

UNIVERSIDAD DE EL SALVADOR  
FACULTAD DE INGENIERÍA Y ARQUITECTURA  
ESCUELA DE INGENIERÍA ELÉCTRICA



**SISTEMA EMBEBIDO DE SEGUIMIENTO Y  
RECONOCIMIENTO FACIAL APLICADO PARA LA  
SEGURIDAD Y ACCESO RESTRINGIDO**

PRESENTADO POR:

**ALVIN ISAAC MERINO BATRES**

PARA OPTAR AL TÍTULO DE:

**INGENIERO ELECTRICISTA**

CIUDAD UNIVERSITARIA, ENERO 2019

**UNIVERSIDAD DE EL SALVADOR**

**RECTOR:**

**M. Sc. ROGER ARMANDO ARIAS ALVARADO**

**SECRETARIO GENERAL:**

**M. Sc. CRISTÓBAL HERNÁN RÍOS BENÍTEZ**

**FACULTAD DE INGENIERÍA Y ARQUITECTURA**

**DECANO:**

**M. Sc. e ING. FRANCISCO ANTONIO ALARCÓN SANDOVAL**

**SECRETARIO:**

**ING. JULIO ALBERTO PORTILLO**

**ESCUELA DE INGENIERÍA ELÉCTRICA**

**DIRECTOR:**

**ING. ARMANDO MARTÍNEZ CALDERÓN**

UNIVERSIDAD DE EL SALVADOR  
FACULTAD DE INGENIERÍA Y ARQUITECTURA  
ESCUELA DE INGENIERÍA ELÉCTRICA

Trabajo de Graduación previo a la opción al Grado de:

**INGENIERO ELECTRICISTA**

Título:

**SISTEMA EMBEBIDO DE SEGUIMIENTO Y  
RECONOCIMIENTO FACIAL APLICADO PARA LA  
SEGURIDAD Y ACCESO RESTRINGIDO**

Presentado por:

**ALVIN ISAAC MERINO BATRES**

Trabajo de Graduación Aprobado por:

Docente Asesor:

**M. Sc. e ING. JOSÉ WILBER CALDERÓN URRUTIA**

SAN SALVADOR, ENERO 2019

Trabajo de Graduación Aprobado por:

Docente Asesor:

**M. Sc. e ING. JOSÉ WILBER CALDERÓN URRUTIA**

## ACTA DE CONSTANCIA DE NOTA Y DEFENSA FINAL

En esta fecha, martes 11 diciembre de 2018, en el aula de postgrado de la Escuela de Ingeniería Eléctrica, a las 10:00 a.m. horas, en presencia de las siguientes autoridades de la Escuela de Ingeniería Eléctrica de la Universidad de El Salvador:

1. Ing. Armando Martínez Calderón  
Director

2. MSc. José Wilber Calderón Urrutia  
Secretario

  
Firma

  
Firma

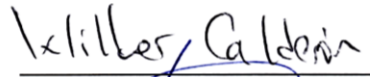



Y, con el Honorable Jurado de Evaluación integrado por las personas siguientes:

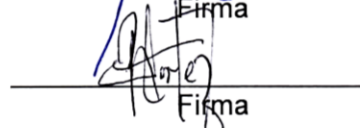
- MSC. JOSE WILBER CALDERON URRUTIA  
(Docente Asesor)

- MSC. SALVADOR DE JESUS GERMAN

- MSC. RICARDO ERNESTO CORTEZ

  
Firma

  
Firma

  
Firma

Se efectuó la defensa final reglamentaria del Trabajo de Graduación:

SISTEMA EMBEBIDO DE SEGUIMIENTO Y RECONOCIMIENTO FACIAL APLICADO  
PARA LA SEGURIDAD Y ACCESO RESTRINGIDO

A cargo del Bachiller:

- MERINO BATRES ALVIN ISAAC

Habiendo obtenido en el presente Trabajo una nota promedio de la defensa final:

8.7

( OCHO . SIETE )

## AGRADECIMIENTOS

Me gustaría comenzar agradeciendo a mi madre Marta Lidia Batres por tantos años de apoyo y cariño brindados incondicionalmente, su esfuerzo y sacrificio me han permitido llegar hasta este punto de mi vida y a ella dedico este logro alcanzado.

A mi fallecido padre José Armando Merino Méndez por mostrarme la visión científica de la vida, por tantos problemas matemáticos que resolvimos juntos y por compartir conmigo sus experiencias de la vida.

A mis hermanos, José Armado Merino y Kevin Alexis Merino por ayudarme a conseguir las herramientas necesarias para desarrollar este trabajo de graduación. Agradezco su apoyo económico y emocional, además, de colaborar con mis actividades académicas.

A mis amigos y compañeros por tantas horas de estudio, preparación para los exámenes que se venían y por tantos momentos divertidos que vivimos durante este tiempo.

Al Ing. Jorge Antonio Martínez, por brindarme una orientación profesional y compartir conmigo su experiencia laboral como ingeniero electricista.

A mi docente tutor, M. Sc. e ING. Wilber Calderón (Sam) por su orientación académica y por brindarme la oportunidad de desarrollar este trabajo de graduación.

Agradezco también a la administración general del complejo deportivo de la Universidad de El Salvador por permitirme realizar las pruebas necesarias de vuelo en la cancha fútbol.

Me encuentro muy agradecido con ustedes y les deseo éxito en la vida.

## ÍNDICE GENERAL

INTRODUCCIÓN.....	xi
<b>CAPÍTULO 1. ENUNCIADO DEL PROBLEMA.....</b>	<b>1</b>
1.1. Descripción del tema.....	2
1.2. Objetivos. ....	3
1.3. Alcances. ....	3
1.4. Antecedentes.....	4
1.5. Planteamiento del problema.....	5
1.6. Justificación.....	5
<b>CAPÍTULO 2. MARCO TEÓRICO. ....</b>	<b>6</b>
2.1. Principios de vuelo de dron quadrotor.....	7
2.2. Sensores. ....	9
2.2.1. Unidad de medición inercial (IMU). ....	9
2.2.2. Sistema de posicionamiento global (GPS).....	10
2.2.3. Cámara PI de 5 megapíxeles, 1/4" CCD.....	12
2.2.4. Barómetro.....	12
2.2.5. Sistema de propulsión.....	13
2.2.5.1. Motores sin escobillas.....	13
2.2.5.2. Controlador electrónico de velocidad.....	14
2.3. Raspberry Pi 3 Modelo B.....	15
2.4. NAVIO Versión 2.....	16
2.5. Ardupilot. ....	17
2.5.1. Modo guiado. ....	17
2.5.2. Implementación PID.....	18
2.5.3. Filtro de Kalman Extendido (EKF).....	19
2.6. DroneKit-Python.....	19
2.7. Biblioteca OpenCV.....	19
<b>CAPÍTULO 3. DISEÑO DEL PROYECTO ....</b>	<b>21</b>
3.1. Configuración física del dron quadrotor.....	22
3.1.1. Sistema de potencia. ....	23

3.1.2. Bateria tipo LiPo. ....	24
3.1.3. Sistema de propulsión DJI E305. ....	25
3.1.4. Sistema GNSS. ....	26
3.2. Calibración de sensores. ....	26
3.3. Sistema de adquisición de imágenes para la base de datos. ....	27
3.4. Sistema de reconocimiento y seguimiento facial. ....	27
<b>CAPÍTULO 4. RESULTADOS</b> .....	<b>29</b>
4.1. Verificación de la calidad de la señal GPS. ....	30
4.2. Comparación métodos de reconocimiento facial. ....	31
4.3. Características dron quadrotor. ....	31
<b>CONCLUSIONES</b> . ....	<b>33</b>
<b>RECOMENCACIONES</b> . ....	<b>34</b>
<b>BIBLIOGRAFÍA</b> . ....	<b>36</b>
<b>ANEXOS</b> . ....	<b>38</b>
ANEXO A. Configuración Raspberry Pi 3 modelo B. ....	38
ANEXO B. Principios de operación de Acelerómetro lineal digital 3D, Giroscopio lineal digital 3D y Magnetómetro lineal digital 3D. ....	45
ANEXO C. Manual de usuario. ....	46



## ÍNDICE FIGURAS

Figura 1. Dron Quadrotor.....	7
Figura 2. Configuración en “X” .....	8
Figura 3. Diagrama de cuerpo libre de un dron quadrotor.....	9
Figura 4. IMU de 9 grados de libertad.....	10
Figura 5. Volumen servicio terrestre.....	11
Figura 6. Ubicación de la red de instalaciones de monitoreo.....	11
Figura 7. Cámara Pi V1 modulo “H” .....	12
Figura 8. Barómetro digital MS5611. ....	13
Figura 9. Motor sin escobillas 2312E.....	14
Figura 10. Controlador electrónico de velocidad 420 lite. ....	14
Figura 11. Circuito general controlador electrónico de velocidad. ....	15
Figura 12. Raspberry Pi 3 Modelo B.....	16
Figura 13. Piloto automático Navio2. ....	16
Figura 14. Diagrama de bloques control PID para estabilidad de rotación de cada eje. ....	18
Figura 15. Algunas Características Haar para la detección de rostros. ....	20
Figura 16. Esquema de configuración dron Quadrotor. ....	22
Figura 17. Montaje antivibraciones Raspberry Pi. ....	22
Figura 18. Módulo de potencia para Navio2.....	23
Figura 19. Regulador DC UBEC de 3A.....	24
Figura 20. Batería tipo LiPo.....	24
Figura 21. Montaje sistema de propulsión E305. ....	25
Figura 22. Conexión controladores electrónicos de velocidad con placa Navio2....	25
Figura 23. Montaje antena GPS.....	26
Figura 24. Captura ejecución código SEGUIRECOFACIAL.py.....	28
Figura 25. Captura de pantalla al ejecutar u-center en PC. ....	30
Figura 26. Comparación métodos de reconocimiento facial. ....	31
Figura 27. Dron quadrotor construido. ....	32

## LISTA DE SIGLAS Y ABREVIATURAS

CSI	Camera Serial Interface (Interfaz serie de cámara)
CCD	Charge Coupled Device (Dispositivo de carga acoplada)
DSI	Display Serial Interface (Interfaz serie de pantalla)
FLD	Fisher's Linear Discriminant (Discriminante Lineal de Fisher)
GNSS	Global Navigation Satellite System (Sistema global de navegación por satélite)
GPIO	General Purpose Input/Output (Entrada/salida de propósito general)
GPS	Global Positioning System (Sistema de posicionamiento global)
HDMI	High Definition Multimedia Interface (Interfaz multimedia de alta definición)
IMU	Inertial Measurement Unit (Unidad de medición inercial)
LAN	Local Area Network (Red de área local)
LBPH	Local Binary Patterns Histograms (Histogramas de patrones locales binaries)
LiPo	Lithium Polymer (Polímero de iones de litio)
MCX	Micro Coaxial
MOSFET	Metal Oxide Semiconductor Field Effect Transistor (Transistor de efecto de campo metal oxido semiconductor)
OpenCV	Open Source Computer Vision (Visión de computadora de código abierto)
PCA	Principal Component Analysis (Análisis de componentes principales)
PID	Controlador Proporcional, Integral y Derivativo
PPM	Pulse Position Modulation (Modulación de posición de puntos)
PWM	Pulse Width Modulation (Modulación por ancho de pulsos)
RAM	Random Access Memory (Memoria de acceso aleatorio)
SSH	Secure SHell
UART	Universal Asynchronous Receiver Transmitter (Transmisor receptor asíncrono universal)

## INTRODUCCIÓN.

En la actualidad existen instituciones que requieren un desempeño más eficiente de sus sistemas de seguridad, para el resguardo de equipos de gran valor y del personal. Esto muchas veces implica que, en los controles de acceso restringido, es necesario un seguimiento y reconocimiento más minucioso de cada persona que ingresa. Una manera de lograr este objetivo es contratando más personal de seguridad o adquiriendo herramientas más sofisticadas.

