



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

**DEPARTAMENTO DE CIENCIAS DE LA ENERGÍA Y
MECÁNICA**

CARRERA DE INGENIERÍA MECATRÓNICA

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL
TÍTULO DE INGENIERO MECATRÓNICO**

**TEMA: INVESTIGACIÓN DEL MÉTODO ÓPTIMO DE VISIÓN
ARTIFICIAL Y REDES NEURONALES PARA LA
IMPLEMENTACIÓN DEL SISTEMA DE ATERRIZAJE DE UN
CUADRICÓPTERO**

**AUTORES: LUIS MARCELO ROBALINO IBARRA
ALEX FABRICIO SEGOVIA FREIRE**

DIRECTOR: ING. DARÍO MENDOZA.

LATACUNGA

2017



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

**DEPARTAMENTO DE CIENCIAS DE LA ENERGÍA Y
MECÁNICA
CARRERA DE INGENIERÍA MECATRÓNICA**

CERTIFICACIÓN

Certifico que el trabajo de titulación **“INVESTIGACIÓN DEL MÉTODO ÓPTIMO DE VISIÓN ARTIFICIAL Y REDES NEURONALES PARA LA IMPLEMENTACIÓN DEL SISTEMA DE ATERRIZAJE DE UN CUADRICÓPTERO”**, realizado por los señores **LUIS MARCELO ROBALINO IBARRA** y **ALEX FABRICIO SEGOVIA FREIRE**, ha sido revisado en su totalidad y analizado por el software anti-plagio, el mismo cumple con los requisitos teóricos, científicos, técnicos, metodológicos y legales establecidos por la Universidad de las Fuerzas Armadas – ESPE, por lo tanto me permito acreditarlos y autorizar a los señores **LUIS MARCELO ROBALINO IBARRA** y **ALEX FABRICIO SEGOVIA FREIRE** para que lo sustenten públicamente.

Latacunga, 1 de marzo de 2017


ING. DARÍO MENDOZA
DIRECTOR



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

**DEPARTAMENTO DE CIENCIAS DE LA ENERGÍA Y
MECÁNICA
CARRERA DE INGENIERÍA MECATRÓNICA**

AUTORÍA DE RESPONSABILIDAD

Nosotros, **LUIS MARCELO ROBALINO IBARRA** y **ALEX FABRICIO SEGOVIA FREIRE**, con cédulas de identidad N^o 180446822-9 y 180478327-0, declaramos que este trabajo de titulación “**INVESTIGACIÓN DEL MÉTODO ÓPTIMO DE VISIÓN ARTIFICIAL Y REDES NEURONALES PARA LA IMPLEMENTACIÓN DEL SISTEMA DE ATERRIZAJE DE UN CUADRICÓPTERO**”, ha sido desarrollado considerando los métodos de investigación existentes, así como también se ha respetado los derechos intelectuales de terceros considerándose en las citas bibliográficas.

Consecuentemente declaramos que este trabajo es de nuestra autoría, en virtud de ello nos declaramos responsables del contenido, veracidad y alcance de la investigación mencionada.

Latacunga, 1 de marzo de 2017.

Luis Marcelo Robalino Ibarra

C.C.: 180446822-9

Alex Fabricio Segovia Freire

C.C.: 180478327-0



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

**DEPARTAMENTO DE CIENCIAS DE LA ENERGÍA Y
MECÁNICA
CARRERA DE INGENIERÍA MECATRÓNICA**

AUTORIZACIÓN

Nosotros, **LUIS MARCELO ROBALINO IBARRA** y **ALEX FABRICIO SEGOVIA FREIRE** autorizamos a la Universidad de las Fuerzas Armadas - ESPE, publicar en la Biblioteca Virtual de la institución el presente trabajo de titulación **“INVESTIGACIÓN DEL MÉTODO ÓPTIMO DE VISIÓN ARTIFICIAL Y REDES NEURONALES PARA LA IMPLEMENTACIÓN DEL SISTEMA DE ATERRIZAJE DE UN CUADRICÓPTERO”**, cuyo contenido, ideas y criterios son de nuestra autoría y responsabilidad.

Latacunga, 1 de marzo de 2017

Una firma manuscrita en tinta azul que parece decir 'Luis Robalino'.

Luis Marcelo Robalino Ibarra

C.C.: 180446822-9

Una firma manuscrita en tinta azul que parece decir 'Alex Segovia'.

Alex Fabricio Segovia Freire

C.C.: 180478327-0

DEDICATORIA

Este escrito lo dedico Principalmente a Dios por ser el que me guía y no me abandona aún entre las sendas más oscuras.

A mis padres por que, sin ellos, sin su amor y su esfuerzo mis sueños no serían más que meras ilusiones.

Lo dedico también a mi familia y amigos por darme su apoyo.

También a mis profesores que fueron luz he inspiración.

Y por último se lo dedico a quien encuentre en este documento una gota más de sabiduría para llenar su mente de conocimientos.

Luis.

El presente proyecto de investigación está dedicado a mis padres Patricio y Nelly, porque me brindaron el apoyo necesario para lograr cumplir mis metas, dedicaron horas de sacrificio para que nunca me faltara nada y han inculcado en mí los valores necesarios para ser una persona de bien.

A mis profesores quienes sembraron en mi la semilla de la curiosidad que inspiro esta investigación, por todo el conocimiento ofrecido en las aulas de clase y el apoyo incondicional hacia mi persona.

A mis amigos, por siempre estar ahí cuando más los necesite, haciendo de la vida universitaria algo increíble que no será borrado con el tiempo.

Fabricio

AGRADECIMIENTO

Te agradezco a ti mi Dios por ser quién ilumina mi mente, guía mis palabras y mis caminos.

Les agradezco a mis padres; Luis y Nely, por estar conmigo hasta en los momentos más difíciles y por siempre brindarme su amor y su tiempo. sus ideas y esperanzas.

Agradezco al Dr. Oscar Chang y a mi primo Darwin por su ayuda, tiempo y conocimientos.

De corazón también agradezco a todos los que me han acompañado en la senda que llamamos vida. Disculpas les pido si no les nombro pero han sido muchos los que conmigo han estado, están y estarán, no quiero omitir a ninguno ni desconocer sus méritos por lo que aunque en este documento no estén sepan que en mi mente y corazón siempre escritos como en mármol vivirán.

Luis

Agradezco a mis padres Patricio y Nelly, que con arduas horas de trabajo lograron brindarme la educación que ellos no pudieron tener, por los consejos de vida que supieron darme, por las ganas de estudiar que me inculcaron desde pequeño, por el apoyo sin esperar nada a cambio, por la fortaleza y determinación para sacar a nuestra familia adelante, por la experiencia para enfrentar al mundo y por ayudarme a construir mis sueños aún acosta de los suyos, gracias.

