Bluetooth 1.0 fue liberada. El Bluetooth está dirigido principalmente a dispositivos que pueden ser computadoras, teléfonos móviles, tabletas electrónicas, audífonos, entre muchas otras cosas más. Esta tecnología es capaz de lograr que las personas se puedan conectar con cualquier tipo de dispositivo que cumpla con las condiciones y especificaciones inalámbricas Bluetooth. también es importante mencionar que cuenta con un estándar libre, lo que nos permite diseñar y crear nuevos productos que se beneficien de este. La conectividad a través del Bluetooth cuenta con dos capas, la capa de enlace y la capa de aplicación. Cada dispositivo que cuenta con Bluetooth está equipado con un transmisor y recibe a una frecuencia de 2.4 GHz (Morando, L., 2004).

- **Redes Inalámbricas Wifi**

De acuerdo con (*Morando, L., 2004*) el Wi-Fi “es un sistema que nos permite crear redes inalámbricas, también conocidas como Wireless”. Las redes WiFi interactúan y construyen una tecnología que cada vez más se incorpora a diversos equipos informáticos como lo son computadoras, consola de videojuegos, teléfonos, impresoras que utilizan este tipo de tecnología para hacer una conexión fácil.

En el trabajo de (*Coñapes, S. y Cerro, Y., 2015*). Se mencionan que “Los diferentes tipos de redes inalámbricas se rigen bajo normas y estándares, las cuales fueron establecidas por el instituto de ingenieros eléctricos y electrónicos, más conocido como la IEEE. La norma más utilizada es la IEEE 802.”.

- Red inalámbrica de área personal: esta red se rige bajo la norma IEEE 802.15

- Red de área local: esta red utiliza la IEEE 802.11

- Red de área metropolitana: utiliza la IEEE 802.16

### **2.5. Python.**

Python surgió a principios de los años 90, cuando Guido Van Rossum, un trabajador del CWI (Centrum Wiskunde & Informatica), el cual es un centro de investigación holandés, se le ocurrió desarrollar un nuevo lenguaje el cual se basó en un proyecto previo, el lenguaje de programación “ABC”, el cual, él mismo había desarrollado junto a sus compañeros (Pérez González, et al., 2019).

De acuerdo con áreas (Pérez González, et al., 2019) “Python es un lenguaje de programación de alto nivel que se utiliza en el desarrollo de todo tipo de aplicaciones, a diferencia de otros lenguajes como Java o .NET, es un lenguaje interpretado”. Esto quiere decir que no es necesario compilar el programa para ejecutar las instrucciones que se realizan o se desarrollan en Python, por lo que se ejecutan directamente por la computadora usando un programa que se denomina denominado interprete o IDE (Integrated Development Environment), y no se requiere pasar el programa a lenguaje máquina para su ejecución.

Este lenguaje es un lenguaje multiplataforma de código abierto, es decir, que es gratuito y permite desarrollar software sin límites. Además de que es sencillo de leer y escribir por su semejanza con el lenguaje de las personas. Debido a la sencillez y sus amplias posibilidades para trabajar con el Python ha tomado importancia, ya que es más fácil aplicar software de inteligencia artificial, data science, big data, machine learning, entre muchas otras áreas (Pérez González, et al., 2019).

Es muy importante mencionar que Python ha ganado mucho terreno en el desarrollo de aplicaciones, debido a una serie de ventajas, tanto para su uso profesional como para aprender a programar, por ejemplo (Marzal Varó, A., & Gracia Luengo, I., 2009):

- Los programas en Python suelen ser más cortos que su equivalente en un lenguaje C;
- Python es muy legible, es decir, la sintaxis de Python es muy elegante y presume la escritura que hace que la lectura resulte más fácil que utilizar otros lenguajes de programación.
- Python puede utilizarse como un lenguaje imperativo procedimental o como lenguaje orientado a objetos.

Una de las grandes ventajas de Python es la la facilidad del interpretar, es decir que el intérprete de Python básicamente tiene versiones para cualquier plataforma en uso, así como: sistemas PC bajo Linux, sistemas PC bajo Microsoft Windows, sistemas Macintosh de Apple, etc. (Marzal Varó, A., & Gracia Luengo, I., 2009).

### **2.5.1. OpenCV con python.**

OpenCV (Open-Source Computer Vision). Esto nos dice que OpenCV es una librería de manejo de imágenes, la cual está orientada principalmente a desarrollar aplicaciones de visión por computadora en tiempo real (Igual, R., & Medrano, C., 2008).

La librería de software open-source para aplicaciones de visión artificial y de machine learning, proporciona una infraestructura que se puede usar en aplicaciones de visión artificial. La librería está escrita en C++, y cuenta con interfaces para C++, C, Python, Java y MATLAB. Estas interfaces funcionan en Windows, Linux, Android y Mac OS (Martín, R., 2020).

OpenCV tiene a su disposición el desarrollo de algunos algoritmos para detectar objetos, o algunas características como pueden ser esquinas, las líneas, patrones de círculos, rectángulos o incluso rostros, entre otros. Esto es posible si no existe una definición cerrada de cómo es el objeto, donde es posible usar una detección de “puntos” los cuales definen la forma de un objeto en específico y la presencia, en una posición espacial de los mismos. Se conocen como puntos característicos o de interés, aquellos llamados sobresalient points, feature points o keypoint, y representan áreas pequeñas de una imagen, que se aproximan de manera muy certera de ser parte de algún objeto en una escena (Agustí Melchor, M., 2019).

Actualmente, la visión artificial o por computadora se utiliza para el análisis y procesamiento de imágenes mediante algoritmos de inteligencia artificial. La detección de rostros y objetos es uno de los usos más importantes de OpenCV en la visión por computadora. La librería OpenCV se maneja bajo una licencia BSD (Berkeley Software Distribution), lo cual permite que se utilice y modifique el código, además de que cuenta con una amplia comunidad de más de, 47000 personas y cuenta con más de 7 millones de descargas. Esta librería es ampliamente usada por empresas como Google, Yahoo, Microsoft, Intel, IBM, Sony, Honda, Toyota, entre otras. La librería cuenta con más de 2500 algoritmos, los cuales incluyen algoritmos de machine learning y de visión artificial (Martín, R., 2020).

Python permite al programador plasmar sus ideas en menos línea de código sin ver reducida la legibilidad, si se compara con lenguajes como C/C++. Python es más

lento que estos; sin embargo, una característica importante de Python es que puede ser compatible fácilmente con C/C++ y esto permite que se puedan desarrollar códigos densos en C/C++ y desarrolla un envoltorio de Python para que se puedan usar como módulos de Python, lo cual permite que el código sea tan rápido como uno de C/C++, y que es muy sencillo de codificar en Python. Es así como funciona OpenCV con Python, donde también la librería de Numpy nos ayuda a que trabajar OpenCV con Python sea un contenedor de Python alrededor de la implementación original de C++. El uso de Numpy hace que llegue a tener una sintaxis similar a MATLAB.

Es importante resaltar que todas las estructuras de arreglos de OpenCV se convierten desde y para arreglos Numpy, es por lo que cualquier operación que sea posible realizar en Numpy se puede combinar con OpenCV, lo que proporciona más herramientas el conjunto de Python y OpenCV, sin olvidar que otras librerías como SciPy, Matplotlib que admiten sin ningún problema Numpy se pueden usar.

Utilizar OpenCv con Python es una herramienta ideal para la creación efectiva y concreta de prototipos de problemas de visión por computadora que pueden ser admisibles para investigación, desarrollo de nuevas tecnologías o dar continuidad y actualización a proyectos existentes que se relacionen con problemas de visión por computadora.

## **2.6. Librerías utilizadas en Python.**

### **2.6.1. *The Open Computer Vision Library (Open CV)***

OpenCV es una librería que proporciona un amplio conocimiento para el desarrollo de aplicaciones que incluyen operaciones de visión por computadora. Así como proporciona estructuras de datos, así como análisis y procesamiento de imágenes, análisis estructural, entre otros. Esta estructura de información hace más sencillo el aprendizaje y la implementación de diferentes técnicas de visión por computadora, por lo que esto para docente, como para investigación en general, es de mucha ventaja. (Arévalo, V., González, J., & Ambrosio, G., 2004).

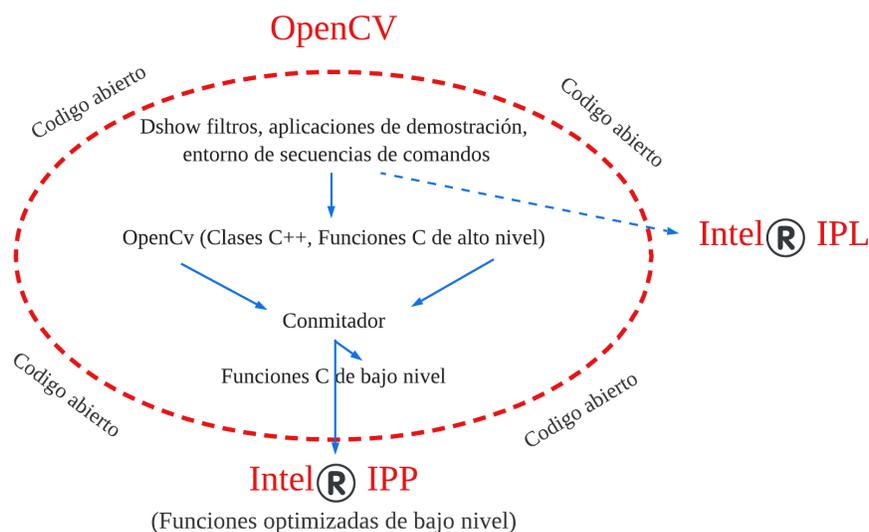
En junio del 2000, Intel® Corporation anuncia una colaboración con investigadores para el desarrollo de una nueva librería que incluye estructuras y funciones en lenguaje C. Esta librería proporcionaría una amplia gama para el trabajo de un nivel medio-alto que ayuda al personal docente y a los investigadores a desarrollar una nueva manera de interactuar con la computadora (Arévalo, V., González, J., & Ambrosio, G., 2004).

La librería OpenCV es una API que cuenta con aproximadamente 300 funciones desarrolladas en lenguaje C que tiene las siguientes características (Arévalo, V., González, J., & Ambrosio, G., 2004):

- Es de código abierto, es decir, que es de uso libre; tanto comercial como no comercial.

- No cuenta con extensas librerías numéricas, sin embargo, eso no quiere decir que no se pueda usar de algunas de estas librerías que están disponibles, para su uso en tiempo de ejecución.
- Es compatible con The Intel® Processing Library (IPL) y usa The Intel® Integrated Performance Primitives (IPP) para poder mejorar su rendimiento, si se encuentra disponible en el sistema.
- Es compatible y cuenta con interfaces para poder usarse con otros lenguajes y entornos de programación.

La librería OpenCV se enfoca principalmente a la visión por computadora en tiempo real. Sin embargo, eso no quiere decir que no destaque en otras áreas como lo son: estructura del movimiento (SFM5); segmentación y reconocimiento de objetos; interacción hombre-máquina (HCI4); reconocimiento de gestos; seguimiento del movimiento; y robots móviles. A continuación, en la Figura 22, se puede ver el acomodo de la estructura de la Librería OpenCV.