En los sistemas de video vigilancia modernos, se pueden encontrar cámaras de alta definición capaces de hacer un paneo, una inclinación o enfocar un objeto o persona por medio de un control manual, pero aun con estas características las zonas de visión son limitadas y es necesario la instalación de más cámaras, lo cual, eleva los costos de instalación y de mantenimiento.

Este trabajo de graduación ofrece un dispositivo aéreo versátil con una cámara abordo capaz de realizar seguimiento y reconocimiento facial de personas.

Un sistema embebido es un sistema integrado que combina componentes mecánicos y eléctricos junto con una computadora, oculta en su interior, para realizar un único propósito. Para su desarrollo se implementan conceptos relacionados con: procesamiento digital de imágenes, unidades de medición inercial, motores sin escobillas, controladores electrónicos de velocidad, teoría de controladores PID, computadoras de placa reducida, comunicación inalámbrica, sistema de posicionamiento global y principios de vuelo de drones quadrotor.

Para el reconocimiento facial se comparan tres métodos, los cuales son: Fisherfaces, el cual utiliza el discriminante lineal de Fisher (FLD); Eigenfaces, el cual utiliza el análisis de componentes principales (PCA) para realizar una proyección lineal que maximiza la variación del espacio de imágenes a un espacio de características de menor dimensión y LBPH, este método resume la estructura local en una imagen comparando cada pixel con su vecindario. Estos métodos se encuentran en la biblioteca de OpenCV. En este trabajo de graduación son implementados en el

lenguaje de programación conocido como Python. Para entrenar estos métodos se utiliza una base de datos con imágenes de rostros generadas con un programa de captura y almacenamiento de rostros. Estos rostros son almacenados en carpetas, cuyos nombres corresponden a los nombres de las personas a las que pertenecen.

---

**CAPÍTULO 1. ENUNCIADO DEL PROBLEMA.**

---

## **1.1. Descripción del tema.**

En el presente trabajo de graduación se desarrollará un sistema embebido de reconocimiento facial que realizará un tratamiento digital de imágenes para identificar a una persona mediante la localización del rostro y un posterior análisis de las características faciales extraídas de una fotografía o de una fuente de video en tiempo real, el cual utilizará un buffer de datos proveniente de una cámara hacia una unidad de placa reducida para luego asociar el rostro con una persona conocida si es que se encuentra en la base de datos almacenada. Se evaluarán tres métodos de reconocimiento facial y considerando las características del hardware a utilizar, se implementará el que mejor se ajuste a los objetivos de este trabajo de graduación.

El dispositivo se desplazará utilizando un dron quadrotor que se explica en el capítulo 2, el cual tendrá un sistema automático de vuelo y permitirá control remoto, utilizando los estándares IEEE 802.15.4f y IEEE 802.22 para su comunicación y geolocalización respectivamente. En espacios cerrados podrá evitar choques y tendrá a bordo una batería como fuente de energía para realizar vuelos rutinarios.

## **1.2. Objetivos.**

### GENERAL:

- Desarrollar un dispositivo capaz de realizar reconocimiento facial y seguimiento aéreo utilizando software y hardware de libre distribución.

### ESPECIFICOS:

- Implementar los algoritmos necesarios para el reconocimiento facial.
- Realizar el tratamiento digital de los rostros capturados.
- Lograr que el dispositivo sea capaz de desplazarse horizontal y verticalmente de forma estable utilizando lazos cerrados de retroalimentación.
- Aplicar sistemas de control automático a los movimientos del dispositivo con la información obtenida de la cámara para realizar seguimiento.

## **1.3. Alcances.**

- El dispositivo tendrá un tiempo de vuelo de 10 minutos.
- La cámara tendrá una resolución de 5 Megapíxeles.
- El vuelo del dispositivo será estable.
- El dispositivo permitirá control remoto.
- Al tener un nivel bajo de carga en batería regresara al punto de origen.
- Capaz de realizar misiones programadas con GPS.

#### 1.4. Antecedentes.

En la década de los sesenta, Woodrow W. Bledsoe, Helen Chan y Charles Bisson<sup>1</sup> investigaron la programación de computadoras para realizar reconocimiento facial de personas. Su trabajo consistió en utilizar una gran base de datos de imágenes y compararlas con una fotografía el problema era seleccionar de la base de datos un pequeño conjunto de registros de manera que uno de los registros de la imagen coincidiera con la fotografía. El éxito del método podía medirse en términos de la relación de la lista de respuestas con el número de registros en la base de datos. Bledsoe describió las siguientes dificultades:

“Este problema de reconocimiento se dificulta debido a la gran variabilidad en la rotación e inclinación de la cabeza, la intensidad y el ángulo de la luz, la expresión facial, el envejecimiento, etc. Algunos otros intentos de reconocimiento facial por máquina han permitido poca o ninguna variabilidad en estas cantidades. Sin embargo, el método de correlación o de coincidencia de patrones de datos ópticos no procesados, que a menudo utilizan algunos investigadores, seguramente fallará en los casos en que la variabilidad sea grande. En particular, la correlación es muy baja entre dos imágenes de la misma persona con dos rotaciones de cabeza diferentes. [1]”

Este proyecto fue etiquetado como hombre-máquina pues requería que el humano localizara las características como ojos, orejas, nariz, y boca en la fotografía, y luego eran utilizadas por la computadora para su reconocimiento.

A finales de la década de los ochenta, se produjo un nuevo punto de referencia cuando Kirby y Sirovich aplicaron una técnica estándar del álgebra lineal: el análisis de componentes principales (PCA); que, hasta el día de hoy, es una de las técnicas que más se utiliza en reconocimiento facial.

---

<sup>1</sup> Jeremy Norman & Co., Inc., «Woodrow Bledsoe Originates of Automated Facial Recognition,» [En línea]. Available: <http://www.historyofinformation.com/expanded.php?id=2495>. [Último acceso: 23 septiembre 2018].



### **1.5. Planteamiento del problema.**

Los sistemas de video vigilancia actuales presentan restricciones en movilidad y procesamiento de imágenes, lo cual no permite identificar personas. Para lograr este objetivo es necesario elevar la inversión realizada con la adquisición de más cámaras y un software especializado. Por esta razón, se requiere un dispositivo con la capacidad de desplazarse y que permita realizar reconocimiento de personas para mejorar la seguridad de las instalaciones, con un costo menor a los sistemas de video vigilancia convencionales.

### **1.6. Justificación.**

- En este trabajo de graduación se busca aplicar las nuevas tecnologías para mejorar la seguridad.
- Profundizar en el tema de reconocimiento facial, pues es un campo que no se ha explorado en el país.
- Reducir los costos que implica el mantenimiento e instalación de un sistema de video vigilancia.
- Brindar una herramienta que sirva como base para el desarrollo de otras aplicaciones.
- Con el desarrollo de un dispositivo de reconocimiento facial se agiliza los procesos de identificación de personas.

---

## **CAPÍTULO 2. MARCO TEÓRICO.**

---

## 2.1. Principios de vuelo de dron quadrotor.

La Figura 1 muestra un dron quadrotor el cual, es un helicóptero con cuatro rotores. Sus hélices “producen un flujo de aire y, a su vez, una fuerza de aceleración en el cuerpo del dron quadrotor [2]” opuesta al de la gravedad. Esta fuerza puede controlarse variando la velocidad angular de cada uno de sus cuatro rotores.



Figura 1. Dron Quadrotor.

Fuente: Parrot SA (2018). Parrot Bebop 2 Power - Pack FPV. [Figura]. Recuperado de <https://www.parrot.com/soluciones-business/drones/parrot-bebop-2-power-pack-fpv>

Un dron quadrotor tiene dos opciones de distribución de rotores, en “X” o en “+” y por lo general se suele utilizar la distribución en “X” cuando se tiene una cámara abordo. Un dron quadrotor posee ciertas ventajas sobre otros dispositivos de vuelo, tales como: una estructura más simple que un helicóptero, esto requiere menos mantenimiento; mayor capacidad de carga que un dron con tres rotores, necesaria para incorporar sensores y al ser controlado permite realizar mejores maniobras de vuelo que un dron hexarotor.

Para evitar una rotación permanente en el eje vertical, el dron quadrotor posee dos hélices que giran en sentido horario (CW) y dos que giran en sentido antihorario (CCW). Tal como se muestra en la Figura 2 cada hélice se encuentra ubicada en el brazo opuesto al que gira en el mismo sentido y de esta manera conserva el momento angular de la aeronave.



Figura 2. Configuración en "X".

Fuente: ArduPilot Dev Team (2018). Motor order diagrams. [Figura]. Recuperado de <http://ardupilot.org/copter/docs/connect-escs-and-motors.html>

En dinámica de vuelo de aeronaves se le conoce como Guiñada (Yaw) a la rotación en el eje vertical, alabeo (Roll) a la rotación en el eje frente-cola y cabeceo (Pitch) a la rotación en el eje izquierda-derecha.

Si se desea una rotación en sentido horario sobre el eje de guiñada basta con aumentar la velocidad angular de las hélices que giran en sentido horario y disminuir la velocidad de las hélices que giran en sentido antihorario. Para una rotación en sentido antihorario, se debe disminuir la velocidad angular de las hélices que giran en sentido horario y aumentar la velocidad de las hélices que giran en sentido antihorario. Por otra parte, si se quiere desplazar el dron quadrotor hacia la derecha, se debe aumentar la velocidad angular de los rotores a la izquierda y disminuir la velocidad de los rotores a la derecha. En caso contrario, si se desea un desplazamiento a la izquierda, se aumentará la velocidad angular de los rotores a la derecha y se disminuirá la velocidad angular de los rotores a la izquierda. Esto sí, la configuración de los rotores es en "X".

Para desplazar el dron quadrotor hacia adelante, se aumenta la velocidad angular de los rotores en la parte trasera y se disminuye la velocidad angular de los rotores que se encuentran en frente de la aeronave.

Para un desplazamiento hacia atrás, se aumenta la velocidad angular de los rotores al frente y se disminuye la velocidad angular de los rotores en la parte trasera. Es

posible combinar estos movimientos básicos, pero si desea mantener la altitud constante, es necesario recordar que la suma total de los empujes de cada rotor sea igual a la contribución de la fuerza de gravedad aplicada al dron quadrotor tal y como se muestra en la Figura 3.

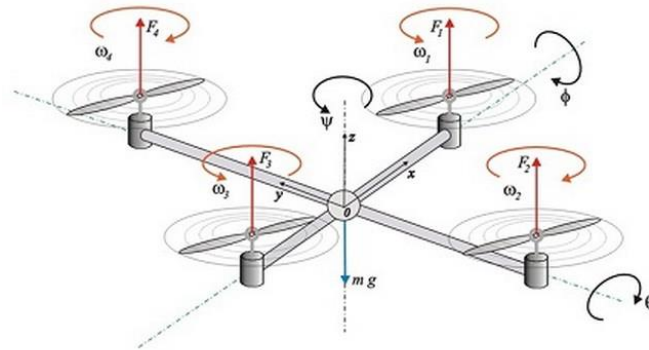


Figura 3. Diagrama de cuerpo libre de un dron quadrotor.