A mis profesores que desde pequeño me mostraron que en este mundo hay mucho por descubrir, por enseñarme que la fuerza más poderosa en del universo es la fuerza de voluntad, por el conocimiento incondicional brindado, por su paciencia y profesionalidad.

Fabricio

ÍNDICE DE CONTENIDO

CARÁTULA.....	I
CERTIFICACIÓN.....	ii
AUTORÍA DE RESPONSABILIDAD.....	iii
AUTORIZACIÓN.....	iv
DEDICATORIA.....	v
AGRADECIMIENTO.....	vi
ÍNDICE DE CONTENIDO.....	vii
ÍNDICE DE TABLAS.....	xii
ÍNDICE DE FIGURAS.....	xiv
RESUMEN.....	xix
ABSTRACT.....	xx
CAPÍTULO I	1
FUNDAMENTOS TEÓRICOS	
1.1 Introducción.....	1
1.2 Problema.....	1
1.3 Justificación.....	2
1.4 Objetivos.....	3
1.4.1 Objetivo general.....	3
1.4.2 Objetivos específicos.....	3
1.5 UAV.....	3
1.5.1 Definición.....	3
1.5.2 Historia.....	4
1.5.3 Clasificación de los UAV.....	6
1.6 Multicópteros.....	7

1.6.1	Tipos de Multicópteros.....	7
1.7.	Redes Neuronales Artificiales (RNA).....	9
1.7.1.	Definición.....	9
1.7.2.	Neurona.....	10
1.7.3.	Arquitectura de una Red Neuronal.....	10
1.7.4.	Características del procesamiento Neuronal.....	11
a)	Funciones de Activación.....	12
b)	Función de Entrada.....	12
c)	Función de Activación.....	13
d)	Función de Salida.....	15
e)	Tipos de Aprendizaje.....	15
1.8	Visión Artificial.....	16
1.8.1	Definición.....	16
1.8.2	Proceso de visión artificial.....	16
1.8.3	Componentes de un sistema de visión artificial.....	18
1.8.4	Procesamiento de imágenes.....	20
1.8.5	Detección de bordes.....	23
1.8.6	Flujo óptico.....	25
1.8.7	Lucas Kanade.....	25
1.9	Usos y aplicaciones.....	26

CAPÍTULO II **29**

EQUIPOS Y CONFIGURACIONES

2.1	Introducción.....	29
2.2	Elección del Multirotor.....	29
2.3	Funcionamiento.....	31

2.4	Estructura	32
2.5	Elección de la Emisora	34
2.5.1	Transmisor Turnigy 9X	36
2.5.2	Configuración	37
2.6	Receptor Turnigy 9X	43
2.7	Elección de la placa de control	44
2.7.1	Arduino Mega 2560	46
2.8	Elección del tipo de Relés	48
2.8.1	Módulo de relés OMROM SSR G3MB-202P	49
2.9	Elección del módulo GPS	51
2.9.1	Módulo GPS GY-GPS6MV2	53
2.10	Elección del sensor de altura	54
2.10.1	Sensor Ultrasónico HC- SR04	56
2.11	Elección del computador	57
2.11.1	PC Intel STCK1A32WFC	59
2.12	Elección de la cámara	60
2.12.1	Cámara SJ4000	62
2.13	Elección de la tarjeta controladora de vuelo	63
2.13.1	Controladora de vuelo Dji Naza M Lite	64
2.14	Elección de los variadores de velocidad	72
2.14.1	Variadores de velocidad Afro ESC 20 A	73
2.15	Elección de los motores	75
2.15.1	Motores Multistar Brushless Mt2213-935kv	76
2.16	Hélices 8x4.5 (CW/CCW)	78
2.17	Elección de la batería	79
2.17.1	Batería LIPO ZOP POWER 5000mah	81

CAPÍTULO III: 83

PROGRAMACIÓN E IMPLEMENTACIÓN

3.1	Introducción	83
3.2	Programación en Arduino	83
3.2.1	Señales de control PWM	83
3.2.2	Sensor ultrasónico HC-SR04	98
3.2.3	Módulo GPS GY-GPS6MV2	101
3.2.4	Comunicación Arduino – PC (Red Neuronal Artificial).....	103
3.2.5	Secuencia Autónoma.....	106
a)	Lectura de Altura.....	108
b)	Despegue	108
c)	Control de Altura.....	109
d)	Ruta GPS	110
e)	Comunicación con la Red Neuronal Artificial (PC)	110
f)	Aterrizaje	112
3.3	Programación de la red neuronal artificial.	113
3.3.1	Entrenador de la primera red neuronal.	114
3.3.2	Entrenador de la segunda red neuronal.	121
3.3.3	Desarrollo del programa buscador.	125

CAPÍTULO IV 129

ANÁLISIS DE RESULTADOS

4.1	Introducción.	129
4.2	Control ARDUINO	129
4.2.1	Señales PWM	129
4.2.2	Despegue	131

4.2.3	Control de Altura.....	137
4.2.4	Ruta	138
4.2.5	Comunicación - PC	140
4.3	Análisis del desarrollo de las redes neuronales artificiales.....	141
4.3.1	Análisis del desarrollo de la primera red neuronal.....	141
4.3.2	Análisis del desarrollo de la segunda red neuronal artificial.....	144
4.4	Análisis del desarrollo de los elementos de visión artificial.....	147
4.5	Análisis de la RNA y la cámara en el aire.....	148
4.6	Aterrizaje.....	151