**Figura 22.** Estructura de la librería OpenCV.

OpenCV proporciona paquetes de alto nivel para la implementación y desarrollo de aplicaciones de visión por computadora y es muy práctico, ya que todos ellos se pueden agrupar en librerías de C/C++ dirigidas a usuarios expertos o de mayor conocimiento en el tema y en herramientas de scripting dirigidas, por otro lado, también es adecuado para usuarios de nivel medio (con el que se pueden practicar las distintas técnicas de procesamiento de imágenes y visión). (Arévalo, V., González, J., & Ambrosio, G., 2004).

### 2.6.2. Librería NumPy.

NumPy está basado en objetos de matrices numéricas y su objetivo es crear bases fundamentales de un entorno útil para la comparación científica, es una librería de Python especializada en el caculo numérico y análisis de datos, con un mayor enfoque a una gran cantidad de datos. También trabaja con una clase de objetos

llamados arreglos o arrays los que permiten representar un conjunto de datos de un mismo tipo de variables dimensionadas, y funciones muy eficientes para su manipulación (Oliphant, T. E., 2006).

Una de las ventajas más significativas de NumPy es que el procesamiento de los arreglos se realiza mucho más rápido que en las listas predefinidas de Python, por lo tanto, hace algo idóneo para el procesamiento de vectores y matrices de grandes dimensiones.

Un array o arreglo es una estructura de datos de un mismo tipo, la cual es organizada en forma de tablas o cuadrícula con diferentes dimensiones. Las dimensiones de un array también son llamadas ejes. (La librería Numpy, 2022).

## **2.7. Tecnología de drones.**

### **2.7.1. ¿Qué son los drones?**

UAS (Unmanned Aircraft Systems) en inglés, que significa “sistemas aéreos no tripulados”, es decir, que son aeronaves que tienen la capacidad de volar sin ningún tipo de tripulación a bordo y de manera autónoma para una función concreta. Un UAS se considera sinónimo parcial de dron, ya que cuando a un sistema aéreo no tripulado se le añade una cámara que sirva para monitorear o vigilar se convierte en uno.

Dependiendo del tipo de dron que se utilice son las ventajas que posee. Sin embargo, la más común es la alta resolución espacial, esto dependiendo de la calidad de la cámara, además al volar a bajas alturas permite tomar imágenes con gran detalle y en lugares que son difíciles de llegar sin esta tecnología. También ofrecen una elevada resolución temporal debido a que su facilidad de despliegue permite realizar vuelos con la frecuencia deseada.

Debido a la precisión de los sistemas de navegación que existen en la actualidad, es posible para los drones repetir vuelos de modo sistemático y de esta manera comparar las imágenes que se obtienen en diferentes tiempos, haciendo uso de herramientas de software específicas, esto es mucho más práctica y conveniente que un censo visual desde una aeronave tripulada.

El principal uso que se le puede adquirir a los drones por lo general es la adquisición de datos, es por esta razón que a la aeronave se le agregan un grupo de sensores como, cámaras fotográficas o de video (en espectro visible, térmico o multiespectral dependiendo del uso), sensores meteorológicos o cualquier otro dispositivo que se utilice para la toma de muestras o recaudación de información. El manejo o funcionamiento de los distintos sensores se pueden controlar desde una estación de control, donde además se reciben los datos que el dron va adquiriendo durante el vuelo (Mandujano, S. et al., 2017).

### 2.7.2. ¿Cómo funciona un dron?

Un dron es capaz de moverse con cuatro grados de libertad, es decir, puede trasladarse en tres direcciones y rotar en una, para poder mantener un momento angular cero, el cual permite que el dron permanezca en un lugar fijo en lugar de rotar en alguna dirección, para que esto sea posible es necesario que dos de los motores de un dron giren en sentido de las manecillas del reloj y los otros dos motores giren en sentido contrario de las manecillas del reloj, tal y como se muestra en la Figura 23.



**Figura 23.** Imagen de cómo debe de girar un dron para mantener un momento angular cero.

Si movemos los cuatro motores de un dron a la misma velocidad generando una fuerza que supere el peso del dron, esto provocará que el dron se mueva hacia arriba, siendo este un movimiento de traslación. Así mismo, si lo que se quiere hacer es bajar el dron, se tiene que disminuir la velocidad de todos los motores para que la elevación sea menor que el peso del dron. Si lo que se quiere es que el dron flote en el aire, se tiene que cambiar la velocidad de los motores para que su elevación sea igual al peso del dron, por lo que permitirá que flote sobre el suelo.

La segunda traslación, es si queremos desplazar el dron hacia la derecha o hacia la izquierda. Para poder moverlo hacia la derecha se disminuirá la velocidad de los motores derechos y aumentaremos la de los izquierdos, así como se puede ver en la Figura 24. De la misma manera, si queremos mover el dron hacia la izquierda, se tendría que hacer lo inverso, es decir, reducir la velocidad de los motores de la izquierda y aumentar la velocidad de los de la derecha, así como se puede apreciar en la Figura 25.



**Figura 24.** Diferencia de velocidad entre los motores para desplazar el dron hacia la derecha.



**Figura 25.** Diferencia de velocidad entre los motores para desplazar el dron hacia la izquierda.

Finalmente, la tercera traslación es la que permite desplazar el dron hacia delante y hacia atrás. Se realiza con una secuencia similar a la del desplazamiento de derecha y de izquierda, pero con los motores delanteros y traseros del dron. Si se quiere desplazar el dron hacia adelante, es necesario disminuir la velocidad de los motores del frente y aumentar la de los de atrás, tal y como se aprecia en la Figura 26. Así mismo, al contrario, para desplazar el dron hacia atrás. Es decir, disminuir la velocidad de los motores traseros y aumentar la de los motores delanteros como se puede ver en la Figura 27.

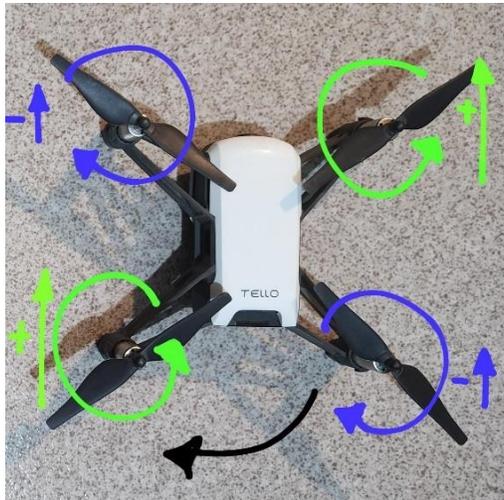


**Figura 26.** Diferencia de velocidad entre los motores delanteros y traseros para desplazar el dron hacia delante.

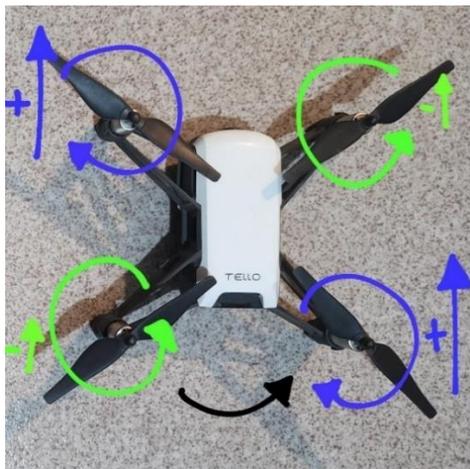


**Figura 27.** Diferencia de velocidad entre los motores delanteros y traseros para desplazar el dron hacia atrás.

Para poder rotar el dron en sentido de las manecillas del reloj, es necesario disminuir la velocidad de los motores que se mueven en el sentido de las manecillas del reloj y aumentar la velocidad de los motores que se están moviendo en sentido contrario de las manecillas del reloj, como se ve en la Figura 28. Para mover el dron en sentido contrario se tiene que disminuir la velocidad de los motores que giran en sentido contrario de las manecillas del reloj y aumentar la velocidad de los motores que giran en sentido de las manecillas, tal y como se ve en la figura 29.



**Figura 28.** Diferencia de velocidad entre los motores que giran en sentido de las manecillas de reloj y sentido contrario rotar el dron en sentido de las manecillas.



**Figura 29.** Diferencia de velocidad entre los motores que giran en sentido de las manecillas de reloj y sentido contrario rotar el dron en sentido contrario de las manecillas.

### 2.7.3. Uso de los drones

Una de las aplicaciones más utilizadas en la actualidad, es para la investigación de fauna y caracterización del hábitat, especialmente para recaudar información acerca de la distribución espacial de los animales donde las zonas de accesos son difíciles. También para el monitoreo de personas en diversos ambientes, ya que como pueden desplazarse con facilidad llegan a lugares complicados de acceder.

En la actualidad, los sistemas aéreos no tripulados que más se utilizan son pequeños drones que cuentan con cámaras fotográficas con un espectro visible o térmico, con una masa máxima al despegue de 2-25 kg aproximadamente, con una autonomía de vuelo de 30 minutos a 4 horas (ya hablando de los drones más sofisticados) y con alcances inferiores a 30 km por lo general (Mandujano, S. et al., 2017).

## 2.8. Normas para el vuelo de drones.

“NORMA Oficial Mexicana NOM-107-SCT3-2019, Que establece los requerimientos para operar un sistema de aeronave pilotada a distancia (RPAS) en el espacio aéreo mexicano (SECRETARIA DE COMUNICACIONES Y TRANSPORTES, 2019)”. Esta norma se puede encontrar en la página <https://www.sct.gob.mx/fileadmin/DireccionesGrales/DGAC-archivo/modulo2/nom-107-sct3-2019-201119.pdf> y únicamente se agregarán los puntos más importantes a continuación.

“La finalidad de esta norma es informar de forma detallada, acerca del procedimiento que se debe de llevar acabo para registrar por Internet un RPAS ante la Agencia Federal de Aviación Civil (AFAC) mediante su página electrónica”. (SECRETARIA DE COMUNICACIONES Y TRANSPORTES, 2019).

Además de que es importante destacar los siguientes puntos citados de la norma: NORMA Oficial Mexicana NOM-107-SCT3-2019.

“La Ley de Aviación Civil en su artículo 4 señala que la navegación civil en el espacio aéreo sobre territorio nacional se rige, además de lo previsto en dicha ley, por los tratados internacionales que los Estados Unidos Mexicanos ha celebrado, siendo el caso que México es signatario del Convenio sobre Aviación Civil Internacional firmado en la ciudad de Chicago, Illinois, Estados Unidos de América, el día 7 de diciembre de 1944”.

“Los Sistemas de Aeronaves Pilotadas a Distancia (RPAS) son un nuevo concepto en el ámbito aeronáutico, que la Autoridad Aeronáutica y la industria aeroespacial requieren comprender, definir e integrar para su adecuada operación. Estos sistemas se basan en novedades tecnológicas aeroespaciales de última generación, que ofrecen avances que pueden proporcionar nuevas y mejores aplicaciones civiles y de uso comercial, así como contribuir a mejorar la seguridad operacional y la eficiencia de toda la aviación civil”.

“La Norma Oficial Mexicana establece los requerimientos del Sistema de Aeronave Pilotada a Distancia o en inglés Remotely Piloted Aircraft System (RPAS) para operar dentro del espacio aéreo mexicano; de la misma manera para su comercialización en el territorio nacional.” (SECRETARIA DE COMUNICACIONES Y TRANSPORTES, 2019).

“El campo de aplicación va dirigido a toda persona física/moral, operadores de estado que pretendan operar u operen un RPAS; así mismo, aplica a los fabricantes y armadores de RPAS, personas físicas/morales que requieran importar RPAS a territorio nacional y a los comercializadores de RPAS.