Fuente: Ziff Davis, LLC. PCMag Digital Group (2018). Drone athleticism: astounding new quadrotor control, and beyond. [Figura]. Recuperado de <http://www.extremetech.com/extreme/158510-drone-athleticism-astounding-new-quadrotor-control-and-beyond>

## 2.2. Sensores.

Los sensores son una parte muy importante del dron quadrotor, permiten orientar y estabilizar su vuelo, al realizar mediciones de magnitudes físicas en su entorno y luego convirtiéndolas en variables eléctricas para corregir su posición o inclinación. Básicamente un dron quadrotor puede estar constituido por los siguientes sensores que se describen en las siguientes secciones:

### 2.2.1. Unidad de medición inercial (IMU).

Las unidades de medición inercial son unidades electrónicas de procesamiento de movimiento, como se puede observar en la Figura 4 están formadas por un acelerómetro lineal digital 3D, un giroscopio lineal digital 3D y con un magnetómetro lineal digital 3D, los cuales son capaces de medir la aceleración, velocidad angular y la intensidad del campo magnético respectivamente. Estas

magnitudes físicas se miden con sensores en cada uno de los ejes y determinan su grado de libertad.

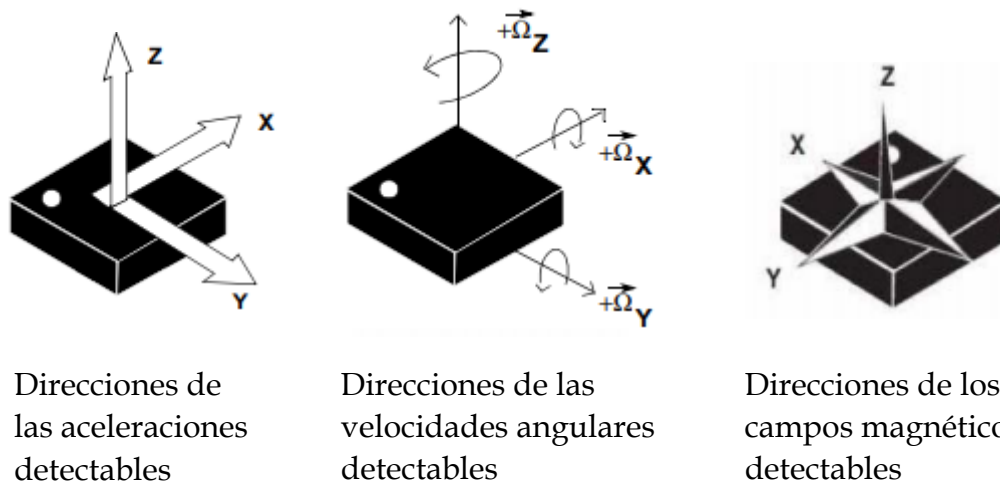


Figura 4. IMU de 9 grados de libertad.

Fuente: SparkFun Electronics (2018). LSM9DS1 Overview. [Figura]. Recuperado de <https://learn.sparkfun.com/tutorials/lsm9ds1-breakout-hookup-guide/all>

### 2.2.2. Sistema de posicionamiento global (GPS).

El receptor GPS es otro elemento muy importante que forma parte de un dron quadrotor, permite ubicar su posición en toda la tierra captando una señal de radio proveniente de satélites en órbita. El GPS es un sistema global de navegación por satélite (GNSS) propiedad de los Estados Unidos de América, su fuerza aérea se encarga de desarrollarlo, operarlo y darle mantenimiento [3].

El GPS está formado por 3 segmentos los cuales son:

1. El segmento espacial, está formado por al menos 24 satélites en órbita terrestre media a una altura aproximada de 20,200km, con una disponibilidad de al menos el 95% del tiempo, transmiten datos de navegación a 50 bits/s. Cada satélite posee un sistema de banda L definida por el estándar IEEE 521-2013, que modula las secuencias binarias en las portadoras de banda L1 (1575.42MHz) y L2 (1227.6MHz) para luego ser transmitidas por la antena de matriz helicoidal [4]. La cobertura de servicio terrestre por satélite, es una región que se extiende

3000km sobre el nivel de la Tierra tal como se muestra en la Figura 5 y es visible desde la porción orbital del satélite. Luego de aplicar un ángulo de mascara para mejorar su rendimiento, esta región representa aproximadamente un tercio de la superficie de la tierra.

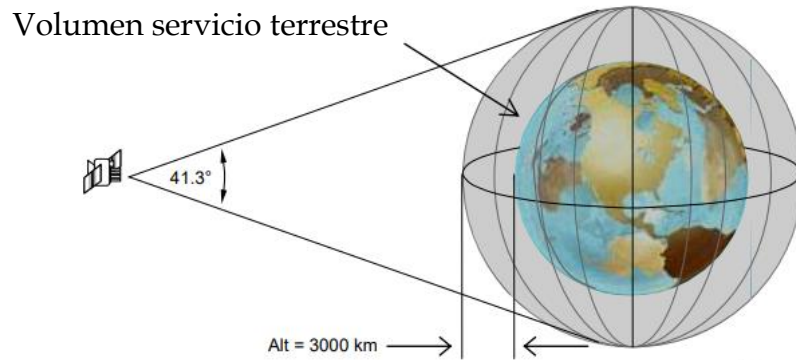


Figura 5. Volumen servicio terrestre.

Fuente: U. D. o. Defense (2018). GLOBAL POSITIONING SYSTEM STANDARD POSITIONING SERVICE PERFORMANCE STANDARD, cuarta edición. [Figura]. Recuperado de <https://www.gps.gov/technical/ps/2008-SPS-performance-standard.pdf>.

2. El segmento de control, está formado por una red global de instalaciones terrestres de monitoreo mostradas en la Figura 6, se encargan de rastrear y enviar comandos a la red de satélites del sistema.

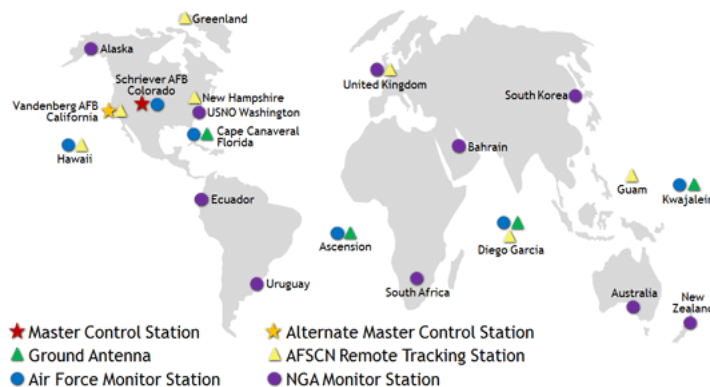


Figura 6. Ubicación de la red de instalaciones de monitoreo.

Fuente: NOAA (2018). Control Segment [Figura]. Recuperado de <https://www.gps.gov/systems/gps/control/>

3. El segmento de usuario, es el equipo receptor en sí, el cual, utiliza la información recibida de los satélites GPS para determinar su posición 3D y el tiempo.

### **2.2.3. Cámara Pi de 5 megapíxeles, 1/4" CCD.**

La Cámara Pi es un sensor que puede ser incorporado en las distintas versiones de la placa reducida Raspberry Pi mediante un cable plano, permitiendo la adquisición de imágenes al leer los píxeles del sensor fila por fila, similar a las cámaras de los teléfonos celulares.

Existen las versiones V1 y V2 en sus distintas derivaciones, como la cámara Pi modulo "H" NoIR mostrada en la Figura 7, la cual permite visión nocturna e incorpora 2 led infrarrojos. La versión V1 tiene el sensor omniVision OV5647 de 5 megapíxeles, 1/4" CCD, su campo de visión es de 160 grados y su voltaje de alimentación es de 3.3V.



Figura 7. Cámara Pi V1 modulo "H".

Fuente: BOTN ROLL (2018). Camara p/ Raspberry Pi c/ lente olho de peixe e suporte p/ visão noturna [Figura]. Recuperado de <https://www.botnroll.com/pt/camaras/1468--camara-p-raspberry-pi-c-lente-olho-de-peixe-e-suporte-p-visao-noturna-.html>

### **2.2.4. Barómetro.**

Un barómetro es un sensor que permite conocer la altitud a la que se encuentra a partir de la presión atmosférica. Actualmente existen barómetros digitales de gran precisión, estos son muy útiles cuando se desea medir la altura a la que se encuentra un dron quadrotor cuando no se tiene una buena recepción GPS.



Un ejemplo de este tipo de sensores es el MS0611 mostrado en la Figura 8, el cual, está incluido en la placa Navio2 y con una precisión de hasta 10cm.

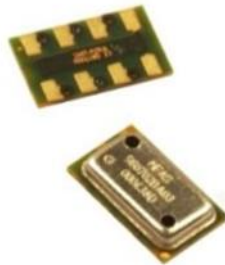


Figura 8. Barómetro digital MS5611.

Fuente: ALIBABA (2018). MS5611-01BA01 Variometer módulo, con tapa de LCP Alta Resolución altímetro sensores [Figura]. Recuperado de <https://spanish.alibaba.com/product-detail/ms5611-01ba01-variometer-module-with-lcp-cap-high-resolution-altimeter-sensors-917465602.html?spm=a2700.8699010.normalList.4.276f2cc9IXynIX>

### **2.2.5. Sistema de propulsión.**

Los motores y controladores electrónicos de velocidad son parte del sistema de propulsión de un dron quadrotor, el cual, permite su desplazamiento mediante pulsos PWM procedentes del dispositivo de control de vuelo y su comportamiento dependerá de las órdenes recibidas desde la estación de control terrestre y calibración de sus controladores PID.

#### **2.2.5.1. Motores sin escobillas.**

Los motores generan el empuje necesario para vencer la fuerza de gravedad que actúa sobre el dron quadrotor. Existe un tipo de motor el cual es conocido como motor sin escobillas, este tipo de motor tiene electroimanes fijos en el exterior con al menos tres bobinas conectadas a tres polos e imanes permanentes en el eje. No necesita mantenimiento y son de alta eficiencia en comparación a los motores de corriente continua con escobillas. Un ejemplo de este tipo de motor para drones es el modelo 2312E fabricado por la empresa china SZ DJI Technology Co., Ltd (DJI) mostrado en la Figura 9.



Figura 9. Motor sin escobillas 2312E.

Fuente: RCDRIVE (2018). DJI E305 2312E Brushless Motor CCW 960kV [Figura]. Recuperado de <http://rcdrive.ru/unit.php?unit=27555&pPage=7>

### 2.2.5.2. Controlador electrónico de velocidad

Los controladores electrónicos de velocidad permiten variar la potencia de un motor utilizando los ciclos de trabajo de una señal digital PWM. Por ejemplo, si el ciclo de trabajo es cero no alimenta los motores, pero, si el ciclo de trabajo es todo el periodo entonces los motores trabajarán a máxima velocidad.

Estos controladores generan una señal de corriente alterna desplazada  $120^\circ$ , induciendo un campo magnético rotativo en los motores sin escobillas a la frecuencia de estas tres señales. Un ejemplo de estos controladores de velocidad es el modelo 420 lite mostrado en la Figura 10, el cual, es fabricado por la empresa SZ DJI Technology Co., Ltd (DJI).



Figura 10. Controlador electrónico de velocidad 420 lite.

Fuente: OÜ OSTAX (2018). 420LITE ESC [Figura]. Recuperado de <http://www.droonikeskus.ee/toode/420lite-esc/>

Este controlador no puede alimentar directamente a los motores y para poder trabajar con altas corrientes utiliza un convertidor de corriente directa a alterna el cual está formado por un condensador y un chip MOSFET tipo N. en la figura 11 se muestra un ejemplo de circuito eléctrico general de los controladores electrónicos de velocidad.