CAPITULO V **154**

CONCLUSIONES Y RECOMENDACIONES

5.1	Conclusiones	154
5.2	Recomendaciones.....	158

BIBLIOGRAFÍA..... 161

ANEXOS.....169

ÍNDICE DE TABLAS

Tabla 1. Tipos de Multicópteros.	8
Tabla 2. Funciones de activación más en Redes Neuronales Artificiales.	13
Tabla 3. Multirrotores Dji Inspire Drone, Dji Phantom 3, Qanum Venture Drone.	30
Tabla 4. Matriz QFD para la elección del multirrotor.	31
Tabla 5. Características adicionales del dron Qanum Venture.	34
Tabla 6. Principales características de las emisoras Turnigy 9X y FlySky TH 9X.	34
Tabla 7. Matriz QFD para la elección de la emisora de vuelo.	35
Tabla 8. Características de las placas Arduino NANO, MEGA 2560 y UNO.	44
Tabla 9. Matriz QFD para la elección de la tarjeta de control.	45
Tabla 10. Distribución de canales de control de la aeronave Qanum Venture Dron.	46
Tabla 11. Características Arduino Mega 2560.	47
Tabla 12. Módulos de relés SONGLE SRD-05VD y OMROM SSR 63MB-202P.	48
Tabla 13. Matriz QFD para la elección del módulo de relés.	49
Tabla 14. Características técnicas de los relés de estado sólido.	50
Tabla 15. Módulos GPS GP3906-TLP y GY-GPS6MV2.	51
Tabla 16. Matriz QFD para la elección del módulo GPS.	52
Tabla 17. Características técnicas módulo GPS GY-GPS6MV2.	53
Tabla 18. Características técnicas de los sensores MB103, HCSR04 y BMP180.	54
Tabla 19. Matriz QFD para la elección del sensor de altura.	55
Tabla 20. Características Técnicas Sensor HC – SR04.	56
Tabla 21. Características técnicas de la Intel STCK1A32WFC y la Raspberry Pi 3.	57
Tabla 22. Matriz QFD para la elección del computador.	58
Tabla 23. Características técnicas PC Intel STCK1A32WFC.	59
Tabla 24. Características técnicas de las cámaras GoPro Hero 3 y SJCAM 4000.	61
Tabla 25. Matriz QFD para la elección de la cámara.	61
Tabla 26. Controladoras de vuelo NAZA M Lite y APM 2.8.	63
Tabla 27. Matriz QFD para la elección de la tarjeta controladora de vuelo.	64
Tabla 28. Características técnicas de la controladora de vuelo Dji Naza M Lite.	67
Tabla 29. Variadores de velocidad UBEC ESC 20 A y AFRO ESC 20 A.	72

Tabla 30. Matriz QFD para la elección de los variadores de velocidad.	73
Tabla 31. Características técnicas de los variadores de velocidad Afro ESC 20 ^a	74
Tabla 32. Motores Multistar Mt2213-935kv y Turnigy D2836/9-950kv.	75
Tabla 33. Matriz QFD para la elección de los motores.	76
Tabla 34. Características técnicas de los motores Brushless MT2213-935kv.....	77
Tabla 35. Características técnicas de las hélices 8x4.5.	78
Tabla 36. Análisis de consumo de energía del proyecto.....	79
Tabla 37. Baterías ZOP POWER, MULTISTAR y TCB.	80
Tabla 38. Matriz QFD para la elección de la batería LIPO.	81
Tabla 39. Características técnicas de la batería LIPO ZOP Power 5000mah.	82
Tabla 40. Cálculo de promedios del sensor ultrasónico para medir la altura del drone.	99
Tabla 41. Códigos de posicionamiento neuronal PC – Arduino, canales Alerón y Elevador.....	104
Tabla 42. Relación entre las neuronas de salida y la posición del centro de la ROI.	117
Tabla 43. Relación entre las neuronas de salida y la posición del centro de la ROI.	124
Tabla 44. Frecuencias establecidas de fábrica para la tarjeta arduino MEGA 2560.....	130
Tabla 45. Pruebas de respuesta del sensor sobre una superficie de cemento.....	132
Tabla 46. Pruebas de respuesta del sensor sobre una superficie de asfalto.....	133
Tabla 47. Pruebas de respuesta del sensor sobre una superficie de arenilla.	134
Tabla 48. Pruebas de respuesta del sensor sobre una superficie de tierra.	134
Tabla 49. Pruebas de respuesta del sensor sobre una superficie de césped.	135
Tabla 50. Análisis de superficies de vuelo.....	137
Tabla 51. Pruebas de ruta GPS realizadas en las ciudades de Ambato y Shell-Puyo.	139
Tabla 52. Posiciones del helipad al centro de la ROI vs posición detectada por la RNA.....	144
Tabla 53. Análisis de respuesta de la salida de la segunda RNA.....	147
Tabla 54. Control del threshold.....	148
Tabla 55. Pruebas de TRACKING en diversos terrenos.....	150
Tabla 56. Desviación de aterrizaje en pruebas de vuelo.	151

ÍNDICE DE FIGURAS

Figura 1. Vehículo Aéreo No Tripulado (UAV).....	4
Figura 2. Primer UAV en el mundo 1916.....	4
Figura 3. UAV Firebee 1970.....	5
Figura 4. Predator UAV 1991.....	5
Figura 5. Proyecto ERAST UAV de la NASA 1991.....	6
Figura 6. Clasificación y configuración de los UAVs.....	7
Figura 7. Neurona.....	10
Figura 8. Estructura de una Red Neuronal.....	10
Figura 9. Funciones de Activación en una Red Neuronal.....	12
Figura 10. Proceso de visión artificial.....	17
Figura 11. Cámara y sus pates.....	19
Figura 12. Luz reflejada por el sol.....	19
Figura 13. Formatos de imágenes digitales.....	21
Figura 14. Resultado de aplicar el detector de bordes de Canny.....	24
Figura 15. Puntos característicos y solución Algoritmos.....	26
Figura 16. Elección del multirrotor, Dji Inspire Drone, Dji Phantom 3, Qunam Venture Drone.....	30
Figura 17. Disposición de hélices en un cuadricóptero.....	32
Figura 18. Ejes de giro presentes en un cuadricóptero.....	32
Figura 19. Qunam Venture Drone. a) Estructura Básica. b) Estructura Avanzada.....	33
Figura 20. Transmisor Turnigy 9X.....	36
Figura 21. Distribución de botones y palancas en el transmisor Turnigy 9X.....	37
Figura 22. Modo “SYSTEM” en transmisor Turnigy 9X.....	38
Figura 23. Categoría selección de modo en el transmisor Turnigy 9X.....	38
Figura 24. Memoria del sistema de vuelo en el transmisor Turnigy 9X.....	38
Figura 25. Nombre del usuario en el sistema de vuelo en el transmisor Turnigy 9X.....	39
Figura 26. Tipo de aeronave para sistema de vuelo en el transmisor Turnigy 9X.....	39
Figura 27. Tipo de modulación del receptor en el transmisor Turnigy 9X.....	40