Accidente. Todo suceso relacionado con la utilización de una aeronave pilotada a distancia, que ocurre entre el momento en que la aeronave está lista para desplazarse con el propósito de realizar un vuelo y el momento en que se detiene al finalizar el vuelo y se apaga su sistema de propulsión principal.

Aeródromo civil: área definida de tierra o de agua adecuada para el despegue, aterrizaje, acuatizaje o movimiento de aeronaves, con instalaciones o servicios mínimos para garantizar la seguridad de su operación.

Aeronave: Cualquier vehículo capaz de transitar con autonomía en el espacio aéreo con personas, carga o correo.

Aeronave autónoma: Aeronave no tripulada que no permite la intervención del piloto en la gestión del vuelo.

Certificado de matrícula: Documento que identifica y determina la nacionalidad de la aeronave.

Certificado de aeronavegabilidad: Documento oficial que acredita que la aeronave está en condiciones técnicas satisfactorias para realizar operaciones de vuelo.

Certificado de matrícula: Documento que identifica y determina la nacionalidad de la aeronave.

Comercializador de RPAS: Empresa que se encarga de mercadear un producto y/o servicio ya existente o manufacturado.

RPAS de uso Comercial: Sistema de aeronave pilotada a distancia destinado por el operador de RPAS a realizar tareas con fines de lucro.

RPAS de uso Recreativo: Es aquel sistema de aeronave pilotada a distancia que se destina por el operador de RPAS a la recreación.

RPAS de uso Privado No comercial: Sistema de aeronave pilotada a distancia, destinado por el operador de RPAS a realizar tareas sin fines de lucro.” (SECRETARIA DE COMUNICACIONES Y TRANSPORTES, 2019).

“Todo operador de RPAS, de conformidad a su peso máximo de despegue y uso, debe dar cumplimiento a lo establecido en la presente Norma Oficial Mexicana.

Todo operador de RPAS de estado, excepto las militares, debe dar cumplimiento a los requerimientos y limitaciones de los RPAS Privado No comercial, de acuerdo con su peso máximo de despegue, establecidos en la presente Norma Oficial Mexicana.” (SECRETARIA DE COMUNICACIONES Y TRANSPORTES, 2019).

“Todo operador de RPAS para investigación científica, debe dar cumplimiento a los requerimientos y limitaciones de los RPAS Privado No comercial, de acuerdo con su peso máximo de despegue, establecidos en la presente Norma Oficial Mexicana.” (SECRETARIA DE COMUNICACIONES Y TRANSPORTES, 2019).

RPAS Pequeño para uso Recreativo.

“El piloto debe operar el RPAS a una distancia de separación de al menos 9.2 Km (5 MN) de cualquier aeródromo. No exceder la velocidad máxima de operación establecida por el fabricante del RPAS; Operar la RPA a una altura máxima de 122 metros (400 ft); excepto en lo indicado en el inciso f) del presente numeral; No operar la RPA más allá de una distancia horizontal de 457 metros (1500 ft.) respecto al

piloto; Operar la RPA a una altitud máxima de 100 metros (328 ft), en el área comprendida entre los círculos de 9.2 Km (5 MN) y 18.5 Km (10 MN); Operar la RPA en condiciones meteorológicas libre de nubes y con la superficie a la vista; Mantener una visibilidad mínima de 1.5 Km. (0.8 MN) desde la localización de la estación de control, antes de iniciar la operación de la RPA.” (SECRETARIA DE COMUNICACIONES Y TRANSPORTES, 2019).

RPAS Pequeño para uso Privado No comercial o Comercial.

“No exceder la velocidad máxima de operación establecida por el fabricante del RPAS; Operar la RPA a una altura máxima de 122 metros (400 ft), excepto en lo indicado en el inciso e) del presente numeral; No operar la RPA más allá de una distancia horizontal de 457 metros (1500 ft.) respecto al piloto; Operar la RPA a una altura máxima de 100 metros (328 ft), en el área comprendida entre los círculos de 9.2 Km (5 MN) y 18.5 Km (10 MN), alrededor de los aeródromos listados en el numeral I.1 del Apéndice "I" Informativo de la presente Norma Oficial Mexicana; Operar la RPA en condiciones meteorológicas libre de nubes y con la superficie a la vista; Mantener una visibilidad mínima de 1.5 Km. (0.8 MN), desde la localización de la estación de control, antes de iniciar la operación de la RPA; No operar sobre personas, a menos que participen directamente en la operación de la RPA o estén situadas debajo de una estructura que les provea de una protección razonable en caso de desplome de la RPA. Salvo lo indicado en el numeral 8.3 de la presente Norma Oficial Mexicana; Mantener una distancia horizontal de seguridad perimetral con respecto a las personas no relacionadas con la operación de al menos 30 metros (98 ft), para los RPAS con un peso máximo de despegue mayor a 2 Kg y hasta 10 Kg o en casi contrario al menos 50 metros (164 ft), para los RPAS con un peso máximo de despegue mayor a 10 Kg y hasta 25 Kg.

El operador y/o piloto del RPAS es el responsable de su operación, uso y en caso de incidente o accidente, de los daños y/o lesiones causadas por la misma.

El operador del RPAS es el responsable del uso que se dé a la información obtenida durante la operación de la aeronave.

El operador y/o piloto del RPAS es el responsable de respetar todas las Leyes, Reglamentos y Normas de índole Federal o Local, relacionadas con Seguridad Nacional, Seguridad Pública, protección de la privacidad, propiedad intelectual, entre otras.

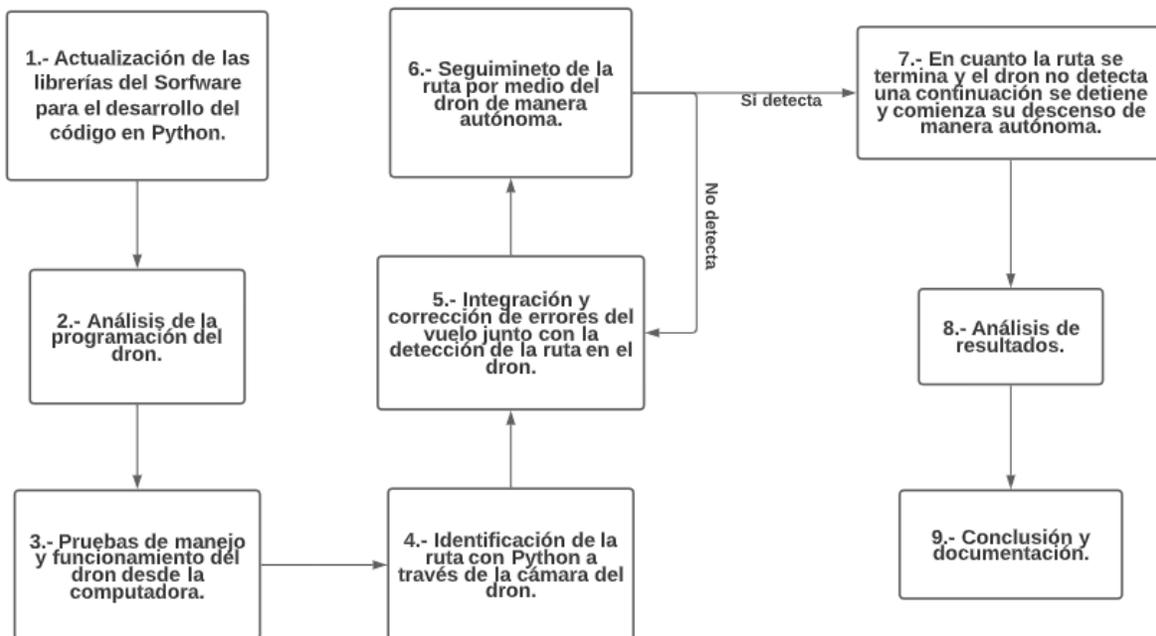
El piloto del RPAS no debe operar la RPA en estado de ebriedad o bajo los efectos de estupefacientes, psicotrópicos o enervantes”. (SECRETARIA DE COMUNICACIONES Y TRANSPORTES, 2019).

## Capítulo 3. Metodología

### 3.1. Metodología.

Considerando lo que se menciona en el manual de usuario del dron en las especificaciones de vuelo del dron (*Manual del usuario v1.0., 2018*), se trabajara en un lugar cerrado par atener un ambiente controlado del mismo.

Para el desarrollo del proyecto, se aplica la siguiente metodología representada con un diagrama de flujo, tal y como se puede ver en la Figura 30:

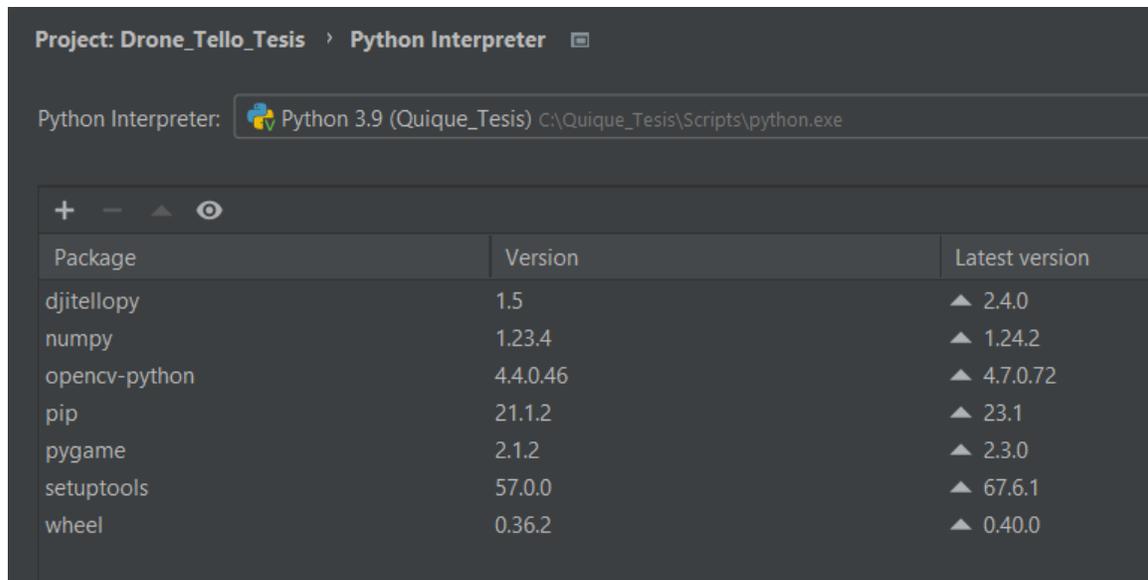


**Figura 30.** Diagrama de flujo de la metodología.

#### 3.1.1. Actualización de las librerías del Software para el desarrollo del código en Python.

Es necesario actualizar y tener las librerías necesarias para el manejo del dron a través de la computadora y de esta manera tener un control específico y fácil donde podamos programar los movimientos y lo necesario para que el dron pueda desplazarse sobre una ruta determinada de manera autónoma. También es necesario tener en cuenta con que versiones del software se va a trabajar, para no tener ningún inconveniente con la compilación o con el funcionamiento del dron.

Se puede ver en la Figura 31 cuáles son las librerías que se utilizan para la realización de este código, así como sus versiones correspondientes. Es importante mencionar que se utiliza Python 3.9 para este proyecto debido a que se intentó con la versión más reciente (Python 3.11, al momento de la redacción de este documento) y causó conflictos de compilación con algunas de las librerías que se requieren.