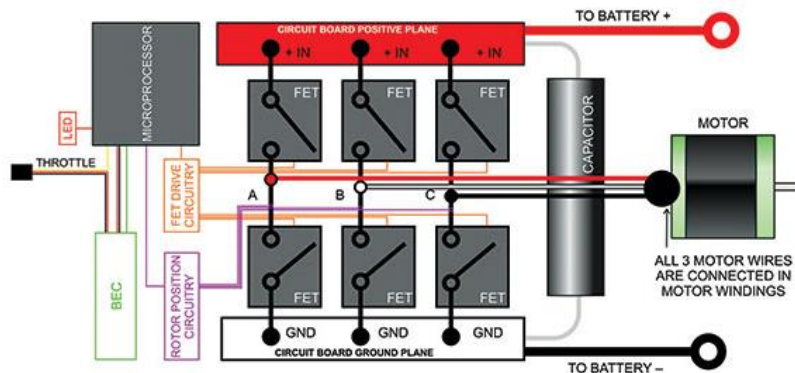


Figura 11. Circuito general controlador electrónico de velocidad.

Fuente: FPV RADIO CONTROL (2018). Como funcionan los controladores Brushless (ESC) [Figura]. Recuperado de <https://www.fpvradiocontrol.com/como-funcionan-los-controladores-brushless-esc/>

### 2.3. Raspberry Pi 3 Modelo B

La Raspberry Pi 3 Modelo B mostrada en la Figura 12, es una computadora de placa reducida versátil, tiene todos los elementos básicos de una computadora de escritorio o una laptop. Una unidad de procesamiento central Broadcom BCM2837 de 64bit con un clúster ARM Cortex A53 de cuatro núcleos, los cuales se ejecutan a 1.2GHz, una memoria RAM de 1GB, permite conectarse a una LAN inalámbrica 802.11 b/g/n, tiene Bluetooth versión 4.1, 4 puertos USB 2.0, un puerto Ethernet, 40 pines GPIO, salida de 4 polos estéreo y puerto de video compuesto, 1 puerto HDMI, un puerto de cámara CSI para conectar una cámara Raspberry Pi, un puerto de pantalla DSI para conectar una pantalla táctil Raspberry Pi y un puerto microSD para cargar sistema operativo y almacenar datos [5].



Figura 12. Raspberry Pi 3 Modelo B.

Fuente: Raspberry Pi Foundation (2018). Raspberry Pi 3 Model B [Figura]. Recuperado de <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>

La Raspberry Pi 3 Modelo B se alimenta por una fuente micro USB de +5.1V con una corriente máxima de 2.5A dependiendo de lo que se le conecte, típicamente una entre 700mA y 1000mA<sup>2</sup>.

## 2.4. NAVIO Versión 2

La placa Navio2 que se muestra en la Figura 13, es un piloto automático para Raspberry Pi con tecnología Ardupilot (se explica en la sección 2.5) que permite convertir la Raspberry Pi en un controlador de drones, utilizando sensores y controladores abordo.



Figura 13. Piloto automático Navio2.

Fuente: Emlid Ltd (2018). Turn your Raspberry Pi® into a drone controller [Figura]. Recuperado de <https://emlid.com/navio/>

---

<sup>2</sup> R. P. FOUNDATION, «RASPBERRY PI 3 MODEL B,» [En línea]. Available: <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>. [Último acceso: 28 octubre 2018].

Tiene un barómetro integrado MS5611, un módulo GPS U-blox M8N GLONASS/GPS/Beidou con conector de antena tipo MCX, una unidad de medición inercial LSM9DS1 de 9 grados de libertad, una unidad de medición inercial MPU9250 de 9 grados de libertad, un puerto para módulo de poder, un puerto UART, un puerto I2C, 12 salidas servo PWM, una entrada PPM/S.Bus y un puerto ADC [6].

Navio2 tiene tres fuentes de alimentación (módulo de poder, USB, riel del servo), todas pueden usarse simultáneamente debido a que están protegidas con diodos ideales. Estas fuentes utilizan un voltaje de alimentación en el rango de 4.8-5.3V. Para desarrollo y pruebas puede conectarse el adaptador de corriente de puerto micro USB de la Raspberry Pi.

## **2.5. Ardupilot.**

Ardupilot<sup>3</sup> es un software para pilotos automáticos de código abierto, puede controlar cualquier sistema de vehículo como por ejemplo drones quadrotor.

Permite monitorear la posición del GPS y comunicarse en tiempo real, con aplicaciones de control terrestre que se ejecutan en dispositivos móviles, laptops o computadoras de escritorio.

Posee muchos modos de vuelo que se adaptan a las necesidades del usuario y proporciona opciones a prueba de fallos.

### **2.5.1. Modo guiado.**

Uno de los modos de vuelo es el modo guiado o "GUIDED", este modo de vuelo permite desplazar un dron al recibir las coordenadas de la ubicación deseada en el espacio tridimensional mediante una conexión con el protocolo de comunicación MAVLink a través de telemetría [7]. Al llegar a esta ubicación el quadrotor espera

---

<sup>3</sup> A. D. Team, «ArduPilot» [En línea]. Available: <http://ardupilot.org/ardupilot/index.html>. [Último acceso: 30 octubre 2018].

nuevas ordenes provenientes de la estacion de control terrestre y tratará de mantener su posicion actual. Este modo se activa luego de seleccionarlo de un listado predefinido de modos de vuelo y ejecutando el arranque del dispositivo. Cuando este modo de vuelo esta activado y se ha realizado un despegue el dispositivo espera ordenes manteniendo su altitud y posicion segun sensores.

### 2.5.2. Implementación PID.

Los parámetros que configuran el controlador PID permiten la estabilidad en vuelo de un dron quadrotor, convirtiendo la velocidad de rotación deseada en salida para cada rotor. La Figura 14 muestra el control de cabeceo (Pitch), alabeo (Roll) y guiñada (Yaw), donde se utiliza un controlador P para convertir el error entre el angulo deseado y el angulo real en una velocidad de rotación deseada junto a un prealimentador, seguido de un controlador PID para convertir el error entre la velocidad de rotación deseada y la velocidad de rotación real en un comando para cada rotor. La constante P multiplica el error por un número escalar para acercar el valor de velocidad de rotación real al valor de velocidad de rotación deseado, la constante D amortigua la respuesta de la constante P, la constante I reduce al máximo el error en estado de equilibrio y la constante FF es una pre alimentacion del sistema.

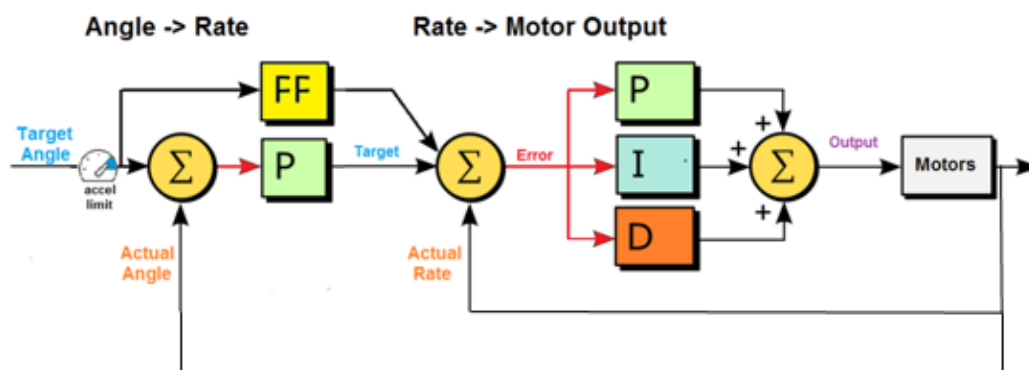


Figura 144. Diagrama de bloques control PID para estabilidad de rotación de cada eje.

Fuente: ArduPilot Dev Team (2018). Copter Attitude Control [Figura]. Recuperado de <http://ardupilot.org/dev/docs/apmcopter-programming-attitude-control-2.html>

### 2.5.3. Filtro de Kalman Extendido (EKF)<sup>4</sup>.

Es un algoritmo para estimar la posición, velocidad y aceleración de un dron quadrotor en función de las mediciones tomadas de las IMU's, barómetro y GPS. Esta estimación une todas estas mediciones para eliminar todos los errores significativos, haciendo al dron quadrotor menos susceptible a las fallas de un sensor en particular. Si el contronador de vuelo tiene dos o mas IMU's disponibles, se ejecutan dos instancias EKF cada una utilizando un IMU diferente, para luego utilizar la instancia EKF que presente datos mas consistentes de su sensor.

### 2.6. DroneKit-Python<sup>5</sup>

DroneKit-Python es un proyecto de código abierto el cual permite crear aplicaciones para el control de vehiculos aereos no tripulados como un dron quadrotor, a traves de una computadora o laptop y que se comunican mediante un enlace de telemetria utilizando el protocolo de comunicación MAVLink<sup>6</sup>, proporcionando información sobre el estado y parametros del vehículo [8]. Posee muchos scripts ejemplos que permiten asimilar mejor el comportamiento de cada uno de sus comandos y un simulador para realizar pruebas.

### 2.7. Biblioteca OpenCV<sup>7</sup>.

OpenCV es una biblioteca de código abierto dedicada al procesamiento de imágenes y visión artificial, con una interfaz de lenguaje de programación para C, C++, Andriod y Python; compatible con Linux y Windows [9].

---

<sup>4</sup> A. D. Team, « Extended Kalman Filter (EKF)» [En línea]. Available: <http://ardupilot.org/copter/docs/common-apm-navigation-extended-kalman-filter-overview.html>. [Último acceso: 30 octubre 2018].

<sup>5</sup> Robotics, «About DroneKit» [En línea]. Available: <http://python.dronekit.io/about/overview.html>. [Último acceso: 31 octubre 2018].

<sup>6</sup> Dronecode Project, «MAVLink Developer Guide» [En línea]. Available: <https://mavlink.io/en/> [Último acceso: 31 octubre 2018].

<sup>7</sup> o. d. team, «Face Recognition with OpenCV,» [En línea]. Available: [https://docs.opencv.org/2.4/modules/contrib/doc/facerec/facerec\\_tutorial.html#](https://docs.opencv.org/2.4/modules/contrib/doc/facerec/facerec_tutorial.html#). [Último acceso: 31 octubre 2018]

OpenCV permite detectar objetos, incluso rostros en una imagen o video usando clasificadores en cascada basados en las características de Haar del objeto que se desea encontrar, tal como se observa en la Figura 15. Estas características pueden encontrarse definidas dentro de archivos clasificadores con extensión .xml almacenados en las librerías de OpenCV. Una de las características para detectar un rostro podría ser que la región de los ojos es mas oscura que la nariz y las mejillas o que la región de los ojos es mas oscura que el puente de la nariz.

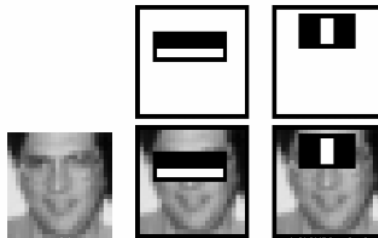


Figura 155. Algunas Características Haar para la detección de rostros.

Fuente: OpenCV (2018). Face Detection using Haar Cascades [Figura]. Recuperado de [https://docs.opencv.org/3.4.1/d7/d8b/tutorial\\_py\\_face\\_detection.html](https://docs.opencv.org/3.4.1/d7/d8b/tutorial_py_face_detection.html)

Las versiones OpenCV 2.4.x incorporan una clase para reconocimiento facial utilizando tres métodos. El primero de estos métodos es el método EigenFace el cual utiliza el análisis de componentes principales (PCA) para realizar una proyección lineal que maximiza la variación del espacio de imágenes a un espacio de características de menor dimensión. El segundo método es el FisherFace, el cual utiliza el discriminante lineal de Fisher (FLD), a estos métodos se les denomina holísticos, pues toman toda la región de la cara como dato de entrada para el reconocimiento y por último se encuentra el método “histogramas de patrones locales binarios” LBPH, este método resume la estructura local en una imagen comparando cada píxel con su vecindario.