Figura 28. Modo de distribución de los Sticks para el sistema de vuelo en el transmisor.	40
Figura 29. Categoría “SYSTEM” del modo de vuelo en el transmisor Turnigy 9X.	41
Figura 30. Canales auxiliares del modo de vuelo en el transmisor Turnigy 9X.	41
Figura 31. Configuración de canales 5,6,7,8,9 en el transmisor Turnigy 9X.	41
Figura 32. Programación de la mezcla de canales del modo de vuelo en el transmisor.	42
Figura 33. Programación del MIX1 del modo de vuelo en el transmisor Turnigy 9X.	42
Figura 34. Programación del MIX 2 del modo de vuelo en el transmisor Turnigy 9X.	43
Figura 35. Programación del MIX 3 del modo de vuelo en el transmisor Turnigy 9X.	43
Figura 36. Diagrama.....	44
Figura 37. Arduino Mega 2560.....	46
Figura 38. Módulo de relés OMROM SSR G3MB-202P.....	50
Figura 39. Conexiones a través de los relés de estado sólido.	51
Figura 40. Módulo GPS GY-GPS6MV2.	53
Figura 41. Conexiones del Módulo GPS GY-GPS6MV2.....	54
Figura 42. Sensor Ultrasónico HC-SR04.....	56
Figura 43. Conexiones realizadas para el Sensor HC – SR04.	57
Figura 44. PC Intel STCK1A32WFC	59
Figura 45. Conexiones PC Intel STCK1A32WFC.	60
Figura 46. Cámara SJ4000.	62
Figura 47. Características técnicas de la cámara DJ4000.	62
Figura 48. Tarjeta estabilizadora de vuelo Dji Naza M Lite.....	65
Figura 49. Módulo Dji GPS e Indicador Led.....	65
Figura 50. Conexiones de la tarjeta estabilizadora de vuelo Dji Naza M Lite.....	66
Figura 51. Selección del tipo de multirrotor la tarjeta estabilizadora de vuelo Dji Naza.	68
Figura 52. Posición del GPS con respecto a la tarjeta estabilizadora de vuelo Dji Naza.	68
Figura 53. Elección del tipo de receptor usado para la tarjeta de vuelo Dji Naza.	69
Figura 54. Calibración de Sticks en la tarjeta estabilizadora de vuelo Dji Naza M.....	69
Figura 55. Modo de vuelo ATTI en la tarjeta estabilizadora de vuelo Dji Naza M.....	70

Figura 56. Modo de vuelo Manual en la tarjeta estabilizadora de vuelo Dji Naza M.	71
Figura 57. Modo de vuelo GPS en la tarjeta estabilizadora de vuelo Dji Naza M.	71
Figura 58. Variadores de velocidad Afro ESC 20A.....	74
Figura 59. Motores Brushless MT2213-935kv.	77
Figura 60. Hélices 8x4.5 y su sentido de giro en el multirrotor.	78
Figura 61. Batería LIPO ZOP POWER 5000mah.	82
Figura 62. Ejes y movimientos de un multirrotor.....	84
Figura 63. Señal PWM Canal # 1 / Alerón.	84
Figura 64. Posición mínima del Stick Alerón b) Señal PWM Canal # 1 / Alerón nivel mínimo (Stick 100% izquierda).	85
Figura 65. a) Posición mínima del Stick Alerón b) Señal PWM Canal # 1 /Alerón nivel medio (Stick 50%).....	86
Figura 66. a) Posición media del Stick Alerón b) Señal PWM Canal # 1 / Alerón nivel máximo (Stick 100% derecha).	87
Figura 67. a) Posición máxima del Stick Alerón b) Señal PWM Canal # 2 / Elevador...88	
Figura 68. a) Posición mínima del Stick Elevador b) Señal PWM Canal # 2 / Elevador nivel mínimo (100% Abajo).....	89
Figura 69. a) Posición medio del Stick Elevador b) Señal PWM Canal # 2 / Elevador nivel medio (50 %).	90
Figura 70. a) Posición máxima del Stick Elevador b) Señal PWM Canal # 2 / Elevador nivel máximo (100% arriba).	91
Figura 71. Señal PWM Canal # 3 / Acelerador.....	92
Figura 72. a) Posición mínima del Stick Acelerador b) Señal PWM Canal # 3 / Acelerador nivel mínimo (100% Abajo).	92
Figura 73. a) Posición media del Stick Acelerador b) Señal PWM Canal # 3 / Acelerador nivel medio (50%).	93
Figura 74. a) Posición máxima del Stick Acelerador b) Señal PWM Canal # 3 / Acelerador nivel máximo (100% Arriba).	94
Figura 75. Diagrama de flujo programación de señales PWM.	95
Figura 76. Diagrama de flujo de la programación para el control de los módulos de relé para el cambio entre el codo manual y el autónomo.	96

Figura 77. a) Conexiones entre el módulo de relés, la tarjeta arduino y la controladora de vuelo b) Implementación del sistema de relés.	97
Figura 78. Diagrama de flujo para la lectura del sensor ultrasónico HC-SR04.....	98
Figura 79. Grafica de acondicionamiento de datos del sensor ultrasónico HC-SR04 ...	100
Figura 80. a) Conexiones del sensor ultrasónico HC – SR04 b) Implementación del sensor ultrasónico HC-SR04 en el multirrotor.	101
Figura 81. Diagrama de flujo de la programación para obtener los datos del módulo GPS GY-GPS6MV2.	102
Figura 82. a) Conexiones del módulo GPS-GY6MV2 b) Implementación del módulo GPS GY-GPS6MV2 en el multirrotor.....	103
Figura 83. Programación para la comunicación Arduino – PC	105
Figura 84. Trama serial de la comunicación arduino – red neuronal artificial.	106
Figura 85. Secuencia autónoma.	107
Figura 86. Diagrama de flujo de la secuencia de despegue autónoma.....	108
Figura 87. Diagrama de flujo para el control de altura del multirrotor.	109
Figura 88. Ruta GPS.	111
Figura 89. Diagrama de flujo de la secuencia autónoma de aterrizaje.....	112
Figura 90. Adquisición de imágenes usando openCV A) adquisición en tiempo real. B) adquisición desde una base de datos.	114
Figura 91. Diagrama de flujo, secuencia de binarización de imagen.....	115
Figura 92. En la sub ventana de color verde se observa la imagen capturada y girada (200x200 pixeles) mientras que en la ventana de color blanco se visualiza la imagen ROI (100x100 pixeles capturada de la imagen de 200 x 200).....	116
Figura 93. Entrenamiento de la red salidas vs posición de la imagen (targets) A) muestra la respuesta de las neuronas sin entrenar (listones rojos) y en B) muestra a la red entrenada (Listones rojos).....	118
Figura 94. Diagrama de flujo, secuencia de entrenamiento de la primera red neuronal artificial.	120
Figura 95. Entrador de la primera red.	121
Figura 96. Diagrama de flujo de entrenamiento segunda red neuronal artificial.....	122

Figura 97. Proceso de alimentación segunda red neuronal artificial.	123
Figura 98. Control de CANNY y mejoramiento de detección de características, A) paso de todos los bordes y B) filtrado de los bordes menos contrastantes.	124
Figura 99. Entrador de la segunda red neuronal artificial.	125
Figura 100. Detección entre helipad y ruido A existe helipad B no existe.	126
Figura 101. Flujo grama del programa buscador.	127
Figura 102. Lucas Kanade.	128
Figura 103. Modo Emergencia NAZA M Lite.	131
Figura 104. Centro de gravedad del multirotor.	131
Figura 105. Superficie de pruebas.	136
Figura 106. Ruta del multirotor afectada por el factor del viento.	140
Figura 107. Ruta del multirotor completada.	140
Figura 108. Resultados del entrenamiento de la primera red neuronal artificial.	143
Figura 109. Respuesta de la segunda RNA A) cuando existe un helipad y B) cuando no existe.	146
Figura 110. Variación del threshold.	148
Figura 111. Variación del threshold desde el aire A) valor normalmente usado en las pruebas B) valor adecuado al terreno.	149
Figura 112. Tipos de terreno A) césped B) gravilla C) Tierra D) asfalto.	150
Figura 113. Aterrizaje afectado por el viento.	152
Figura 114. Aterrizaje cerca del objetivo helipad.	153