The screenshot shows the 'Python Interpreter' window in PyCharm for a project named 'Drone\_Tello\_Tesis'. It displays a list of installed packages with their current versions and the latest available versions. The packages listed are djitellopy, numpy, opencv-python, pip, pygame, setuptools, and wheel.

Package	Version	Latest version
djitellopy	1.5	▲ 2.4.0
numpy	1.23.4	▲ 1.24.2
opencv-python	4.4.0.46	▲ 4.7.0.72
pip	21.1.2	▲ 23.1
pygame	2.1.2	▲ 2.3.0
setuptools	57.0.0	▲ 67.6.1
wheel	0.36.2	▲ 0.40.0

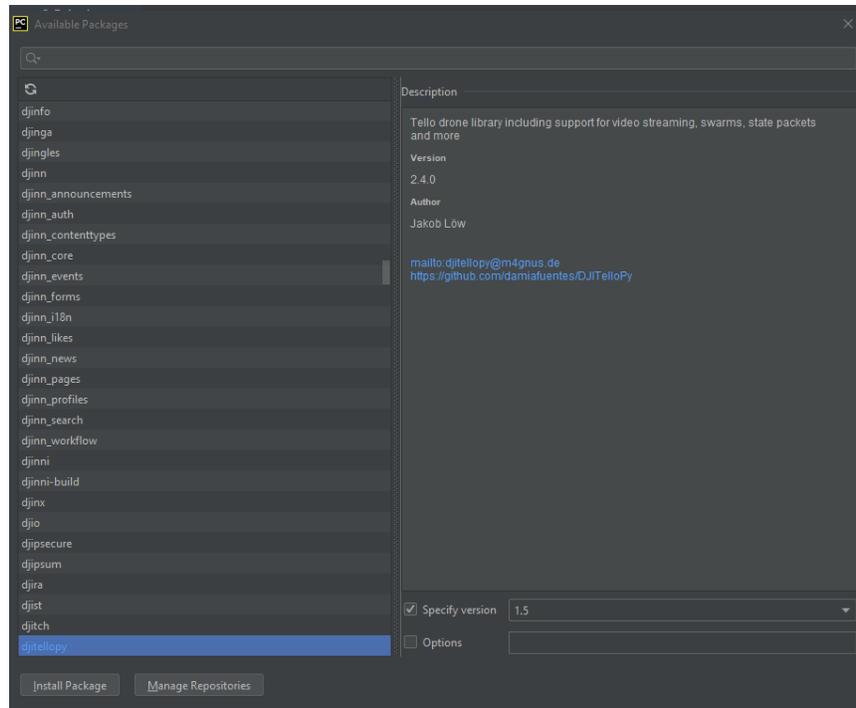
**Figura 31.** Versiones de las librerías y de Python utilizadas.

### 3.1.2. Análisis de la programación del dron.

Se trabaja con la información que ya existe para poder programar el vuelo de un dron y poder aplicar las funciones principales como despegue, aterrizaje, estabilización y desplazamiento en las 6 direcciones básicas que un dron puede desplazarse (arriba, abajo, izquierda, derecha, adelante y atrás) por medio de programación en Python, así como el funcionamiento de la cámara y lo necesario para obtener la ruta a través de la cámara y de esta manera el dron pueda seguir la ruta.

Esta información se encuentra dentro del paquete “djitellopy”, el cual contiene la implementación de todos los comandos de Tello, la recuperación sencilla de una transmisión de video y mandar y transmitir información, la cual es necesaria para el funcionamiento del dron, una de las librerías más importantes es la “tello.py” que contiene todos los comandos de Tello.

Este paquete que se utiliza para interactuar con la interfaz de Python de Drones DJI Tello la cual es desarrollada basándose en el [Tello SDK](#) oficial y el [Tello EDU SDK](#), Este paquete es desarrollado por Jakob Löw y Damià Fuentes Escoté como principales desarrolladores, por lo menos en la versión 1.5 de “djitellopy”. Esta versión es la que se utiliza para la implementación de este proyecto. Esta información se obtiene del Pycharm al momento de descargar los paquetes necesarios, tal y como se puede ver en la Figura 32. Debido a que es Open Source, es posible que se actualice este tipo de bibliotecas.

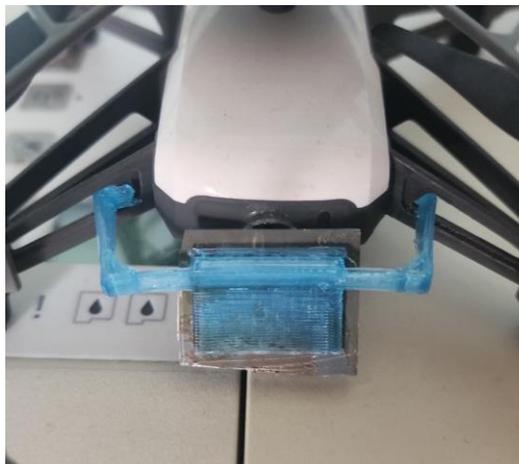


**Figura 32.** información acerca del paquete “djitellopy”.

### **3.1.3. Pruebas de manejo y funcionamiento del dron desde la computadora.**

Con la información que se encuentra en el marco teórico respecto a la programación del dron, los comandos necesarios y su funcionamiento para el manejo del dron Tello y del IDE de Python “PyCharm Community Edition”, se realizará el desarrollo del código que se usa para este proyecto.

Para lograr que el dron tome imágenes del suelo cuando el dron está en vuelo, fue necesario diseñar una pieza con espejo que se coloca frente a la cámara del dron. En la Figura 33 se muestra la pieza utilizada junto con el espejo que se usó montadas en el dron.



**Figura 33.** Imagen de la pieza montada en el dron

Hasta este paso se implementó el manejo del dron de manera manual, es decir, con comandos específicos ingresados por el usuario para hacer sus desplazamientos, y de esta manera probar que todo sea funcional, para posteriormente programar el código, el cual cumpla el objetivo de este proyecto.

#### **3.1.4. Identificación de la ruta con Python a través de la cámara del dron.**

Con ayuda de una cámara que el dron trae incorporada y la información que se recaudó se realizara la identificación de objetos por medio de Python, es decir, se procesara la imagen digitalmente para posteriormente utilizarla la detección de la ruta establecida. Esto para tener una vista de por dónde está pasando el dron y donde es necesario detenerse o pueda hacer un filtro entre lo que sí es parte de la ruta y lo que no.

Para lo anterior se implementa la siguiente estrategia:

- Iniciar el vuelo del dron y activación de la cámara;
- Umbralizar la ruta que se quiere seguir y que se detecte con la cámara del dron;
- Iniciar el desplazamiento del dron sobre la ruta que se detectó con ayuda de la umbralización y la delimitación de la ruta. Para la delimitación se utiliza la detección del contorno que solo es la ruta umbralizada;
- Continuar con el desplazamiento del dron sobre la ruta, hasta que ya no detecte más continuación y posterior a eso comenzar con el descenso.

#### **3.1.5. Integración y corrección de errores del vuelo junto con la detección de la ruta en el dron.**

Como ya se mencionó, se trabajó de manera manual con el manejo del dron para tener identificado como es que se puede desplazarse en cualquier dirección sin ningún tipo de inconvenientes, para posteriormente desarrollar un programa donde el dron sea capaz de distinguir una ruta por la cual se desplazara con ayuda de la cámara que trae integrada, y así mismo pueda realizar el vuelo de manera autónoma.

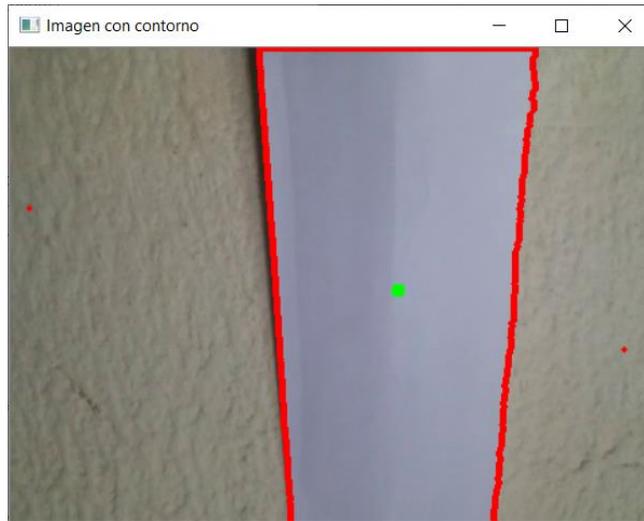
Como primera prueba se desplaza sobre una ruta únicamente recta, esto con el propósito de que solo se trabaje con el desplazamiento del dron sin ningún tipo de giro, donde se involucra el procesamiento de imágenes y el filtro para distinguir únicamente la ruta. Como paso secundario es hacer que el dron pueda seguir rutas no sola una línea recta, sino también con giros o curvaturas.

Así mismo, para que el dron tuviera una precisión muy buena y un seguimiento de la ruta ideal, se tuvo que realizar un ajuste y corrección del código para mejorar el funcionamiento, ya que se presentaron defectos como la incorrecta detección la ruta y problemas para lo mismo.

#### **3.1.6. Seguimiento de la ruta por medio del dron de manera autónoma.**

En esta etapa se tomará una imagen de la cámara del dron, la cual es procesada con ayuda de Python para identificar si se encuentra la ruta o no y de este modo el

dron pueda desplazarse por toda la ruta que sea detectada. Así como se muestra en la Figura 34.



**Figura 34.** Imagen tomada con el dron de la detección de la ruta.

### **3.1.7. Freno y descenso de manera autónoma.**

Una vez que la ruta se termine y el dron no detecte más ruta sobre la que pueda desplazarse se detendrá y comienza su descenso de manera autónoma.

### **3.1.8. Análisis de resultados.**

En esta etapa se reportan los resultados obtenidos en base a la metodología presentada. A partir de esta etapa el resto de la metodología se muestra en la siguiente capítulo.

## **3.2. Materiales.**

### **3.2.1. Software y Hardware**

A continuación, se lista el Hardware utilizado:

- Computadora Dell con Windows 10
- Dron de la marca Tello power by DJI

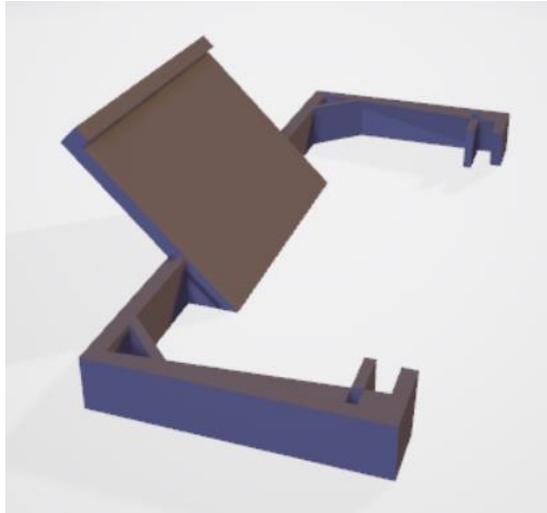
A continuación, se muestra el Entorno de Desarrollo Integrado o IDE por sus siglas en inglés que se usó para la programación del dron:

- PyCharm Community Edition 2021.2

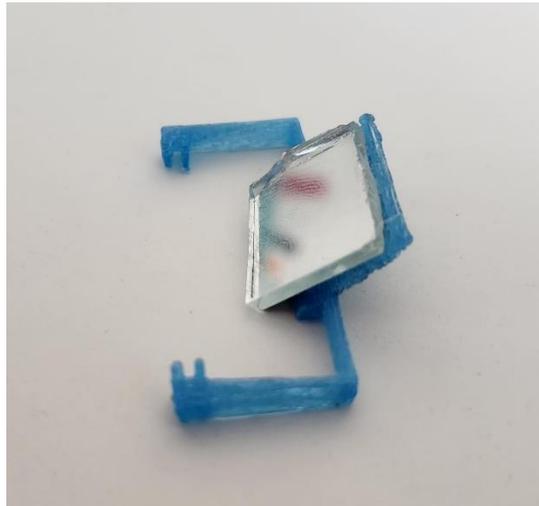
A continuación, se muestra el lenguaje de programación que se utilizó y su versión:

- Python 3.9

Componente extra que se le añadió al dron para que la visión de la cámara, la cual ya la trae incluida el dron, fuese capaz de dirigirse hacia el piso y poder hacer el seguimiento de la ruta de manera correcta. El diseño de la pieza se puede ver en la Figura 35 y en la Figura 36 se ve la pieza ya impresa.