---

## **CAPÍTULO 3. DISEÑO DEL PROYECTO**

---

### 3.1. Configuración física del dron quadrotor.

La configuración física del dron quadrotor para este trabajo de graduación se muestra en la Figura 16 y está formada basicamente de un marco DJI F450, un kit de propulsión DJI E305, un placa Navio2, una Raspberry Pi 3 modelo B, un kit de radio telemetría para 433MHz y una batería tipo LiPo de 6000mAh.

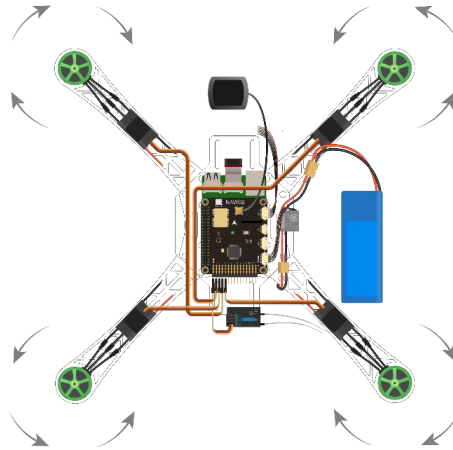


Figura 16. Esquema de configuración dron Quadrotor.

Fuente: EMLID (2018). Typical setup schemes [Figura]. Recuperado de <https://docs.emlid.com/navio2/ardupilot/typical-setup-schemes/>

Como primer paso se ajusta la placa Navio2 sobre la Raspberry Pi 3 Modelo B, y luego el conjunto se coloca sobre una estructura para amortiguar las vibraciones que producen errores de medición en los sensores de la placa Navio2 tal como se muestra en la Figura 17.

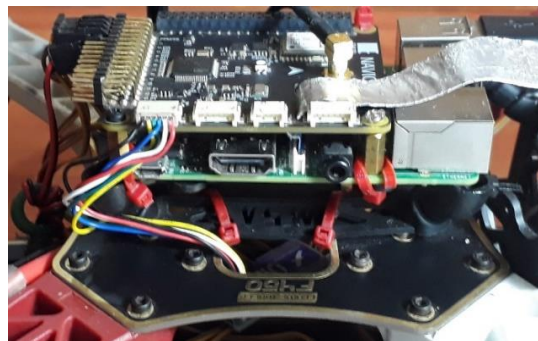


Figura 17. Montaje antivibraciones Raspberry Pi.

### 3.1.1. Sistema de potencia.

El módulo de potencia mostrado en la Figura 18, admite baterías de hasta 6S (6 celdas), tiene conectores para alimentación general XT60 los cuales permiten una corriente máxima de 60A. También, posee una salida de comunicación de 5.3V/2.25A, la cual utiliza un cable DF13 de 6 polos para la conexión con el puerto “POWER” de la placa Navio2. La Raspberry Pi también es alimentada a través de la placa Navio2.



Figura 188. Módulo de potencia para Navio2.

Fuente: EMLID (2018). Navio2 power module [Figura]. Recuperado de <https://store.emlid.com/product/power-module/>

Este módulo de potencia permite censar la tensión y corriente que circula hacia el dron quadrotor.

Para el caso en que el módulo de potencia falle, el dron quadrotor cuenta con un alimentador de respaldo desde el riel de servos conocido como UBEC de 3<sup>a</sup> mostrado en la Figura 19, el cual provee energía a partir de la tensión DC proveniente de la batería y lo convierte en una tensión DC segura para la placa Navio2.

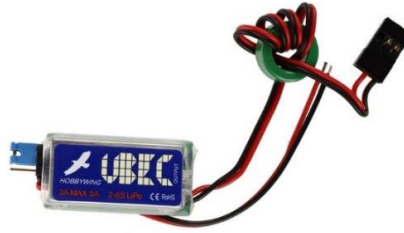


Figura 199. Regulador DC UBEC de 3A.

Fuente: Ebay (2018). Hobbywing UBEC 3A con salida de control de radio RF de reducción de ruido eliminador de batería para el modo de conmutación de circuito Lipo AD [Figura]. Recuperado de <https://www.ebay.com/itm/Hobbywing-3A-UBEC-w-RF-Noise-Reduction-RC-Output-BEC-Switch-Mode-for-Lipo-AD-/112216294451>

### 3.1.2. Batería tipo LiPo.

La batería de polímero de iones de litio (LiPo) utilizada en este trabajo de graduación es de 6000mAh personalizada con 3 celdas (3s/11.1V) tal como se muestra en la Figura 20, trae un conector XT60 y un conector para balanceo JST-XH.



Figura 20. Batería tipo LiPo.

Fuente: Unmanned Technology S.L. (2018). Batería LiPo U-TECH PRO 3s 11.1V 6000mAh 30C [Figura]. Recuperado de <https://rc-innovations.es/Baterias-lipo-3s-calidad-drones/bateria-lipo-u-tech-pro-3s-11.1v-6000mah-30c-alta-duracion>

Cada una de las celdas de las baterías tipo LiPo provee una tensión promedio de 3.7V. Este tipo de batería puede llegar a permitir hasta 300 ciclos de carga y descarga con un buen cuidado, pero se debe evitar una descarga total de la batería, pues provoca daños en las celdas hasta el punto de no permitir la carga. Comparado con las baterías de NiCd/NiHm, las baterías tipo LiPo son la mejor opción para el

vuelo de dispositivos eléctricos pues son ligera, pueden ser de cualquier forma o tamaño, son de gran capacidad y tienen una tasa de descarga alta. Este tipo de baterías requieren un cargador especial que permita una carga balanceada, donde cada celda sea cargada de manera independiente.

### 3.1.3. Sistema de propulsión DJI E305.

Este sistema de propulsión está formado por 4 motores 2312E (2 con rotación CW y 2 con rotación CCW), 4 propelas 9450 de 9.5 pulgadas de diámetro (2 con rotación CW y 2 con rotación CCW) y 4 controladores electrónicos de velocidad 420LITE. Se puede observar en la Figura 21 que cada rotor se coloca junto con su controlador electrónico de velocidad en la parte externa de cada brazo de la estructura de soporte.



Figura 21. Montaje sistema de propulsión E305.

Los controladores electrónicos de velocidad se conectan en las salidas del riel de servos de la placa Navio2 tal como se muestra en la Figura 22.

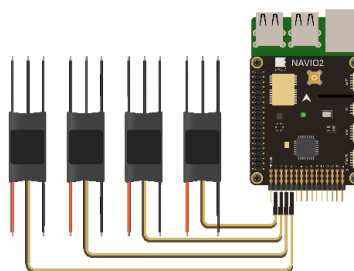


Figura 22. Conexión controladores electrónicos de velocidad con placa Navio2.

Fuente: EMLID (2018). RC output ESCs [Figura]. Recuperado de <https://docs.emlid.com/navio2/ardupilot/hardware-setup/>

### 3.1.4. Sistema GNSS.

El módulo GPS integrado en la placa Navio2 requiere la instalación de una antena activa con conector MCX.

Los dispositivos electrónicos a bordo del dron quadrotor pueden producir ruido de RF que afecta la calidad de recepción en la señal GPS, por esta razón la antena se coloca entre 15-20cm sobre la placa Navio2 tal como se muestra en la Figura 23. Además, se coloca un plano de tierra de 12cm de diámetro bajo la antena para proporcionar blindaje y mejorar la calidad de recepción de señal GPS.

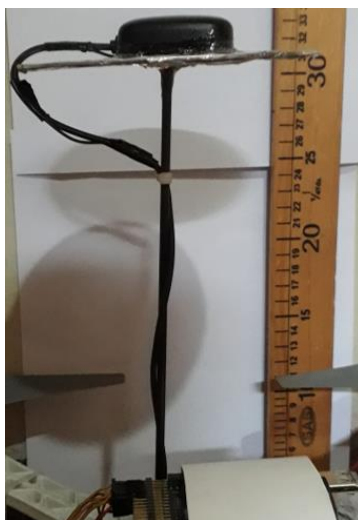


Figura 23. Montaje antena GPS.

### 3.2. Calibración de sensores.

Para realizar calibración de sensores se debe utilizar un enlace de telemetría vía radio y la aplicación Qgroundcontrol. Esta aplicación permite realizar una calibración de acelerómetro, brújula, barómetro y nivel de horizonte, únicamente se siguen las instrucciones dadas por la aplicación para lograr la calibración de los distintos sensores.

Una vez realizada la calibración de sensores no será necesario realizarla de nuevo, a menos que se realice un cambio en la estructura del dron quadrotor.

### 3.3. Sistema de adquisición de imágenes para la base de datos.

Este sistema depende del código `capture.py`, el cual, no necesita que el dron quadrotor este en vuelo. Se encuentra almacenado dentro de la carpeta `SEGUIRECOFACIAL`, cuya estructura es la siguiente:

```
SEGUIRECOFACIAL
  Rostros
    nombre1
    capture.py
    haarcascade_frontalface_default
    SEGUIRECOFACIAL.py
```

Al momento de ejecutar el código desde terminal se pasa como parámetro el nombre de la persona a ingresar en la base de datos. Se genera una carpeta con el nombre de la persona dentro de la carpeta rostros y en esta nueva carpeta se almacena una secuencia de 20 fotografías del rostro detectado por la cámara en cada ejecución del código.

### 3.4. Sistema de reconocimiento y seguimiento facial.

Este sistema es la estructura principal del trabajo de graduación depende del código “`SEGUIRECOFACIA.py`” cuya ejecución se muestra en la Figura 24. Este código permite la ejecución de múltiples procesos de manera simultánea. Muestra la imagen que transmite la cámara Pi en la pantalla principal que se genera al ejecutar el código y realiza una detección de rostros en la zona delimitada por el rectángulo grande. En la parte inferior aplica los 3 métodos de reconocimiento facial. Al detectarse un rostro muestra el nombre de la persona identificada y el nivel de diferencia con el rostro almacenado en la base de datos que posee características similares al rostro detectado por la cámara. En la parte inferior izquierda muestra una serie de parámetros de estado y de ubicación del dron quadrotor. Tiene un botón azul **Tomar Foto** que toma una captura instantánea de la imagen de video. El botón **Armar y despegar** pone al dron quadrotor en modo guiado y enciende los motores hasta que este alcanza la Altura relativa deseada. El botón **RTL** desplaza el dron

quadrotor al punto de despegue y luego desciende hasta llegar al suelo. El boton Aterrizar desciende el dron quadrotor hasta llegar al suelo. Los botones de seguimiento habilitan y deshabilitan el desplazamiento del dron quadrotor de acuerdo a la detección de rostros. El botón Ir a Ubicación desplaza el dron quadrotor a las coordenadas geográficas ingresadas en los campos de Ubicación Relativa Global deseada. El botón Ir a Posición desplaza el dron quadrotor a las coordenadas geográficas locales ingresadas en los campos de Posición Objetivo Local NED (Norte – Este - Abajo). El boton Rotar gira el dron quadrotor paralelo al plano de tierra según se indique en Yaw Absoluto (°).