RESUMEN

Uno de los principales problemas al momento de pilotar una aeronave es el aterrizaje, ya que requiere una concentración extrema y ejecución precisa por parte del piloto, en la vida moderna el volar es la manera más segura de transporte pero también es indudable que en el momento de originarse un accidente, que normalmente son ocasionados por fallas humanas, se obtiene mayores daños que los sucedidos en otros medios de transporte, con el objetivo de resolver este problema, se desarrolla el presente trabajo de investigación de un sistema de aterrizaje autónomo e innovador sobre un cuadricóptero, donde un sistema de despegue controlado lleva al multirrotor a una altura deseada gracias a un controlador, una ruta GPS donde se le informa a la aeronave la posición de la zona de aterrizaje, para que se dirija a la misma, un sistema de posicionamiento neuronal, en un computador abordo, mediante el proceso de identificación de imágenes gracias al conocimiento implantado en un sistema de redes neuronales artificiales en cascada que permiten localizar la zona de aterrizaje, al ubicar el objetivo helipad para el posterior posicionamiento del multirrotor sobre el mismo y finalmente un proceso de aterrizaje autónomo y seguro en tierra.

Palabras clave:

- **TARJETA ARDUINO**
- **REDES NEURONALES ARTIFICIALES**
- **VISIÓN ARTIFICIAL**

ABSTRACT

One of the main problems when flying an aircraft is the landing, since it requires an extreme concentration and precise execution by the pilot, it is true that in modern life flying is the safest way to transport, but it is also true that At the moment of an accident, which are normally caused by human faults, greater damages are obtained than those occurred in other means of transportation, in order to solve this problem the present research work was developed on an autonomous landing system and Innovative approach on a quadcopter, where a controlled take-off system takes the multicopter to a desired height thanks to a controller, a GPS route where the aircraft is informed of the position of the landing zone, so that it is directed to it System of neuronal positioning, in an onboard computer, through the process of identification of images thanks to the knowledge implanted in a system of neuronal networks in cascade that allow to locate the zone of landing, when locating the objective helipad for the later positioning of the multicopter on the And finally an autonomous and safe land landing process.

Keywords:

- **ARDUINO CARD**
- **ARTIFICIAL NEURAL NETWORKS**
- **ARTIFICIAL VISION**

CAPÍTULO I

FUNDAMENTOS TEÓRICOS

1.1 Introducción

En el primer capítulo abarca los conceptos fundamentales mediante los cuales se realiza una investigación sobre el método óptimo de aterrizaje autónomo de un dron, mediante el uso del aprendizaje por redes neuronales artificiales y visión artificial, así también se efectuará una investigación sobre los avances logrados hasta el día de hoy y cómo contribuir en el adelanto en este campo.

El proyecto se desarrolla en un vehículo aéreo no tripulado (UAV) de cuatro rotores o cuadricóptero sobre el cual se explica su definición, historia y clasificación existente en el mercado, gracias a estos modelos a escala de aeronaves se puede realizar análisis e investigaciones que permite observar el comportamiento de sus similares en la vida real, de igual manera se expone lo referente a las redes neuronales que permitan en colaboración con la visión artificial el desarrollo del presente proyecto.

1.2 Problema

“Así como el despegue, y el aterrizaje de una aeronave presupone del piloto concentración extrema y ejecución precisa de las técnicas correspondientes” (Zeidan, 1990), es verdad que en la vida moderna el volar es la manera más segura de transporte pero también es cierto que en el momento de originarse un accidente, que normalmente son ocasionados por fallas humanas, se obtiene mayores daños que los sucedidos en otros medios de transporte, es por ello que se realiza el presente proyecto de investigación en busca del método óptimo de visión artificial y redes neuronales para la implementación del sistema de aterrizaje de un cuadricóptero de forma autónoma.

El trabajar con visión artificial resulta muy complejo, “ya que el principal inconveniente para la adquisición de imágenes aéreas es la necesidad de contar con amplias zonas de despegue y aterrizaje” (Angulo Valencia, 2014), pero el uso de redes neuronales artificiales (RNA) como solución para el control de un procedimiento como este, resulta muy innovador.

1.3 Justificación

El desarrollo del presente proyecto se efectúa sobre un vehículo aéreo no tripulado (UAV) de cuatro rotores o cuadricóptero y tiene como objetivo realizar su aterrizaje autónomo y seguro sobre una plataforma para helicópteros, gracias al aprendizaje de la imagen (helipad) por medio de un computador y haciendo uso de redes neuronales artificiales (RNA) y visión artificial; el control de un procedimiento como este resulta muy innovador ya que “tanto el despegue como el aterrizaje de una aeronave requiere del piloto una concentración extrema y ejecución precisa de la técnicas correspondientes” (Zeidan, 1990), por lo cual no es labor fácil la de ejecutar el aterrizaje de una aeronave, una vez entrenada la RNA se conseguirá darle al cuadricóptero la capacidad de reconocer una gran cantidad de referencias de aterrizaje (helipad), pudiendo así inferir conocimiento sobre situaciones que nunca se le enseñaron, gracias a lo cual se podrá eventualmente aterrizar drones de forma segura tanto en aplicaciones cotidianas como en situaciones especiales, desastres naturales, supervisión de sembradíos, revisión de líneas eléctricas de alta tensión y entre muchas más. Solamente se necesita dibujar un helipad en el lugar de aterrizaje y el dron realiza su trabajo, este sistema podría también ser utilizado en vehículos aéreos de presentarse alguna eventualidad con los pilotos, permitiendo que en esta emergencia el sistema tome el control y aterrice la aeronave en forma segura.

Se investiga sobre la combinación de RNA en diferentes arreglos que permitan reducir el error que comete la red, así como la capacidad de inferencia ante imágenes que nunca ha visto, verificar su comportamiento en relación con la eficiencia en el control de aterrizaje, hasta encontrar el arreglo óptimo que ayude a llegar al objetivo planteado, así también se abre las puertas a investigaciones futuras tanto sobre las redes neuronales artificiales como la visión artificial dentro de la carrera.

1.4 Objetivos

1.4.1 Objetivo general

Investigar el conjunto óptimo de técnicas en visión artificial y redes neuronales que permitan la implementación del sistema de aterrizaje en un cuadricóptero.