**Figura 35.** Imagen de la simulación de la pieza que se fabricó para una visión hacia el piso por parte de la cámara del dron.



**Figura 36.** Imagen de la pieza impresa en 3D.

Además, se le incluyo una jaula protectora al dron para evitar que se dañara con el impacto accidental de algún objeto no deseado durante el vuelo de este, la jaula se puede ver en la Figura 37



**Figura 37.** Jaula protectora para el dron Tello.

Además, se agrega una imagen de como quedo el dron después de los accesorios que se le integraron para poder realizar el proyecto, la imagen se muestra en la Figura 38.



**Figura 38.** Imagen del dron que se utilizó para este trabajo.

# Capítulo 4. Resultados

## 4.1. Proceso de programación.

### 4.1.1. Funciones y movimientos principales.

Para el desarrollo de este proyecto se comenzó analizando los movimientos y funciones básicas que se tienen en la librería “tello” la cual es extraída del paquete “djitellopy”. Estos movimientos y funciones principales son el desplazamiento en las 6 direcciones en las que se puede mover el dron despegar y aterrizar, el porcentaje de batería que le queda al dron, la altura a la que está elevado y la velocidad de vuelo, por mencionar algunas de las funciones más importantes.

Es importante mencionar que los comandos que se encuentran dentro de la librería “tello” están diseñados con base en los requerimientos del fabricante y con los estándares que el mismo fabricante pone para su óptimo funcionamiento, es decir en al [Tello SDK](#) oficial y el [Tello EDU SDK](#).

### 4.1.2. Transmisión de video y conexión del dron.

Lo primero que se debe de hacer es establecer la conexión de la PC con el dron y posteriormente empezar la transmisión del video, no para comenzar una grabación, sino para empezar a visualizar lo que se ve a través de la cámara del dron. Existen comandos específicos para realizar su conexión, los cuales son llamados por las funciones que se ven en la Figura 39.

```
def streamon(self):
    """Set video stream on. If the response is 'Unknown command' means you have to update the Tello firmware. That
    can be done through the Tello app.
    Returns:
        bool: True for successful, False for unsuccessful
    """
    result = self.send_control_command("streamon")
    if result is True:
        self.stream_on = True
    return result

def connect(self):
    """Entry SDK mode
    Returns:
        bool: True for successful, False for unsuccessful
    """
    return self.send_control_command("command")
```

**Figura 39.** Estructura para enviar la transmisión de video y conexión con el dron.

## 4.2. Explicación de los códigos más relevantes.

Los movimientos más importantes y principales para que el dron pueda desplazarse son; las 4 direcciones bidimensionales básicas, arriba, abajo y rotación sobre su propio eje, estos movimientos son los que se requieren para que el dron pueda desplazarse sobre la ruta sin ningún inconveniente, así como el poder suspenderse en el aire y hacer el descenso hacia el suelo cuando ya está en el aire el dron. Es importante poder manipular estos movimientos principales para el correcto funcionamiento del dron.

### 4.2.1. Desplazar el dron.

Para poder desplazar el dron en las cuatro direcciones bidimensionales (adelante, atrás, izquierda y derecha), arriba y abajo, y realizar el giro, es decir, la rotación sobre su propio eje vertical, se utiliza una estructura la cual manda un intervalo de valores entre -100 y 100 para indicar la velocidad a la que los motores deben de girar. Por ejemplo, para comenzar a desplazarse hacia adelante el valor debe de ser igual o mayor a 1 hasta 100 y cuando se quiere desplazar hacia atrás es igual o mayor a -1 hasta -100, este intervalo indicara la velocidad de desplazamiento y así cuando se ingrese en la estructura y la función se mande a llamar este responderá según sea su valor ingresado. De la misma manera para ir a la izquierda o derecha, arriba o abajo, y la rotación sobre su propio eje vertical.

La estructura que se usa con los valores que se le asignan y hace el desplazamiento del dron se puede ver en la Figura 40:

```
@accepts(left_right_velocity=int, forward_backward_velocity=int, up_down_velocity=int, yaw_velocity=int)
def send_rc_control(self, left_right_velocity, forward_backward_velocity, up_down_velocity, yaw_velocity):
    """Send RC control via four channels. Command is sent every self.TIME_BTW_RC_CONTROL_COMMANDS seconds.
    Arguments:
        left_right_velocity: -100-100 (left/right)
        forward_backward_velocity: -100-100 (forward/backward)
        up_down_velocity: -100-100 (up/down)
        yaw_velocity: -100-100 (yaw)
    Returns:
        bool: True for successful, False for unsuccessful
    """
    if int(time.time() * 1000) - self.last_rc_control_sent < self.TIME_BTW_RC_CONTROL_COMMANDS:
        pass
    else:
        self.last_rc_control_sent = int(time.time() * 1000)
        return self.send_command_without_return('rc %s %s %s %s' % (left_right_velocity, forward_backward_velocity,
                                                                    up_down_velocity, yaw_velocity))
```

Figura 40. Estructura que se usa para desplazar en dron.

### 4.2.2. Suspender el dron en el aire.

Para poder detener el dron y que se mantenga estático en el aire, es decir, sin que descienda, es necesario mandar ceros en la estructura que hacer que el dron se mueva en las cuatro direcciones posibles (adelante, atrás, izquierda y derecha), la cual se muestra en la Figura 39, por lo tanto, el dron se mantiene estático en el aire detectando el fin de la ruta durante unos segundos. Cuando se utiliza comando con

los valores que requiere la función manda a llamar a la estructura y este tomar los valores que están en el comando, el comando utilizado es el siguiente:

```
--- dron.send_rc_control(0, 0, 0, 0);
```

### 4.2.3. Detección de la ruta.

Para poder distinguir la ruta que se va a seguir es necesario hacer una umbralización y distinguir la ruta como un objeto único de todo lo demás que puede procesarse a través de la cámara del dron. Para esto se usa el modelo HSV (por sus siglas en inglés) para poder segmentarla en Tonalidad, Saturación y Valor y poder modificar sus valores para que únicamente se distinga la ruta y no el fondo de la imagen u otros objetos que no nos interesen en este momento. Se decidió usar el modelo de HSV debido a su versatilidad y la gama de variaciones que se pueden adquirir.

Para esto es necesario procesar la imagen que esté transmitiendo con el dron, donde se tiene la ruta a seguir y mover los valores de tonalidad, saturación y valor para poder distinguir únicamente la ruta de esta imagen, una vez obteniendo los valores para que solo se vea la ruta se guardan en un arreglo y se imprimen para poder ingresarlos a nuestro condigo principal y los valores HSV de la ruta. La implementación en código para la adquisición de los valores de HSV se puede apreciar en la Figura 41.

```
def empty(a):
    pass

cv2.namedWindow("HSV")
cv2.resizeWindow("HSV", 640, 240)
cv2.createTrackbar("HUE Min", "HSV", 0, 179, empty)
cv2.createTrackbar("HUE Max", "HSV", 179, 179, empty)
cv2.createTrackbar("SAT Min", "HSV", 0, 255, empty)
cv2.createTrackbar("SAT Max", "HSV", 255, 255, empty)
cv2.createTrackbar("VALUE Min", "HSV", 0, 255, empty)
cv2.createTrackbar("VALUE Max", "HSV", 255, 255, empty)
cap = cv2.VideoCapture(1)
frameCounter = 0

while True:
    imagen = drone.get_frame_read().frame
    # imagen = cap.read()
    imagen = cv2.resize(imagen, (FrameWidth, FrameHeight))
    #imagen = cv2.flip(imagen, 0)
    imagenHSV = cv2.cvtColor(imagen, cv2.COLOR_BGR2HSV)

    h_min = cv2.getTrackbarPos("HUE Min", "HSV")
    h_max = cv2.getTrackbarPos("HUE Max", "HSV")
    s_min = cv2.getTrackbarPos("SAT Min", "HSV")
    s_max = cv2.getTrackbarPos("SAT Max", "HSV")
    v_min = cv2.getTrackbarPos("VALUE Min", "HSV")
    v_max = cv2.getTrackbarPos("VALUE Max", "HSV")

    lower = np.array([h_min, s_min, v_min])
    upper = np.array([h_max, s_max, v_max])
    mask = cv2.inRange(imagenHSV, lower, upper)
    result = cv2.bitwise_and(imagen, imagen, mask=mask)
    print(f'[{h_min}, {s_min}, {v_min}, {h_max}, {s_max}, {v_max}]')

    mask = cv2.cvtColor(mask, cv2.COLOR_GRAY2BGR)
    hStack = np.hstack([imagen, mask, result])
    cv2.imshow('Stacking', hStack)
    if cv2.waitKey(1) and 0xFF == ord('q'):
        break
```

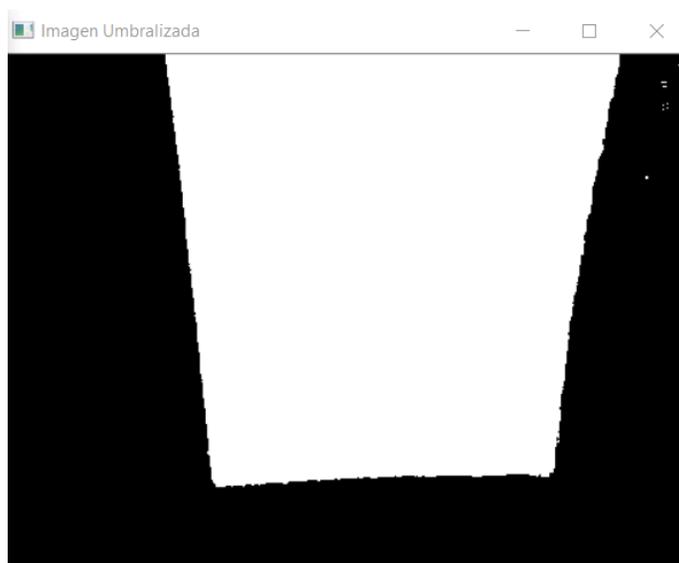
Figura 41. Código para la obtención de los valores HSV.