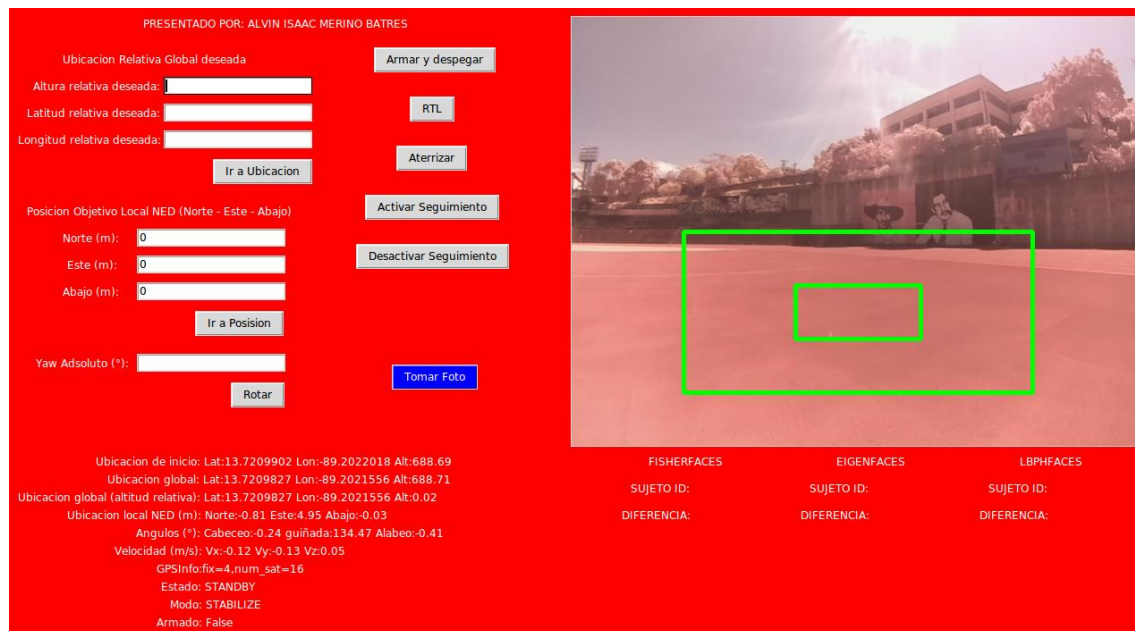


Figura 24. Captura ejecución código SEGUIRECOFACIAL.py.

Al activar el modo seguimiento se puede seguir un rostro que se encuentre a una distancia no mayor a 2 metros respecto a la cámara, para distancias mayores es necesario utilizar una cámara con mayor resolución de megapíxeles o adicionar un lente con aumento de 8mm a la cámara. En modo seguimiento, el dron quadrotor tratará de mantener dentro del rectángulo pequeño el centro del rostro.



---

## **CAPÍTULO 4. RESULTADOS**

---

#### 4.1. Verificación de la calidad de la señal GPS.

Dentro del repositorio Navio se encuentra la aplicación ublox-spi-to-tcp que al ejecutarse en la Raspberry Pi permite al programa u-center (ejecutado desde la computadora de control terrestre) conectarse al módulo GPS ublox integrado en la placa Navio2. Para este propósito se deben ejecutar las siguientes líneas de comandos:

```
tesis@eie.fia.ues :/ > cd Navio/Utilities/ublox-spi-to-tcp
```

```
tesis@eie.fia.ues :/ > make
```

```
tesis@eie.fia.ues :/ > ./ublox-spi-to-tcp 5000
```

Luego de ejecutar estos comandos espera una conexión. En u-center se navega en la pestaña Receiver - Port - Network connection - New para introducir la IP de la Raspberry Pi y el puerto de enlace 5000. Al realizar este procedimiento se obtiene la pantalla mostrada en la Figura 25.

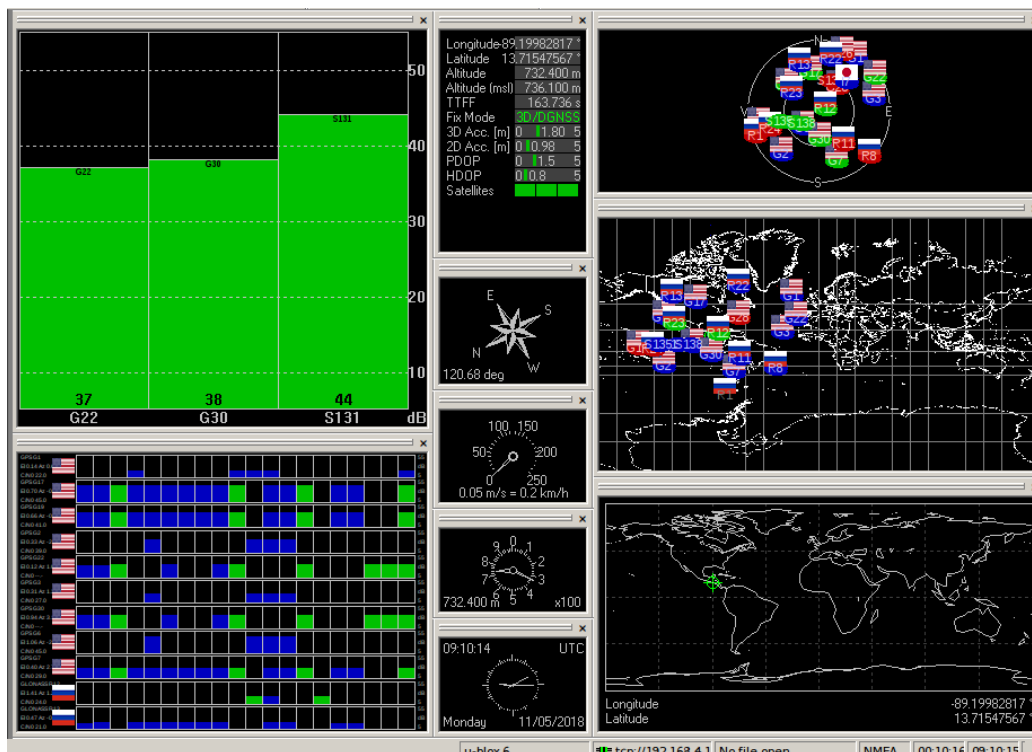


Figura 25. Captura de pantalla al ejecutar u-center en PC.

u-center muestra un modo de corrección 3D/DGNSS, lo que indica que se ha logrado una solución de posición con al menos 4 satélites y las correcciones se proporcionan desde una fuente que mide las diferencias entre los valores de pseudodistancia para cada satélite en la estación de referencia topográfica. La incertidumbre horizontal es de 0.8 y la incertidumbre 3D es de 1.5.

#### 4.2. Comparación métodos de reconocimiento facial.

Al comparar los tres métodos para el reconocimiento facial tal como se observa en la Figura 26, el método con mayor dificultad para reconocer un rostro es el método de EigenFaces y que el método que menos diferencias presenta al reconocer rostros es el método de LBPH. El método FisherFaces logra hacer una identificación de rostro, pero muestra un valor de diferencia superior al método de LBPH.

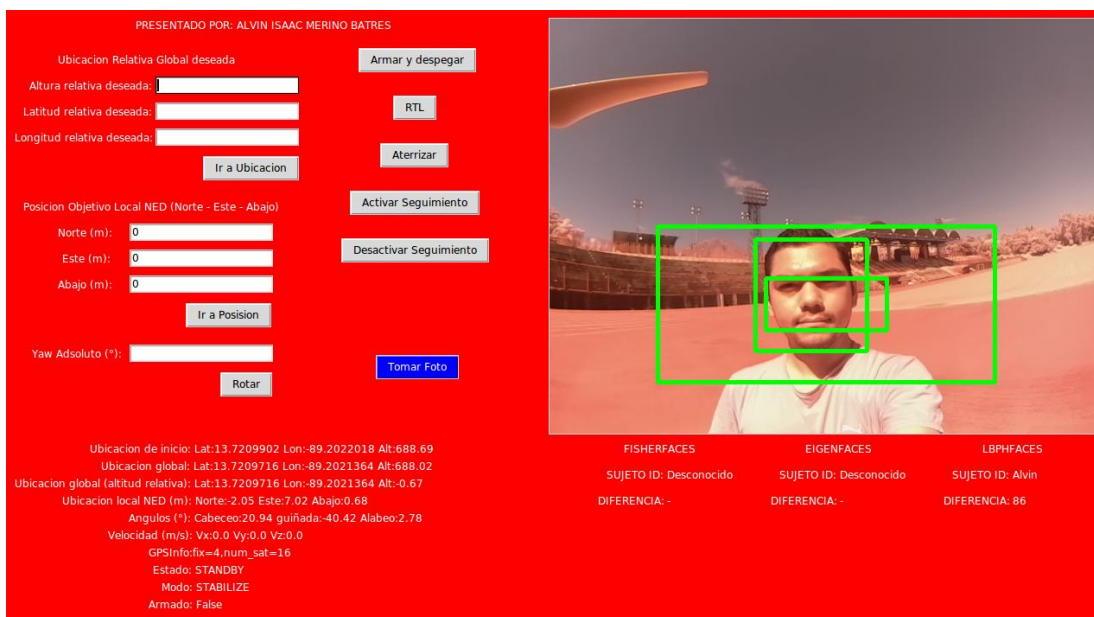


Figura 26. Comparación métodos de reconocimiento facial.

#### 4.3. Características dron quadrotor.

El dron quadrotor realizado en este trabajo de graduación mostrado en la Figura 27, permite un tiempo de vuelo aproximado de 10 minutos, pero, para aumentar este tiempo de vuelo es necesario aumentar la capacidad de la batería.

Posee una cámara Pi de 5 megapíxeles con led infrarrojos, un sistema de controladores PID que proporcionan estabilidad en los 3 ejes de rotación y al momento de acelerar o desacelerar. Es controlado a partir de una aplicación que permite ejecutar misiones en modo guiado utilizando navegación por GPS.



Figura 27. Dron quadrotor construido.

## CONCLUSIONES.

- El método LBPH para el reconocimiento facial tiene un rendimiento superior en comparación con los otros dos métodos holísticos, debido a que tiene la cualidad de extraer las características locales de la imagen de un rostro.
- La aplicación de sensores para la estabilidad de vuelo del dron quadrotor desarrollado en este trabajo de graduación, permite cierta autonomía en el manejo del mismo.
- Al aplicar principios de sistemas de control automático en este trabajo de graduación se reduce el riesgo de accidentes provocados por un mal manejo del dispositivo.
- Al aplicar un sistema de control automático de balanceo del dispositivo para mejorar la estabilidad ante los estímulos externos durante el vuelo, permite mantener al dispositivo paralelo al suelo.
- Debido a las variaciones en la medición de la posición global, el sistema de corrección de posición es deficiente y requiere un sistema de mejoramiento avanzado, utilizando un segundo receptor GPS, para obtener la mayor precisión posible de la posición global.
- Con la aplicación del filtro Kalman Extendido para proporcionar una estimación de altitud, utilizando los sensores barómetro y GPS, se obtiene una medición más precisa de la altitud del dron quadrotor.
- Al utilizar una mayor cantidad de imágenes almacenadas en la base de datos de cada persona para realizar un reconocimiento facial, se mejora la precisión en la identificación de personas.
- Mientras mayor sea la resolución de megapíxeles de la cámara a bordo del dron quadrotor, mejor será el rendimiento de los métodos para el reconocimiento facial de personas de la biblioteca OpenCV.
- Al colocar una cámara a bordo de un dron quadrotor se obtiene una amplia cobertura de visión y libertad de movimiento en comparación a una cámara colocada sobre una pared o poste.