1.4.2 Objetivos específicos

- Entrenar las redes neuronales artificiales con una base de datos para identificar el objetivo (helipad) o punto de aterrizaje.
- Identificar los parámetros del vehículo aéreo no tripulado (UAV) necesarios para controlar el aterrizaje
- Insertar las variables-controladas dentro del sistema de redes neuronales artificiales para realizar el aterrizaje.
- Acoplar el sistema de control al vehículo aéreo no tripulado (UAV) para realizar las pruebas de su funcionamiento.
- Calibrar y corregir posibles eventualidades para optimizar el proyecto.
- Analizar los resultados obtenidos.

1.5 UAV

1.5.1 Definición

Un avión sin tripulante se define utilizando los términos UAV (unmanned aerial vehicle o vehículo aéreo no tripulado), “el mismo es controlado por la navegación autónoma o la navegación por control remoto, basado en la sustentación dinámica y empuje aerodinámico” (Nonami, Kendoul, Suzuki, Wang, & Nakazawa, 2010), como se muestra en la figura 1.



Figura 1. Vehículo Aéreo No Tripulado (UAV).

Fuente: (*HobbyKing, 2015*)

1.5.2 Historia

El primer UAV fue diseñado por los americanos Lawrence y Sperry en 1916, quienes añadieron un giroscopio para su estabilización con el objetivo de implementar un piloto autónomo. Esto se conoce como “el principio de control de altitud, que llega a ser utilizado para la dirección automática de una aeronave” (Nonami, Kendoul, Suzuki, Wang, & Nakazawa, 2010), como se muestra en la figura 2 fue nombrado “torpedo de aviación” el cual alcanzaba una distancia superior a 48 kilómetros.



Figura 2. Primer UAV en el mundo 1916.

Fuente: (Nonami, Kendoul, Suzuki, Wang, & Nakazawa, 2010)

Debido al poco desarrollo, los UAV no fueron utilizados en las dos primeras guerras mundiales, su verdadero progreso comenzó en el año 1950, en los conflictos de Vietnam

y la guerra fría, prolongándose así hasta el año de 1970 con el UAV llamado “Firebee” el cual se muestra en la figura 3.



Figura 3. UAV Firebee 1970.

Fuente: (Nonami, Kendoul, Suzuki, Wang, & Nakazawa, 2010)

A medida que avanza el tiempo, los UAV han mejorado sus características como por ejemplo el ser más compactos, con mejores prestaciones, sensores avanzados y cámaras de video, tanto en el ambiente militar como en el civil, igualmente como el dron militar usado en 1991 por los Estados Unidos en la guerra del golfo, llamado “Predator” que se muestra en la figura 4, o el dron de la NASA de 1990 que en el ámbito civil realizó un proyecto de drones llamado “ERAST (Aviones de Investigación y Sensores de Tecnología Ambiental)” que se muestra en la figura 5.



Figura 4. Predator UAV 1991.

Fuente: (Nonami, Kendoul, Suzuki, Wang, & Nakazawa, 2010)



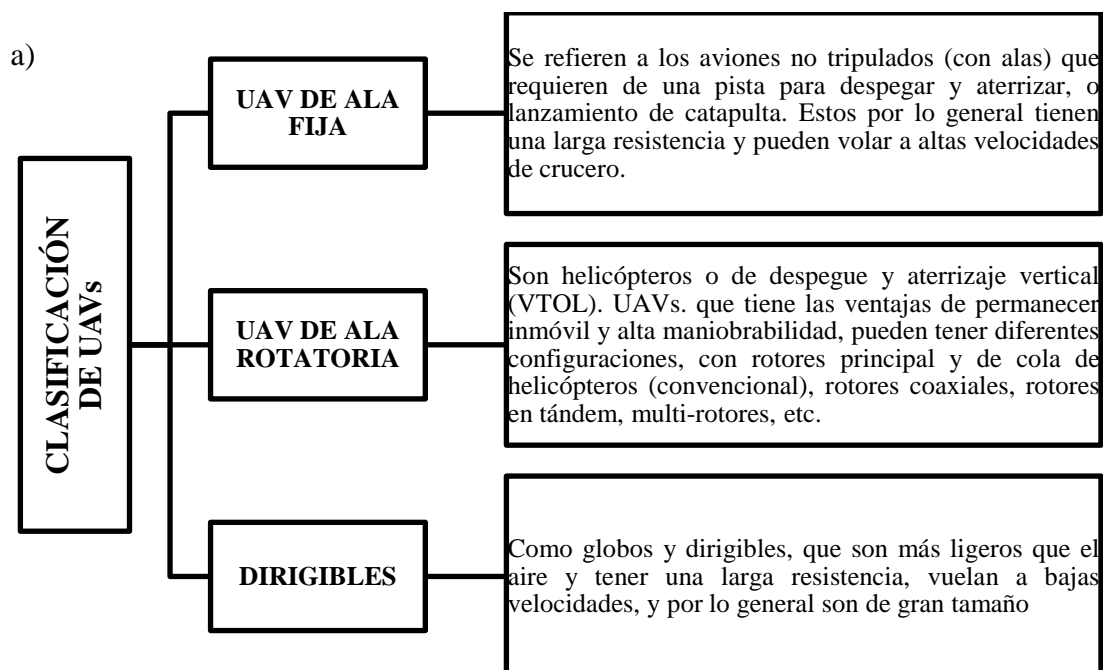
Figura 5. Proyecto ERAST UAV de la NASA 1991.

Fuente: (Nonami, Kendoul, Suzuki, Wang, & Nakazawa, 2010)

Hoy en la actualidad hay una demanda muy grande por el uso cada vez más común de los UAV y la facilidad de países como China que son los principales desarrolladores de estos dispositivos, lo cual ha hecho que cada vez se vuelvan más económicos y de fácil adquisición.

1.5.3 Clasificación de los UAV

En los últimos años se han desarrollado un sin número de modelos y diferentes tipos de vehículos aéreos no tripulados, poseen diferentes tamaños, niveles más exigentes de robustez y prestaciones de vuelo, su clasificación y configuración se representa a continuación: Ver Figura 6.