Los valores que se guardan en un arreglo son tomados por la función que se muestra en la Figura 42, para cuando se mande a llamar la función poder integrarlos al código principal y usarla para que el dron se puede desplazar sobre la ruta.

```
def umbralizacion(imagen):  
    hsv = cv2.cvtColor(imagen, cv2.COLOR_BGR2HSV)  
    lower = np.array([hsvVals[0], hsvVals[1], hsvVals[2]])  
    upper = np.array([hsvVals[3], hsvVals[4], hsvVals[5]])  
    mask = cv2.inRange(hsv, lower, upper)  
    return mask
```

**Figura 42.** Umbralización la ruta para distinguirla.

En la siguiente imagen se puede ver la umbralización que se le da a la ruta y que es lo que se visualiza y toma en cuenta únicamente para considerarlo ruta. La imagen original se puede apreciar en la Figura 45.



**Figura 43.** Imagen de la ruta umbralizada capturada por el dron.

#### **4.2.4. Obtención de los contornos de la ruta.**

Para la obtención del contorno de la ruta primero se obtiene la ruta que se va a delimitar, esto para poder resaltarla con líneas que indique visualmente la ruta. Posteriormente, con ayuda de una función para distinguir la esquina de una imagen y poder trazar el contorno, adicionalmente se quiere conocer el centro de la imagen, por lo tanto, por medio de una fórmula básica de matemáticas para obtener el centroide de la figura resaltada con líneas (la ruta) y poder visualizar tanto el contorno como el centro cuando se mande a llamar la función, la cual se presenta en la Figura 44.

```
def ObtenerContornos(ImgUmbralizada, imagencontorno):  
    contornos, jerarquia = cv2.findContours(ImgUmbralizada, cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_NONE)  
    biggest = max(contornos, key=cv2.contourArea)  
    x, y, w, h = cv2.boundingRect(biggest)  
    cx = x + w // 2  
    cy = y + h // 2  
    cv2.drawContours(imagencontorno, contornos, -1, (0, 0, 255), 3)  
    cv2.circle(imagencontorno, (cx, cy), 5, (0, 255, 0), cv2.FILLED)  
  
    return cx, cy
```

**Figura 44.** Función para marcar el contorno de la ruta.

El procesamiento de la imagen se hace dentro de la computadora por medio de la cámara del dron, la cual captura la imagen, detecta la ruta, umbraliza la misma ruta para ponerle un perímetro sobre la ruta y poder desplazarse de manera autónoma. A continuación, se puede ver en la Figura 45, cómo se visualiza la ruta con los contornos trazados sobre la imagen que se está procesando del dron, tanto la imagen original que el dron toma, como los contornos de la ruta sobre la que se desplaza.



**Figura 45.** Imagen de la ruta original y los contornos de la detección.

#### **4.2.5. Descenso del dron al suelo.**

Para el descenso se utiliza una estructura la cual cuando manda a llamar el comando para hacer que descienda el dron, esto cuando el dron deja de detectar una, la continuación de una ruta a seguir y comienza el descenso. El comando que se utiliza se aprecia en la Figura 46.

```
def land(self):  
    """Tello auto land  
    Returns:  
        bool: True for successful, False for unsuccessful  
    """  
    return self.send_control_command("land")
```

Figura 46. Estructura que se utiliza para aterrizar el dron.

### 4.3. Explicación de la técnica para identificación de la ruta.

Para la identificación de la ruta se realizó un umbralización de la imagen para que únicamente se distinguiera la ruta que será recorrida por el dron. Al mismo tiempo se traza un contorno sobre la imagen umbralizada la cual no indica los límites de la ruta. Además, se agrega el centro de lo que la cámara está detectando como ruta, esto para que el dron pueda estar centrado sobre la ruta durante todo su desplazamiento.

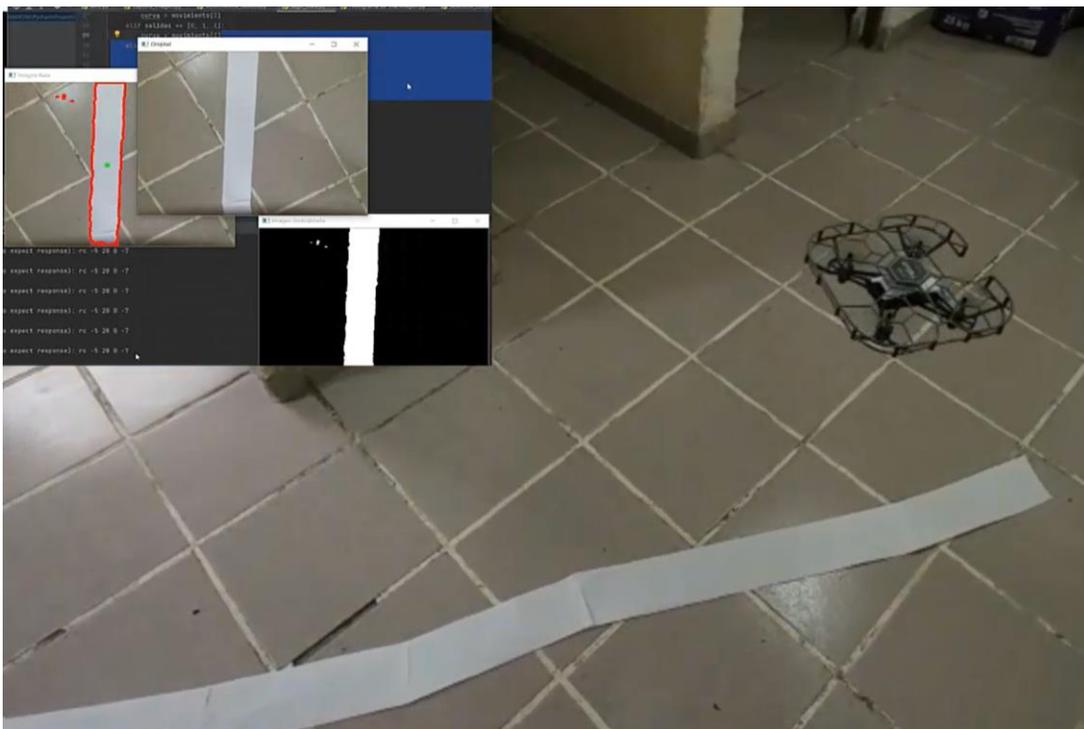
Adicionalmente, se integran unas condiciones en la imagen umbralizada para que cuando se detecte más de un tercio en los costados de la ruta se puede fijar en el centro de la ruta el dron, esto como medida de seguridad para que no pierda la ruta el dron.

El desplazamiento del dron se mantiene constante durante su recorrido, a menos que se deje de detectar la ruta o sea interrumpida. En este caso se detendrá el dron y comenzará su descenso, ya sea por falta de ruta (el fin del camino) o incluso por una detección excesiva de elementos que por factores como el brillo o la luz directa reflejada en el piso sean detectados como parte de la ruta, es decir que los valores HSV que se declaran para al inicio para la umbralización de la ruta sean iguales o similares al suelo o a otro elemento que se encuentre dentro de la toma aérea del dron entre 2 y 4 segundos continuos. Esto se puede apreciar en la Figura 47.



**Figura 47.** Imagen de una detección de un elemento externo a la ruta.

En la Figura 48 se puede ver al dron detectando y desplazándose sobre la ruta trazada en el piso por medio de una hoja de papel. Dentro de lo que el dron procesa por medio de la cámara la ruta segmentada del resto de lo que detecta para que pueda trazarse una circunferencia sobre la ruta.



**Figura 48.** Imagen del dron cuando la ruta no es recta.

Es importante mencionar que por fines prácticos al momento de la grabación que se le hizo al dron cuando estaba desplazándose sobre la ruta se estableció la velocidad del dron en un 20 por ciento de la velocidad máxima que tiene el dron

## Capítulo 4. Resultados

esto para que se pudiera ver mejor la corrección autónoma que hacía el dron al momento de desplazarse, sin embargo también se probó a una velocidad del 70 por ciento y la respuesta fue la misma en cuestión de que el dron siguió correctamente la ruta de manera autónoma; sin embargo, la corrección de su trayectoria fue más rápido y en el video no se podría apreciar, por eso se disminuyó su velocidad.

Para que el dron pudiera aumentar o disminuir su velocidad se utilizaba el comando que se ve en la Figura 49.

```
def set_speed(self, x):  
    """Set speed to x cm/s.  
    Arguments:  
        x: 10-100  
  
    Returns:  
        bool: True for successful, False for unsuccessful  
    """  
    return self.send_control_command("speed " + str(x))  
  
last_rc_control_sent = 0
```

**Figura 49.** Comando utilizado para disminuir o aumentar la velocidad del dron en cm/s.

En la Figura 50 se puede ver que la ruta no es totalmente recta, y cuando el dron salga del centro de la imagen se intentara regresar al centro de la imagen, es decir, el dron comienza hacer una corrección de vuelo para poder centrar la ruta en medio de la toma aérea que está capturando el dron.



**Figura 50.** Imagen del dron detectando y trasladándose sobre la ruta.

## Capítulo 5. Conclusiones

Se analizó toda la información obtenida, como es, el vuelo del dron, el procesamiento de imágenes, la detección de la ruta, la eficiencia y el manejo de vuelo que el dron tiene al momento de detectar la ruta y desplazarse sobre ella y del mismo modo saber que tan bien lo hizo. Haciendo una comparativa de la teoría entre un dron de arquitectura cerrada que realiza actividades similares y el dron que nosotros programamos para este proyecto, se puede decir que es mejor trabajar con una arquitectura abierta, ya que puede implementar código propio y poner valores y características específicas para solucionar problemas de detección, desplazamiento, trayectoria de vuelo o parámetros como su velocidad y estabilidad de vuelo. Por esto, si lo que se quiere es trabajar con fines experimentales y saber cómo y porque se puede desplazar y detectar imágenes específicas, el dron es mejor usar una arquitectura abierta o programación del dron con código abierto.

Se realizó una documentación acerca de todo lo que se trabajó con el dron y de las actividades que se realizaron, así como de los resultados que se obtuvieron del mismo, desde la teoría y el cómo y porque funciona el dron y su detección, hasta los resultados finales del objetivo de este proyecto.

Se hace una redacción de manera clara y concisa, la cual contendrá todo lo que se trabajó a lo largo del proyecto, como es que funciona el dron, ventajas y desventajas de este, cuanto tiempo tardo en realizar las actividades programadas, así como el tiempo realizarse el proyecto en su totalidad y qué utilidad se le puede dar en la vida diaria.

Adicionalmente, es importante mencionar que este proyecto puede continuar, es decir, se le puede dar un seguimiento al trabajo, ya que una vez que el dron ya detecto la ruta y la puede seguir de manera autónoma, podemos continuar con el siguiente paso. En este paso la detección de obstáculos y si la ruta es interrumpida no se detenga el dron o comience su descenso, sino que pueda continuar su camino y pueda seguir desplazándose.

Finalmente, es importante mencionar que el dron logra trasladarse a través de una ruta establecida previamente con hojas de papel color blanco, utilizando un lenguaje de programación Open Source. Lo anterior es para que pudiera terminar el desplazamiento de la ruta de inicio a fin, tal y como se esperaba en los objetivos generales de este proyecto. Además, es necesario mencionar que la elección de esta marca de dron fue debido a su precio y facilidad para programarlo ya que es Open Source, su arquitectura y con la información que proporciona el fabricante.