## RECOMENCACIONES.

- Para próximos trabajos de graduación en esta línea, se recomienda utilizar una protección contra la lluvia, de lo contrario, puede ocasionarse graves daños en el sistema eléctrico del dron quadrotor.
- Se recomienda a los próximos trabajos de graduación en esta línea, proteger el barómetro de los rayos UV y corrientes de aire para evitar errores de medición y daños.
- Se recomienda a los próximos trabajos de graduación en esta línea, establecer un enlace de telemetría vía serial para calibrar sensores.
- Se recomienda a los próximos trabajos de graduación en esta línea, colocar el regulador de voltaje a una distancia no menor a 5cm de la placa Navio2.
- Se recomienda a los próximos trabajos de graduación en esta línea, envolver el cable de la Cámara Pi con aluminio para evitar que afecte el rendimiento de la antena GPS.
- Se recomienda a los próximos trabajos de graduación en esta línea, previo a despegue verificar que el sentido de giro de cada uno de los motores es el correcto, si no es así, cambiar la configuración de conexión con los controladores electrónicos de velocidad hasta que los motores giren en el sentido correcto.
- Se recomienda a los próximos trabajos de graduación en esta línea, almacenar las imágenes en la base de datos con el nombre de la persona a la que pertenecen.
- Se recomienda a los próximos trabajos de graduación en esta línea, generar más de 60 capturas de rostro en la base de datos de una persona para optimizar el reconocimiento facial.
- Se recomienda a los próximos trabajos de graduación en esta línea, verificar en la base de datos que las imágenes en cada carpeta pertenecen a la persona correspondiente.

- Se recomienda a los próximos trabajos de graduación en esta línea, utilizar una memoria micro SD de 16GB de almacenamiento clase 10 HC.
- Se recomienda a los próximos trabajos de graduación en esta línea, utilizar la imagen del sistema modificada por EMLID.
- Los próximos trabajos de graduación en esta línea, se recomienda evitar cometer el error de despegar el dispositivo sin realizar el prearmado.
- Para todos los trabajos de graduación que en el futuro sigan esta línea, se recomienda utilizar dos receptores de señal GPS para obtener mejores resultados en la medición de la posición.

## BIBLIOGRAFÍA.

- [ Jeremy Norman & Co., Inc., «Woodrow Bledsoe Originates of Automated Facial  
1 Recognition,» [En línea]. Available:  
] <http://www.historyofinformation.com/expanded.php?id=2495>. [Último  
acceso: 23 Septiembre 2018].
- [ J. Sturm, «Edx, Autonomous Navigation for Flying Robots, Lecture 1.3: Flying  
2 Principle,» [En línea]. Available: [https://courses.edx.org/courses/course-  
\] v1:TUMx+AUTONAVx+2T2015/courseware/535451105f364d2e852366ed8204cf  
68/5767c53391344068a583a8fa7fe79dad/1?activate\\_block\\_id=block-  
v1%3ATUMx%2BAUTONAVx%2B2T2015%2Btype%40vertical%2Bblock%40e4a  
0c8b069234c9e8782ffef0d5401e](https://courses.edx.org/courses/course-v1:TUMx+AUTONAVx+2T2015/courseware/535451105f364d2e852366ed8204cf68/5767c53391344068a583a8fa7fe79dad/1?activate_block_id=block-v1%3ATUMx%2BAUTONAVx%2B2T2015%2Btype%40vertical%2Bblock%40e4a0c8b069234c9e8782ffef0d5401e). [Último acceso: 23 Septiembre 2018].
- [ N. a. T. (. National Executive Committee for Space-Based Positioning, «Official  
3 U.S. government information about the Global Positioning System (GPS) and  
] related topics,» [En línea]. Available: <https://www.gps.gov/systems/gps/>.  
[Último acceso: 10 Octubre 2018].
- [ U. D. o. Defense, «GLOBAL POSITIONING SYSTEM STANDARD  
4 POSITIONING SERVICE PERFORMANCE STANDARD, cuarta edición,» [En  
] línea]. Available: [https://www.gps.gov/technical/ps/2008-SPS-performance-  
standard.pdf](https://www.gps.gov/technical/ps/2008-SPS-performance-standard.pdf). [Último acceso: 11 Octubre 2018].
- [ R. P. FOUNDATION, «RASPBERRY PI 3 MODEL B,» [En línea]. Available:  
5 <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>. [Último  
] acceso: 28 Octubre 2018].
- [ EMLID, «NAVIO2,» [En línea]. Available: <https://emlid.com/navio/>. [Último  
6 acceso: 28 Octubre 2018].  
]



[ A. D. Team, «Guided Mode,» [En línea]. Available:  
7 [http://ardupilot.org/copter/docs/ac2\\_guidedmode.html](http://ardupilot.org/copter/docs/ac2_guidedmode.html). [Último acceso: 30  
] Octubre 2018].

[ 3. Robotics, «About DroneKit,» [En línea]. Available:  
8 <http://python.dronekit.io/about/overview.html>. [Último acceso: 31 Octubre  
] 2018].

[ o. d. team, «Face Recognition with OpenCV,» [En línea]. Available:  
9 [https://docs.opencv.org/2.4/modules/contrib/doc/facerec/facerec\\_tutorial.ht  
\] ml#](https://docs.opencv.org/2.4/modules/contrib/doc/facerec/facerec_tutorial.html#). [Último acceso: 31 Octubre 2018].

## ANEXOS.

### ANEXO A. Configuración Raspberry Pi 3 modelo B.

Para cargar el sistema operativo este trabajo de graduación utiliza una memoria micro SD HC clase 10 de 16GB. La imagen de este sistema operativo es una versión modificada de Raspbian (Emlid Raspbian) proporcionada por el siguiente enlace:

<http://files.emlid.com/images/emlid-raspbian-20180525.img.xz>

Esta imagen se escribe en la memoria micro SD utilizando la aplicación Etcher, disponible para Ubuntu y Windows.

Para establecer un enlace de comunicación vía SSH (protocolo y programa que permite acceso remoto a un servidor por medio de un canal seguro, en el que toda la información está cifrada) en el primer arranque, es necesario conectar la Raspberry Pi a una red de área local por medio de su conector Ethernet y obtener su dirección ip para luego ejecutar el siguiente comando desde un terminal en una computadora con sistema operativo Ubuntu 16.04 conectada en la misma red:

```
tesis@eie.fia.ues :/ > sudo ssh -X pi@<dirección ip de Raspberry Pi>
```

El usuario para ingresar es pi y la contraseña es Raspberry. Al ingresar al sistema de la Raspberry Pi se realiza una actualización con el siguiente comando:

```
tesis@eie.fia.ues :/ > sudo apt-get update && sudo apt-get dist-upgrade
```

Luego es necesario expandir el sistema de archivos para ocupar todo el espacio disponible en la memoria micro SD utilizando el siguiente comando:

```
tesis@eie.fia.ues :/ > sudo raspi-config --expand-rootfs
```

Y se reinicia el sistema.

Al restablecer la conexión SSH se procede con la configuración de la Raspberry Pi como un punto de acceso inalámbrico, se inicia ejecutando las siguientes ordenes en la línea de comandos del sistema:

```
tesis@eie.fia.ues :/ > sudo apt-get install hostapd
```

```
tesis@eie.fia.ues :/ > sudo apt-get install dnsmasq
```

Es necesario modificar los archivos de configuración de estos programas y para ello se apagan con la ejecución de las líneas siguientes:

```
tesis@eie.fia.ues :/ > sudo systemctl stop dnsmasq
tesis@eie.fia.ues :/ > sudo systemctl stop hostapd
```

A continuación, se procede a configurar una IP estática para la interfaz wlan0 editando el archivo /etc/dhcpd.conf con el siguiente comando:

```
tesis@eie.fia.ues :/ > sudo nano /etc/dhcpd.conf
```

dentro del archivo, se agregan las líneas siguientes:

```
interface wlan0
    static ip_address=192.168.4.1/24
    nohook wpa_supplicant
```

Después de ingresar las líneas se guardan los cambios realizados en el archivo. Se reinicia el servicio con el siguiente comando:

```
tesis@eie.fia.ues :/ > sudo service dhcpd restart
```

Se utiliza dnsmasq como servidor DHCP para permitir la conexión de al menos 2 equipos al punto de acceso inalámbrico. Se crea una copia del archivo de configuración original de dnsmasq como respaldo y se crea uno nuevo con los comandos siguientes:

```
tesis@eie.fia.ues :/ > sudo mv /etc/dnsmasq.conf /etc/dnsmasq.conf.orig
tesis@eie.fia.ues :/ > sudo nano /etc/dnsmasq.conf
```

El contenido del nuevo archivo es el siguiente:

```
interface=wlan0
    dhcp-range=192.168.4.2,192.168.4.3,255.255.255.0
```

Luego de guardar los cambios se procede a crear el archivo de hostapd para configurar la Raspberry Pi como punto de acceso inalámbrico con el comando siguiente:

```
tesis@eie.fia.ues :/ > sudo nano /etc/hostapd/hostapd.conf
```

El contenido de este archivo es el siguiente:

```
interface=intwifi0
```

```
ssid=pi_AP
hw_mode=g
channel=6
ieee80211n=1
wmm_enabled=1
ht_capab=[HT40][SHORT-GI-20][DSSS_CCK-40]
macaddr_acl=0
auth_algs=1
ignore_broadcast_ssid=0
wpa=2
wpa_passphrase=al70567840
wpa_key_mgmt=WPA-PSK
rsn_pairwise=CCMP
```

Es necesario indicarle al sistema la ubicación de este nuevo archivo y para ello se sustituye la línea: `#DAEMON_CONF=""` por `DAEMON_CONF="/etc/hostapd/hostapd.conf"` ubicada dentro del archivo `/etc/default/hostapd`.

Se reinician los programas con los comandos:

```
tesis@eie.fia.ues :/ > sudo systemctl start hostapd
tesis@eie.fia.ues :/ > sudo systemctl start dnsmasq
```

Para permitir reenviar el tráfico a través del cable Ethernet cuando este se conecte, se elimina el símbolo “#” de la línea `#net.ipv4.ip_forward=1` en el archivo `/etc/sysctl.conf`.

A continuación, se agrega enmascaramiento de IP para el tráfico saliente en `eth0` usando `iptables`:

```
tesis@eie.fia.ues :/ > sudo iptables -t nat -A POSTROUTING -o enx827eb51b53a -j MASQUERADE
```

Y se guarda la nueva regla de `iptables`:

```
tesis@eie.fia.ues :/ > sudo sh -c "iptables-save > /etc/iptables.ipv4.nat"
```

Para cargar la regla en el arranque, es necesario editar el archivo `/etc/rc.local` y agregar la siguiente línea antes de `exit 0`:

```
iptables-restore < /etc/iptables.ipv4.nat
```

Para elegir el tipo de vehículo (en este trabajo de graduación un quadrotor) y la versión de Ardupilot, se ejecuta el siguiente comando:

```
tesis@eie.fia.ues :/ > sudo emlidtool ardupilot
```

Este comando abrirá una nueva ventana, en la cual, se selecciona en vehículo copter, la versión 3.5, el marco arducopter, para iniciar Ardupilot en el arranque se selecciona enable, se inicia Ardupilot seleccionando start y luego apply.

Luego se modifica el archivo /etc/default/arducopter para definir el IP de la estación terrestre de control y el puerto USB para conectar el radio.

El archivo queda de la siguiente manera:

```
TELEM1="-A udp:192.168.4.3:14550"
TELEM2="-C /dev/ttyUSB0"
# Options to pass to ArduPilot
ARDUPILOT_OPTS="$TELEM1 $TELEM2"
# -A is a console switch (usually this is a Wi-Fi link)
# -C is a telemetry switch
# Usually this is either /dev/ttyAMA0 - UART connector on your Navio
# or /dev/ttyUSB0 if you're using a serial to USB convertor
# -B or -E is used to specify non default GPS
```

Para clonar el repositorio de ejemplos de la placa Navio2 se ejecuta el siguiente comando:

```
tesis@eie.fia.ues :/ > git clone https://github.com/emlid/Navio2.git
```

Para transmitir video en tiempo real es necesario habilitar la cámara utilizando el comando siguiente:

```
tesis@eie.fia.ues :/ > sudo raspi-config
```

Es necesario también instalar la paquetería necesaria para el manejo de la cámara Pi con el siguiente comando:

```
tesis@eie.fia.ues :/ > sudo pip install picamera
```

Luego se crea un archivo vacío para la transmisión de video en el arranque con el siguiente comando:

```
tesis@eie.fia.ues :/ > sudo touch /etc/systemd/system/raspicam.service
```

Y se agregan las siguientes líneas:

```
[Unit]
Description=raspivid
After=network.target
[Service]
ExecStart=/bin/sh -c "/usr/bin/python /home/pi/tesisAlvinPI/stream.py" [Install]
WantedBy=default.target
```

Después, se ejecutan estos comandos: `sudo systemctl daemon-reload` para que el sistema conozca este servicio, `sudo systemctl enable raspicam` para habilitar el servicio en el arranque y `sudo systemctl start raspicam` para probarlo.