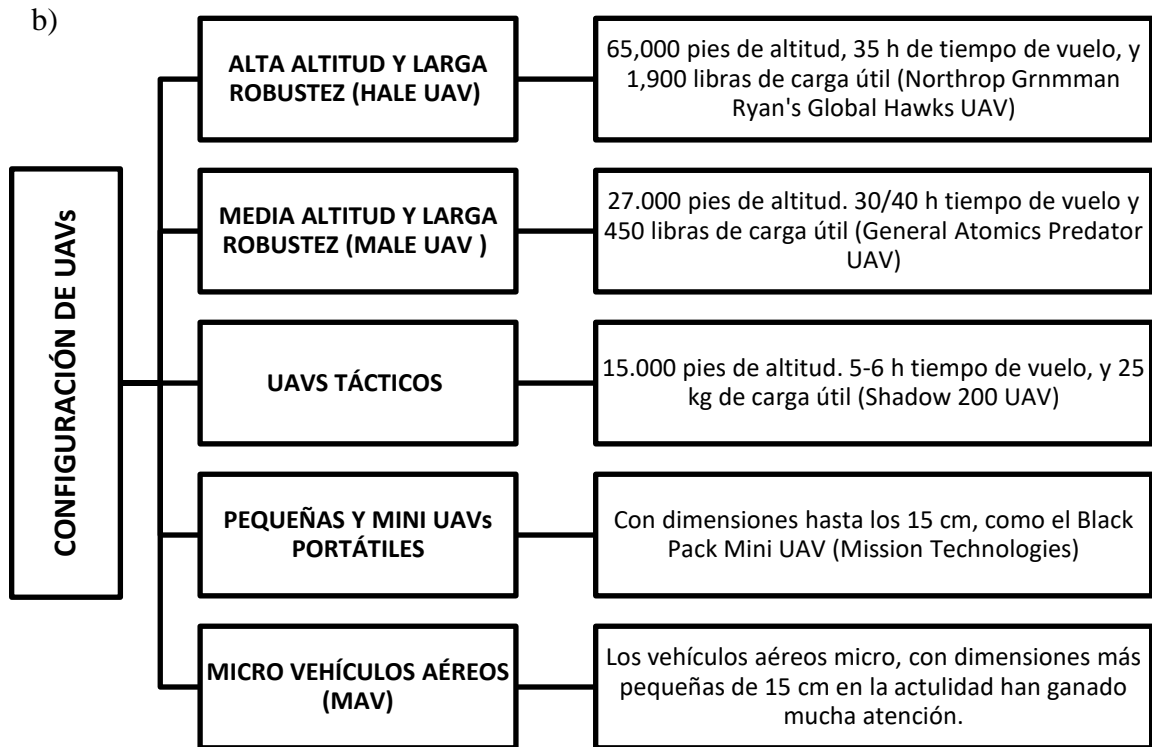


Figura 6. a) Clasificación de los UAV. b) configuración de los UAV.

Fuente: (Nonami, Kendoul, Suzuki, Wang, & Nakazawa, 2010)

1.6 Multicópteros


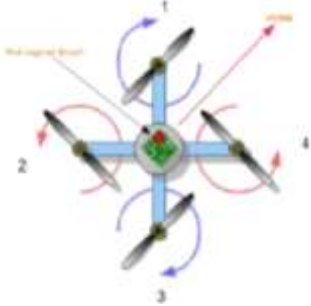
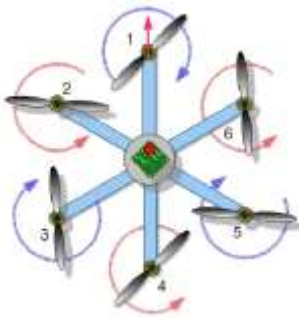
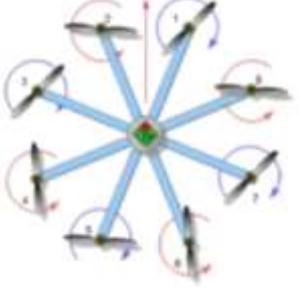
“Un dron es considerado multicóptero si posee tres o más hélices dispuestas en un plano y sin poseer un rotor de cola” (López, 2012), esta clase de drones en la actualidad son los más utilizados en el ámbito civil, debido a que pueden brindar ventajas tales como la estabilidad de vuelo, control de velocidad de cada motor y sus mecanismos son relativamente económicos.

1.6.1 Tipos de Multicópteros

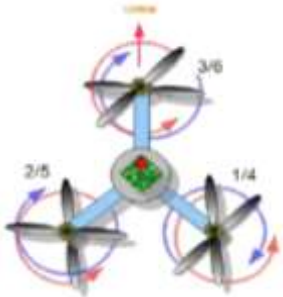
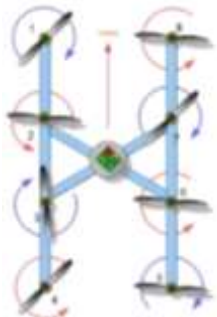
Existen diversas configuraciones de multicópteros, según la disposición de sus hélices, se establece un resumen de los principales, los cuales se muestran en la tabla 1.

Tabla 1.

Tipos de Multicópteros.

Tipo de	Descripción	Ilustración
Multicópteros		
Tricóptero	Económico y de fácil construcción, estabilidad reducida, mecanismos móviles en la estructura (cola móvil por un servo), empuje y tiempo de vuelo reducido (debido a que los motores deben girar a mayor velocidad para mantenerlo en el aire).	
Cuadricóptero	Con un diseño más simple que el tricóptero. Dispone de 1/3 más de empuje, con un peso similar y posee mayor estabilidad al no disponer de partes móviles. Mayor tiempo de vuelo ya que pueden llevar baterías más grandes y a que los motores trabajan a menos revoluciones.	
Hexacóptero	Posee las ventajas de un cuadricóptero, pero con más potencia y más capacidad de carga. Si pierde un motor todavía puede aterrizar, (perdemos el control del ángulo 'YAW'). Son más caros y más grandes.	
Octocóptero	Dispone de las ventajas de un Hexacóptero y además mayor redundancia. Al fallar un motor su estabilidad en vuelo no se ve afectada. Se usan para llevar equipos fotográficos de alto precio. Económicamente son más costosos y necesitan de mucha energía para volar.	

CONTINÚA 

Y6	Se define como Y6 a un tricóptero coaxial, es decir 2 motores en cada brazo dispuestos en su mismo eje, rotando de forma inversa. Trabaja igual que un tricóptero, con cola móvil. Pero posee mayor capacidad de carga y redundancia.	
V8	Dispone de la ventaja panorámica para la fotografía aérea y su redundancia. Fue diseñado para la fotografía aérea y es un modelo patentado por una empresa alemana, “Ascending Technologies” la cual comercializa este octocóptero.	

Fuente: (López, 2012)

1.7. Redes Neuronales Artificiales (RNA)

Una de las ramas más interesantes del campo científico de la “Inteligencia Artificial” es la que corresponde a las “Redes Neuronales Artificiales (RNAs)”, “entendiendo como tales aquellas redes en las que existen elementos procesadores de información de cuyas interacciones locales depende el comportamiento del conjunto del sistema” (Zeidan, 1990).

1.7.1. Definición

Una red neuronal artificial es definida por (Flórez & Fernández, 2008) como “sistemas que simulan el comportamiento del cerebro humano, representado por el aprendizaje a través de la experiencia y la extracción de conocimiento genérico desde un conjunto de datos, estos sistemas imitan esquemáticamente la estructura neuronal del cerebro, gracias a un programa de ordenador (simulación), un modelado a través de estructuras de procesamiento con cierta capacidad de cálculo paralelo (emulación), o la construcción física de sistemas cuya arquitectura se aproxima a la estructura de la red neuronal biológica”.