Se estudió como es que el dron funciona, la forma de comunicarse con el mismo y el procesamiento de imagen para la detección de la ruta. El procesamiento de la imagen se hace a través de la cámara del dron y por medio de la conexión vía WiFi llega a la computadora, la cual es la encargada de procesar la imagen, umbralizar y trazar la trayectoria, además de poder detectar la ruta y poder desplazar el dron de manera autónoma, tanto su despegue como su aterrizaje, tal y como se esperaba en los objetivos específicos del proyecto.

## Capítulo 5. Conclusiones

Por último, me gustaría mencionar que adicionalmente a este trabajo se realizó un artículo acerca de este proyecto, del cual se realizó una conferencia presentación del artículo en el congreso nacional de mecatrónica, el cual dio fecha y lugar el sábado 6 de mayo del 2023 en Morelia, se anexa la carta de aceptación del artículo para el congreso.

# Referencias

Abesinghe, A. M. S. K., KSP, A., Ekanayake, E. M. A. C., Kavindi, M. A. R., & Mohotti, A. J. (2020). DEVELOPING A SELECTIVE TEA PLUCKING MECHANISM USING IMAGE PROCESSING FOR A DRONE-BASED TEA HARVESTING MACHINE. *GSJ*, 8(11).

Aguirre Dobernack, N. (2013). IMPLEMENTACIÓN DE UN SISTEMA DE DETECCIÓN DE SEÑALES DE TRÁFICO MEDIANTE VISIÓN ARTIFICIAL BASADO EN FPGA. Proyecto Final de Carrera. Sevilla: Escuela Superior de Ingenieros. Universidad de Sevilla.

Agustí Melchor, M. (2019). Introducción a la detección de puntos característicos con OpenCV.

Alcón Clemente, J. (2019). Design and developement of an aplicacion for simulating, visualizing and comparing trajectories of Tello drones (Bachelor's thesis, Universitat Politècnica de Catalunya).

Angulo, J., & Serra, J. (2005). Segmentación de imágenes en color utilizando histogramas bi-variables en espacios color polares luminancia/saturación/matiz. *Computación y Sistemas*, 8(4), 303-316.

Arévalo, V., González, J., & Ambrosio, G. (2004). La librería de visión artificial opencv. aplicación a la docencia e investigación. *Base Informática*, 40, 61-66.

Atienza Vanacloig, V. L. (2011). Digitalización de imágenes con ayuda del histograma.

Bennett, M. K., Younes, N., & Joyce, K. (2020). Automating drone image processing to map coral reef substrates using google earth engine. *Drones*, 4(3), 50.

Blázquez, J. P. (2015). Introducción a los sistemas de comunicación inalámbricos. Universitat Oberta de Catalunya. [https://www.exabyteinformatica.com/uoc/Informatica/Tecnologia\\_y\\_desarrollo\\_en\\_dispositivos\\_moviles/Tecnologia\\_y\\_desarrollo\\_en\\_dispositivos\\_moviles\\_\(Modulo\\_1\).pdf](https://www.exabyteinformatica.com/uoc/Informatica/Tecnologia_y_desarrollo_en_dispositivos_moviles/Tecnologia_y_desarrollo_en_dispositivos_moviles_(Modulo_1).pdf)

Calderón Jácome, M. T., & Maldonado Andrade, D. J. (2014). Control por visión de un cuadricóptero utilizando ROS (Bachelor's thesis, Quito: EPN, 2014.).

Castañeda, W. (2005). Color. Universidad de Caldas. <https://books.google.es/books?hl=es&lr=&id=VIWVK1LkDM0C&oi=fnd&pg=PA7&dq=que+es+el+color+&ots=p0Ny18JP8L&sig=jWo2sQPBGnYV9jRxBQcluoiRjaA#v=onepage&q=ue%20es%20el%20color&f=false>

Cattaneo, C. A., Larcher, L. I., Ruggeri, A. I., Herrera, A. C., & BIASONI, E. M. (2011). Métodos de umbralización de imágenes digitales basados en entropía de shannon y otros. *Mecánica Computacional*, 30(36), 2785-2805.

Challenger-Pérez, I., Díaz-Ricardo, Y., & Becerra-García, R. A. (2014). El lenguaje de programación Python. *Ciencias Holguín*, 20(2), 1-13. <https://www.redalyc.org/pdf/1815/181531232001.pdf>

Clark, Bryan y Walton, Timothy. (2020). El Radar. Conquista del espectro electromagnético: El Pentágono revela una nueva estrategia. Recuperado de

## Referencias

<https://www.elradar.es/conquista-del-espectro-electromagnetico-el-pentagono-revela-una-nueva-estrategia/>

Coñapes, S. y Cerro, Y. (2015). "Tipos De Redes inalámbricas". *Redes Inalámbricas*. <<https://sites.google.com/site/redesinalambricas3/tipos-de-redes-inalambricas>> [Consulta: 15 de octubre de 2021].

Covantec R.L. (2018). *Programación en Python - Nivel básico*.

Cuevas, E., Zaldívar, D., & Pérez, M. (2016). *Procesamiento digital de imágenes con MATLAB & Simulink*. Ra-Ma.

Di Leo, Néstor C. (2015). *Drones: nueva dimensión de la teledetección agroambiental y nuevo paradigma para la agricultura de precisión*. <http://rephip.unr.edu.ar/bitstream/handle/2133/13373/2AM41.pdf?sequence> [Consulta: 1 de septiembre de 2022].

Elizondo, J. E., & Maestre, L. P. (2005). *Fundamentos de procesamiento de imágenes*. Mexicali: Universidad Autónoma de Baja California. [https://d1wqtxs1xzle7.cloudfront.net/35915704/FundamentosDeProcesamientoDeImagenes-with-cover-page-v2.pdf?Expires=1661051217&Signature=RX41DEklURtwlRTZyu-i3BcrxGVnj1QTZEbZvBq4taNPoX~XtHdtEKULxKi1UEcRHF9b1xcHi~rDrMZNPCMQG3fe5u0e8aqPE100i~Lv4K5V0skZe-ym5i4teSAJSn9JbVREkHjnzfF-b7QJZpt8MSYwmtZu2a~NMG5rDsXMH1e0yu-yGsTwx3XPoTDf1zl9pcz4qeOJ3C0qsX76j5teaEoojRxqz1DMyFCgwraH5StZTC-1D0GZ9JFbShvNCZ2JfToDA~mncX9evD3NjlfNUAsS37EWN9STI3mcJvo448jqVPgfULNnrWhl5l0i27qgeWfTgQQRb9EvRAXmMHXhg\\_&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA](https://d1wqtxs1xzle7.cloudfront.net/35915704/FundamentosDeProcesamientoDeImagenes-with-cover-page-v2.pdf?Expires=1661051217&Signature=RX41DEklURtwlRTZyu-i3BcrxGVnj1QTZEbZvBq4taNPoX~XtHdtEKULxKi1UEcRHF9b1xcHi~rDrMZNPCMQG3fe5u0e8aqPE100i~Lv4K5V0skZe-ym5i4teSAJSn9JbVREkHjnzfF-b7QJZpt8MSYwmtZu2a~NMG5rDsXMH1e0yu-yGsTwx3XPoTDf1zl9pcz4qeOJ3C0qsX76j5teaEoojRxqz1DMyFCgwraH5StZTC-1D0GZ9JFbShvNCZ2JfToDA~mncX9evD3NjlfNUAsS37EWN9STI3mcJvo448jqVPgfULNnrWhl5l0i27qgeWfTgQQRb9EvRAXmMHXhg_&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA)

Enrique Alegre, Gonzalo Pajares y Arturo de la Escalera. (2016). *Conceptos y métodos en Visión por Computador*. Grupo de Visión del Comité Español de Automática (CEA). <http://intranet.ceautomatica.es/sites/default/files/upload/8/files/ConceptosyMetodosenVxC.pdf>

GALIPIENSO, A., ISABEL, M., Cazorla Quevedo, M. A., Colomina Pardo, O., ESCOLANO RUIZ, F. R. A. N. C. I. S. C. O., & LOZANO ORTEGA, M. A. (2003). *Inteligencia artificial: modelos, técnicas y áreas de aplicación*. Editorial Paraninfo. [Consulta: 12 de diciembre de 2022].

Gallacher, D. (2016). *Drone applications for environmental management in urban spaces: A review*. *International Journal of Sustainable Land Use and Urban Planning*, 3(4).

Group, B. Julio 9, (2019). *Drones, su evolución a través de la historia*. Besttechnology Group. <<https://normas-apa.org/referencias/citar-un-blog/>> [Consulta: 15 de octubre de 2021].

Guan, Y., Zhou, F., & Zhou, J. (2019, November). *Research and Practice of Image Processing Based on Python*. In *Journal of Physics: Conference Series* (Vol. 1345, No. 2, p. 022018). IOP Publishing.

Guillen, G. (2019). *Digital Image Processing with Python and OpenCV*. In *Sensor Projects with Raspberry Pi* (pp. 97-140). Apress, Berkeley, CA.

## Referencias

Gur, A., Erim, M., & Karakose, M. (2020, October). *Image Processing Based Approach for Crime Scene Investigation Using Drone*. In *2020 International Conference on Data Analytics for Business and Industry: Way Towards a Sustainable Economy (ICDABI)* (pp. 1-6). IEEE.

Hayes-Huer, R. (2020). *Image Processing of Narrow Band Solar Eclipse Data Using Python and MaxIm DL*.

Marzal Varó, A., & Gracia Luengo, I. (2009). *Introducción a la programación con Python*. Universitat Jaume I. [Consulta: 02 de noviembre de 2022]. <http://repositori.uji.es/xmlui/bitstream/handle/10234/24305/s23.pdf?sequence=6&isAllowed=y>

Igual, R., & Medrano, C. (2008). *Tutorial de OpenCV. Laboratorio de visión por computadora*.

Jaskolka, K., Seiler, J., Beyer, F., & Kaup, A. (2019). *A Python-based laboratory course for image and video signal processing on embedded systems*. *Heliyon*, 5(10), e02560.

Jorge Alberto Marcial. (2011). *Detección de imágenes con contenido explícito usando los modelos de color HSV y YCbCr*. Instituto Politécnico Nacional. <https://www.repositoriodigital.ipn.mx/handle/123456789/12664> [Consulta: 22 de agosto de 2022].

Jothi, C. V., Gokulnath, S., Vetrivel, K., Balaji, H., & Issac, D. S. I. (2021). *Aircraft Visual Damage Inspection with Autonomous Drone and Image Processing*. *International Journal of Research in Engineering, Science and Management*, 4(4), 170-173.

Kostadinov, D., Pachovski, V., & Stojmenovska, I. (2020). *Home security system based on drone automation-IoT approach*. [Consulta: 18 de octubre de 2022].

Kumar, K. S., Sahoo, J. R., Panda, A., & Mishra, R. *A Python-based laboratory course for image and video signal processing on embedded systems*. [Consulta: 7 de enero de 2023].

La librería Numpy. (12 de mayo del 2022). *Published with Wowchemy Website Builder*. [Consulta: 23 de noviembre de 2022]. <https://aprendeconalf.es/docencia/python/manual/numpy/>

Luque López, B. (2016). *Aplicaciones de visión por computador en drones* (Bachelor's thesis, Universitat Politècnica de Catalunya). <https://upcommons.upc.edu/bitstream/handle/2117/89169/Belen%20Luque%20Memoria%20TFG.pdf?sequence=1&isAllowed=y>

Malacara Hernández, D. (1997). *Optica tradicional y moderna*. Secretaría de Educación Pública. [Consulta: 21 de abril de 2022].