### **Configuración Ubuntu 16.04.**

Esta configuración se realiza en la computadora de control terrestre.

Primero es necesario asegurarse que todo el sistema este actualizado:

```
tesis@eie.fia.ues :/ > sudo apt-get update
```

Luego se instalan todas las dependencias:

```
tesis@eie.fia.ues :/ > sudo apt install python-gi python-gi-cairo python-gi python-gi-cairo gir1.2-gtk-3.0
```

```
tesis@eie.fia.ues :/ > apt-file update
```

```
tesis@eie.fia.ues :/ > sudo apt-get install apt-file
```

```
tesis@eie.fia.ues :/ > apt-file find gstreamer-video #busca paquetes que hacen falta y que se deberán instalar
```

```
tesis@eie.fia.ues :/ > sudo apt-get install python-pip
```

```
tesis@eie.fia.ues :/ > sudo apt-get install build-essential cmake pkg-config python-dev libgtk2.0-dev libgtk2.0 zlib1g-dev libpng-dev libjpeg-dev libtiff-dev libjasper-dev libavcodec-dev swig unzip
```

```
tesis@eie.fia.ues :/ > sudo apt-get install checkinstall yasm
```

```
tesis@eie.fia.ues :/ > sudo apt-get install git gfortran
```

```
tesis@eie.fia.ues :/ > sudo apt-get install libtiff5-dev
```

```
tesis@eie.fia.ues :/ > sudo apt-get install libjpeg8-dev libpng12-dev
```

```
tesis@eie.fia.ues :/ > sudo apt-get install libavformat-dev libswscale-dev libdc1394-22-dev
```

```
tesis@eie.fia.ues :/ > sudo apt-get install libxine2-dev libv4l-dev
```

```

tesis@eie.fia.ues :/ > sudo apt-get install libgstreamer0.10-dev libgstreamer-plugins-
base0.10-dev
tesis@eie.fia.ues :/ > sudo apt-get install qt5-default libtbb-dev
tesis@eie.fia.ues :/ > sudo apt-get install libatlas-base-dev
tesis@eie.fia.ues :/ > sudo apt-get install libfaac-dev libmp3lame-dev libtheora-dev
tesis@eie.fia.ues :/ > sudo apt-get install libvorbis-dev libxvidcore-dev
tesis@eie.fia.ues :/ > sudo apt-get install libopencore-amrnb-dev libopencore-amrwb-dev
tesis@eie.fia.ues :/ > sudo apt-get install x264 v4l-utils
tesis@eie.fia.ues :/ > sudo apt-get install libprotobuf-dev protobuf-compiler
tesis@eie.fia.ues :/ > sudo apt-get install libgoogle-glog-dev libgflags-dev
tesis@eie.fia.ues :/ > sudo apt-get install libgphoto2-dev libeigen3-dev libhdf5-dev doxygen
tesis@eie.fia.ues :/ > sudo apt-get install python-imaging python-imaging-tk
tesis@eie.fia.ues :/ > sudo apt-get install libavdevice-dev libavutil-dev

```

A continuación, se descarga y descomprime el código fuente de OpenCV 2.4.13:

```

tesis@eie.fia.ues :/ > wget http://downloads.sourceforge.net/project/opencvlibrary/opencv-
unix/2.4.13.4/opencv-2.4.13.4.zip
tesis@eie.fia.ues :/ > unzip opencv-2.4.13.zip
tesis@eie.fia.ues :/ > cd opencv-2.4.13

```

Se debe generar el Makefile usando cmake y se instala OpenCV:

```

tesis@eie.fia.ues :/ > mkdir build
tesis@eie.fia.ues :/ > cd build
tesis@eie.fia.ues :/ > cmake -G "Unix Makefiles" -D CMAKE_CXX_COMPILER=/usr/bin/g++
CMAKE_C_COMPILER=/usr/bin/gcc -D CMAKE_BUILD_TYPE=RELEASE -D
CMAKE_INSTALL_PREFIX=/usr/local -D WITH_TBB=ON -D
BUILD_NEW_PYTHON_SUPPORT=ON -D WITH_V4L=ON -D
INSTALL_C_EXAMPLES=ON -D INSTALL_PYTHON_EXAMPLES=ON -D
BUILD_EXAMPLES=ON -D WITH_QT=ON -D WITH_OPENGL=ON -D
BUILD_FAT_JAVA_LIB=ON -D INSTALL_TO_MANGLED_PATHS=ON -D
INSTALL_CREATE_DISTRIB=ON -D INSTALL_TESTS=ON -D ENABLE_FAST_MATH=ON
-D WITH_IMAGEIO=ON -D BUILD_SHARED_LIBS=OFF -D WITH_GSTREAMER=ON ..
tesis@eie.fia.ues :/ > make
tesis@eie.fia.ues :/ > sudo make install

```

Luego de instalar es necesario configurar las librerías a manera de variables de entorno indicando ubicación para poder vincularlas a cualquier proyecto.

Es necesario abrir el archivo `/etc/ld.so.conf.d/opencv.conf` y escribir la siguiente ruta en él:

```
/usr/local/lib
```

Luego de guardar los cambios es necesario vincular las librerías recientemente creadas:

```
tesis@eie.fia.ues :/ > sudo ldconfig
```

Abrir el archivo `bash.bashrc`. Una vez abierto el archivo copiar las siguientes dos líneas al final del texto

```
PKG_CONFIG_PATH=$PKG_CONFIG_PATH:/usr/local/lib/pkgconfig  
export PKG_CONFIG_PATH
```

Finalmente se guardan los cambios.

Para instalar `dronekit` y `dronekit-sitl` se ejecutan las siguientes líneas de comandos:

```
tesis@eie.fia.ues :/ > pip instalar dronekit  
tesis@eie.fia.ues :/ > pip instalar dronekit-sitl
```

Para clonar el repositorio `dronekit-python` se ejecuta el siguiente comando:

```
tesis@eie.fia.ues :/ > git clone http://github.com/dronekit/dronekit-python.git
```



## **ANEXO B. Principios de operación de Acelerómetro lineal digital 3D, Giroscopio lineal digital 3D y Magnetómetro lineal digital 3D.**

Un acelerómetro lineal digital 3D, es capaz de medir la aceleración en las 3 dimensiones espaciales, conocidas como ejes X, Y, Z, pues la aceleración es un vector. Al existir un cambio de velocidad sobre un cuerpo en determinada dirección, se produce una aceleración que puede tener componentes en cada uno de los 3 ejes. Este tipo de sensores utiliza un efecto capacitivo para determinar la aceleración en cada uno de los ejes. El recinto de cada uno de los ejes se compone de dos capacitores que trabaja en conjunto. Cuando se aplica una aceleración, aumenta una capacitancia y la otra disminuye. Estos envían una señal eléctrica analógica a un convertidor analógico/digital de 16 bits en cada eje. Luego esta señal digital es condicionada con un factor de escala calibrado de fábrica e independiente de la tensión de alimentación.

Cuando un objeto gira alrededor de un eje se obtiene una velocidad angular y puede ser medida en  $^{\circ}/s$ . Un giroscopio consta de tres independientes masas de resonancia, las cuales son muy pequeñas. Cuando se hace girar el giroscopio esta masa se desplaza con los cambios de la velocidad angular, el efecto CORIOLIS provoca una vibración que se detecta por un capacitor y se convierte en señales eléctricas de muy bajas corrientes. El resultado de la señal es amplificada, modulada y filtrada para producir una tensión que es proporcional a la velocidad angular. Esta tensión es digitalizada utilizando convertidores analógicos/digitales de 16bits en cada eje.

Un Magnetómetro se compone de tres puentes wheatstone para medir los campos magnéticos, tanto la intensidad del campo como la dirección del mismo en los ejes X, Y, Z. con la potencia aplicada a los puentes, los sensores convierten cualquier campo magnético aplicado en una diferencia de tensión en la salida. Esta señal eléctrica es amplificada y enviada a un circuito aritmético que procesa la señal de cada sensor para luego convertirse en una señal digital a través de un convertidor analógico/digital de 16bits.

## **ANEXO C. Manual de usuario.**

### **Cuidados de utilización y mantenimiento.**

Este dron quadrotor no es apto para la manipulación de niños menores de edad.

Para utilizar en dron quadrotor Realizar vuelos en una zona suficientemente despejada para obtener bloqueo GPS 3D/GNSS.

Mantener siempre el control visual sobre el dron quadrotor cuando sea utilizado, para evitar que entre en contacto con personas, animales u objetos.

La utilización del dron quadrotor en espacios públicos no se encuentra autorizado.

Asegurarse de gestionar los permisos necesarios.

Utilizar el dron quadrotor respetando la vida privada de las personas.

Asegurarse que particulares no se encuentren cerca del despegue o aterrizaje del dron quadrotor.

Las hélices del dron quadrotor durante el vuelo pueden causar daños a las personas que intenten tomarlos. Mantener siempre una distancia entre las hélices y otras personas.

No obstruir los orificios de ventilación de los motores del dron quadrotor.

No volar el dron quadrotor cerca de líneas de alta tensión u otras zonas potencialmente peligrosas.

### **Cuidados de la batería.**

Si se produce una fuga de líquido de batería, impedir que entre en contacto con la piel y los ojos. Si entra en contacto con los ojos enjuagarse con abundante agua fría y consultar un médico.

No cargar una batería hinchada o que haya perdido líquido.

No sobrecargar la batería cuando esté completamente cargada, desconectar el cargador.

No cargar la batería cerca de materiales inflamables.

Recargar la batería a una temperatura de entre 0°C y 40°C.

## Utilización del dron quadrotor

Colocar el dron quadrotor en una superficie plana y seca, en una zona libre de obstáculos.

Conectar los terminales XT60 de la batería y del módulo de poder.

Conectar una laptop con Linux a la red wifi pi\_AP

Verificar calibración de sensores y prearmado de dispositivo con una aplicación de control terrestre (QGroundcontrol) en una laptop con linux.

Desde un terminal en la laptop con Linux ejecutar el siguiente comando:

```
tesis@eie.fia.ues :/ > python SEGUIRECOFACIAL.py -o SEGUIRECOFACIAL/ --connect  
192.168.4.2:14550
```

Luego de que aparezca la ventana principal ingresar la altura de despegue deseada en el campo **Altura relativa deseada** y presionar el botón **Armar y despegar**.

Para ir a una ubicación se ingresan las coordenadas geográficas globales o locales y se presiona el botón **Ir a Ubicación** o **Ir a Posición** respectivamente.

Para girar el dron quadrotor paralelo al suelo se ingresa el ángulo con referencia al norte en el campo **Yaw Absoluto (°)** y se presiona el botón **Rotar**.

Para tomar una captura de la pantalla de cámara de video se presiona el botón azul **Tomar Foto**.

Para aterrizar el dron quadrotor se presiona el botón **RTL** o **Aterrizar** y esperar que las hélices terminen de girar para desconectar el dron quadrotor de la batería.