Mandujano, S., Pazmany, M., & Rísquez-Valdepeña, A. (2017). *Drones: una nueva tecnología para el estudio y monitoreo de fauna y hábitats*. *Agroproductividad*. 2017; 10: 79–84., 10(10), 79-84.

Manual del usuario v1.0. (2018). TELLO. [https://dl-cdn.rzerobotics.com/downloads/Tello/201806mul/Tello%20User%20Manual%20V1.0\\_ES.pdf](https://dl-cdn.rzerobotics.com/downloads/Tello/201806mul/Tello%20User%20Manual%20V1.0_ES.pdf), consultada el 03 de Noviembre del 2021.

## Referencias

Martín, R. (2020). ¿Qué es OpenCV? Instalación en Python y ejemplos básicos. *Revista Digital INESEM*. <<https://revistadigital.inesem.es/informatica-y-tics/opencv/>> [Consulta: 15 de octubre de 2021].

Marzal Varó, A., & Gracia Luengo, I. (2009). *Introducción a la programación con Python*. Universitat Jaume I.

Medición de eficiencia de algoritmos de visión artificial implementados en raspberry pi y ordenador personal mediante Python», *Ingenium*, vol. 18. n.º 35, pp. 105-119, febrero. 2017.

Mery, D. (2004). *Visión por computador*. Santiago de Chile. Universidad Católica de Chile. <http://dmery.sitios.ing.uc.cl/Prints/Books/2004-ApuntesVision.pdf> [Consulta: 1 de agosto de 2022].

Morando, L. (2004). "La nueva era de conectividad inalámbrica". Canal AR. <<https://www.canal-ar.com.ar/Nota.asp?id=1479>> [Consulta: 15 de octubre de 2021].

Mordvintsev, A., & Abid, K. (2014). *Opencv-python tutorials documentation*. Obtenido de <https://storage.googleapis.com/kaggle-forum-message-attachments/904222/16192/OpenCV-Python%20Tutorials-2017.pdf?GoogleAccessId=web-data@kaggle-161607.iam.gserviceaccount.com&Expires=1665074722&Signature=i99M0uTynGbmRsNYffMgSSyrns0PCEm46BVF7SRkt68UZMS%2FuPpyV6etjWuWXZX1SA8o%2BMIWWe8LEunK5T7%2Bq8aN3bnZrLBeW9Jn9OsqJQhMFioLmNWUwqGaD7EMc0el%2BWqxwte2EZq2ucfQNYwSsYRH%2FT9slGIJVq8SoEe0wwwT%2FIko8QJBfzVjve7Si6LH5FJd2RQefx0jcHDqspGdBLxX1Vi%2FwB%2BC44b9RBB8HhLsOOaGrIWICOs2AB%2FIPqWCP4n0VTCOVkveYalKfEXqlpgDQAc1cia%2F%2BbApgku2j%2BH%2Fa%2B0bnYOE5Cuq6bUSG4Dckq2nNvXyHWpvK6Cp%2B3L2iQ%3D%3D>

Oliphant, T. E. (2006). *A guide to NumPy* (Vol. 1, p. 85). USA: Trelgol Publishing. Obtenido de: <https://ecs.wgtn.ac.nz/foswiki/pub/Support/ManualPagesAndDocumentation/numpybook.pdf>

Oltra, G. N., & Mellado, J. R. A. (2008). *Una introducción a la imagen digital y su tratamiento*. Univ de Castilla La Mancha. <https://books.google.es/books?hl=es&lr=&id=t4eEAEc9XSQC&oi=fnd&pg=PA9&dq=Imagen+digital+&ots=EPpFdFulWz&sig=zhx2sPOKfoIE9j3aBxLsMzCGdqo#v=onepage&q&f=false>

Ospina, J. C. M., Ortiz, F. A. P., & Bedoya, J. W. B. (2007). Corrección de iluminación para imágenes aéreas de cultivos tomadas a baja altitud. *Revista Facultad Nacional de Agronomía* Medellín, 60(2), 4077-4104. <https://revistas.unal.edu.co/index.php/refame/article/view/24460/25053> [Consulta: 28 de agosto de 2022].

Pérez González, J. M., Roquero Bravo, J. M., & Peraza Geist, T. D. (2019). *Inteligencia computacional aplicada a vigilancia con una flota de drones*.

Pino, E. (2019). Los drones una herramienta para una agricultura eficiente: un futuro de alta tecnología. *Idesia (Arica)*, 37(1), 75-84. [https://www.scielo.cl/scielo.php?pid=S0718-34292019000100075&script=sci\\_arttext&tlng=en](https://www.scielo.cl/scielo.php?pid=S0718-34292019000100075&script=sci_arttext&tlng=en)

## Referencias

Pinto, L. G. M., Mora-Camino, F., de Brito, P. L., Ramos, A. C. B., & Castro Filho, H. F. (2019). A ssd-ocr approach for real-time active car tracking on quadrotors. In *16th International Conference on Information Technology-New Generations (ITNG 2019)* (pp. 471-476). Springer, Cham.

Redacción APD. (2021). *Métodos y técnicas de inteligencia artificial: ¿cuáles son y para qué se usan?*. Zurbano, 90 28003 Madrid: APD. <<https://www.apd.es/tecnicas-de-la-inteligencia-artificial-cuales-son-y-para-que-se-utilizan/>> [Consulta: 10 de octubre de 2021].

Rouhiainen, L. (2018). *Inteligencia artificial*. Madrid: Alienta Editorial.

Sánchez Otálora, N. A. (2018). Definición y seguimiento de rutas tridimensionales para drones apoyados en realidad virtual. <https://repositorio.uniandes.edu.co/handle/1992/35057>, consultada el 13 de enero del 2022

Santos-Castro, A. (2017). A novel real-time image processing verification framework targeted to the zynq soc for drone localization applications. *Rochester Institute of Technology*.

SECRETARIA DE COMUNICACIONES Y TRANSPORTES. (2019). NORMA Oficial Mexicana NOM-107-SCT3-2019. <<https://www.sct.gob.mx/fileadmin/DireccionesGrales/DGAC-archivo/modulo2/nom-107-sct3-2019-201119.pdf>>

Shahmoradi, J., Talebi, E., Roghanchi, P., & Hassanalain, M. (2020). A comprehensive review of applications of drone technology in the mining industry. *Drones*, 4(3), 34.

Singh, S. (2021). Image filtration in Python using openCV. *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, 12(6), 5136-5143.

Sriratana, W., Mukma, S., Tammarugwattana, N., & Sirisantisamrid, K. (2018, July). Application of the OpenCV-Python for Personal Identifier Statement. In *2018 International Conference on Engineering, Applied Sciences, and Technology (ICEAST)* (pp. 1-4). IEEE.

Vergouw, B., Nagel, H., Bondt, G., & Custers, B. (2016). Drone technology: Types, payloads, applications, frequency spectrum issues and future developments. In *The future of drone use* (pp. 21-45). TMC Asser Press, The Hague.

Yoza, M. M. B., Zuñiga, K. M., Zambrano, S. M. Z., & Pionce, W. R. C. (2021). EL USO DE LOS DRONES EN EL ÁMBITO PROFESIONAL. *UNESUM-Ciencias. Revista Científica Multidisciplinaria*. ISSN 2602-8166, 5(4), 75-82. <https://revistas.unesum.edu.ec/index.php/unesumciencias/article/view/516>

Zúñiga-López, A., & Avilés-Cruz, C. (2020). Digital signal processing course on Jupyter-Python Notebook for electronics undergraduates. *Computer Applications in Engineering Education*, 28(5), 1045-1057.

OpenCV-Python Tutorials — OpenCV-Python Tutorials beta documentation. (s/f). Readthedocs.io. Recuperado el 15 de mayo de 2023, de [https://opencv24-python-tutorials.readthedocs.io/en/latest/py\\_tutorials/py\\_tutorials.html](https://opencv24-python-tutorials.readthedocs.io/en/latest/py_tutorials/py_tutorials.html)

# ANEXOS

Se adjunta hasta el termino de este trabajo, diploma de la conferencia que se realizó del artículo acerca de este trabajo de tesis, así como la imagen de la portada del artículo, tal y como se ve en la Figura 51.

Ingenio del Quehacer Mecatrónico, Capítulo 23, pp. 289 – 301  
ISBN: 978-607-9394-27-, 2023



## Propuesta de metodología para la programación de un vehículo aéreo no tripulado

Ramos Juárez Enrique Iván, Juan Manuel Ramos Arreguín<sup>✉</sup>, Pedraza Ortega Jesús Carlos, Aceves Fernández Marco Antonio, Avendaño Juárez José Luis, Gorrostieta Hurtado Efrén

Universidad Autónoma de Querétaro, Facultad de Ingeniería

<sup>✉</sup>[juan.ramos@uaq.edu.mx](mailto:juan.ramos@uaq.edu.mx)

### Resumen

*El uso de los drones se ha extendido bastante hoy en día. Desde aplicaciones de entretenimiento, vigilancia, reconocimiento remoto, y muchas otras aplicaciones. En este trabajo se presenta una propuesta de metodología para implementar un algoritmo de seguimiento de ruta para un drone marca Tello. Para esto, se usa el lenguaje Python y la librería djitellopy, la cual permite establecer una comunicación por WiFi con el drone. De esta manera, se puede obtener imágenes de la cámara del drone, y procesarla en la computadora para identificar la ruta a seguir. La librería permite el envío de comandos, para que el drone realice diversos movimientos como subir, bajar, girar a la derecha, izquierda, suspenderse en el aire y aterrizar, entre otros movimientos. Se logró el reconocimiento de la ruta, a diferentes velocidades. Este trabajo marca la pauta para que otros estudiantes puedan continuar con este trabajo y así lograr mejorar los resultados.*

**Palabras clave:** Drone, Python, procesamiento de imágenes, reconocimiento, ruta.

### 1. Introducción

El uso de drones está cobrando cada vez una mayor importancia en el mundo actual. Los podemos usar con fines recreativos, de vigilancia, como mensajería, y una gran variedad de aplicaciones. Por todos esto, la tecnología de los drones se considera como parte del desarrollo de tecnología actual, especialmente a partir del año 2017 [1]. Siendo uno de los aspectos que se deben trabajar en ese tipo de sistemas, es la autonomía de vuelo. En este sentido, se han desarrollado diversos trabajos enfocados en el control de vuelo de los drones, siendo algunos de estos trabajos los siguientes.

En 2018, Anton presenta un trabajo donde un drone sigue el sendero de un bosque, y aplica aprendizaje profundo o redes neuronales convolucionales [2]. También los drones pueden ser usados en mensajería, para la entrega de mercancías. Miranda presenta un sistema de navegación que permite la entrega de mercancía con drones autónomos, generando un punto inicial y un punto final, con la ayuda de un sistema GPS, barómetro y otros [3]. El aterrizaje se basa en la detección de un marcador usando una cámara, y el método se basa en campos vectoriales para seguir la ruta [3].

En el 2016, Vergouw [4] menciona la diferencia entre drones de acuerdo a la aplicación, incluyendo el uso de sensores y de comunicación inalámbrica. Así mismo, realizan un análisis de los aspectos legales respecto al espectro de frecuencias y equipos electrónicos, así como el alcance y autonomía de los drones.

Los drones se presentan como una de las 11 tecnologías de mayor relevancia en la década, debido a su crecimiento en aplicaciones de entretenimiento, industrial y en muchas ocasiones, de rescate [5].