1.7.2. Neurona

El componente básico de una red neuronal biológica es una neurona o mecanismo de procesamiento y a esta se la define como “un dispositivo muy simple que transforma en su soma, o cuerpo celular varias señales de entrada (dendritas) en una única salida (axón)” (Pino, Gómez, & de Abajo, 2001), como se ilustra en la figura 7. Las entradas pueden provenir de otras neuronas o ser entradas a la red desde fuera del sistema, de igual forma las salidas pueden transferir información a otras neuronas.

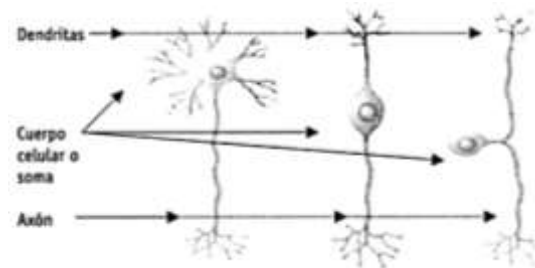


Figura 7. Neurona.

Fuente: (Flórez & Fernández, 2008)

1.7.3. Arquitectura de una Red Neuronal artificial

Para describir una red neuronal artificial no es suficiente con establecer el comportamiento individual de sus elementos (neuronas), hay que especificar también “las conexiones entre ellas, se agrupan en capas (figura 8), cada neurona se relaciona con un conjunto de neuronas de número variable y comportamiento similar, constituyendo varias capas de una red neuronal artificial” (Flórez & Fernández, 2008).

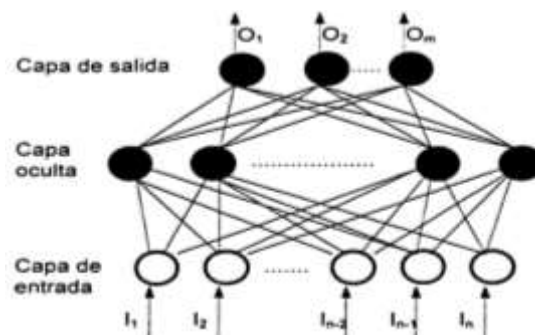


Figura 8. Estructura de una Red Neuronal.

Fuente: (Pino, Gómez, & de Abajo, 2001)

Cada capa está conectada a la anterior o siguiente, excepto la última, que compone la salida total de la red neuronal, según (Pino, Gómez, & de Abajo, 2001) existen tres tipos de capas:

- **Capa de entrada:** La cantidad y el tipo de neuronas que organizan esta capa, depende de los datos del problema.
- **Capas Intermedias:** Debido al tipo y complejidad del problema a resolver con la red, pueden existir más de una capa intermedia, gracias a la organización apropiada de estas capas se obtienen las características de generalización, adaptabilidad, etc. que hacen muy innovador el trabajar con redes neuronales.
- **Capa de salida:** La cantidad de neuronas de esta capa obedece al diseño de salida de la red.

1.7.4. Características del procesamiento Neuronal

Las redes neuronales artificiales disponen de algunas características especiales que las diferencian de los otros métodos de inteligencia artificial, gracias a ello han hecho de ellas un sistema de rápido aprendizaje y gran capacidad de inferencia, las principales características son descritas por (Pino, Gómez, & de Abajo, 2001) son:

- **Paralelismo masivo:** Brinda una gran capacidad de procesamiento, una red neuronal artificial se distribuye en elementos individuales, cada una se dispone a calcular una función elemental sin necesidad de requerir ayuda de las demás, lo cual es una enorme ventaja frente al procesamiento secuencial, donde cada elemento de computo puede proceder una vez que el anterior haya terminado con su trabajo.
- **Memoria asociativa distribuida:** La información no se encuentra en un solo punto de la estructura sino repartida a lo largo de la misma.
- **Tolerancia a fallos:** Si falla un elemento, la estructura no se ve afectada de gran manera, el sistema continuo con una función de error continua, sin rupturas totales

- **Tratamiento simultáneo** de grandes cantidades de información
- **Aprendizaje:** La estructura se va modificando hasta adaptarse a los datos proporcionados, adquiriendo así un conocimiento artificial.

a) Funciones de Activación

Las funciones de activación “otorgan los cálculos matemáticos necesarios para modificar el valor del que disponen los pesos (Memoria) en las diferentes neuronas para corregir errores y acercarnos al valor deseado (Aprender), se identifican las siguientes funciones de activación” (Matich, 2001).

b) Función de Entrada

La neurona procesa muchos valores de entrada como si fueran si tratase de uno solo; proporcionando una entrada global de datos ($in_{i1}, in_{i2}...$) como se muestra en la figura 9. La combinación de estos datos se logra a través de la función de entrada, como se muestra a continuación y se calcula a partir de un vector de entrada.

$$input_i = (in_{i1} \cdot w_{i1}) * (in_{i2} \cdot w_{i2}) * \dots * (in_{in} \cdot w_{in})$$

Dónde: * representa al operador apropiado (por ejemplo: máximo, sumatoria de productos, etc.), n al número de entradas a la neurona N_i y W_i el peso.

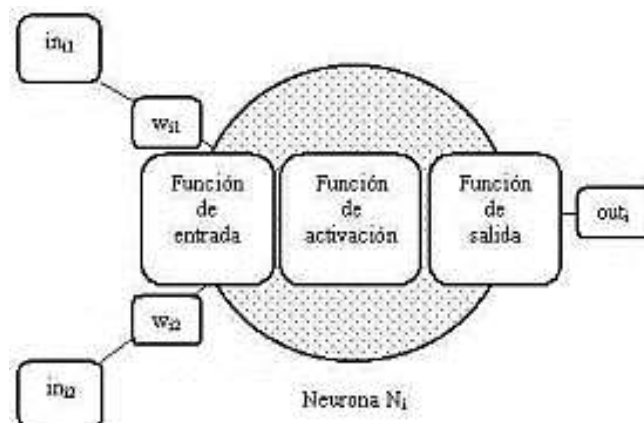


Figura 9. Funciones de Activación en una Red Neuronal.

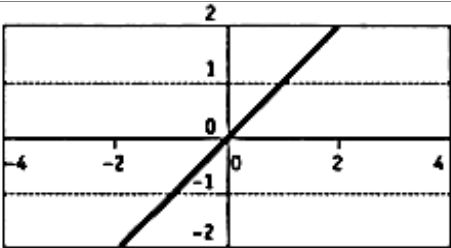
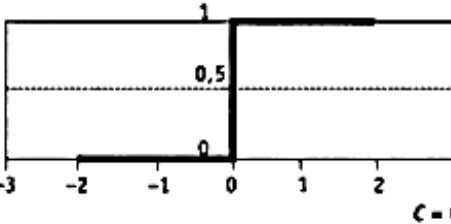
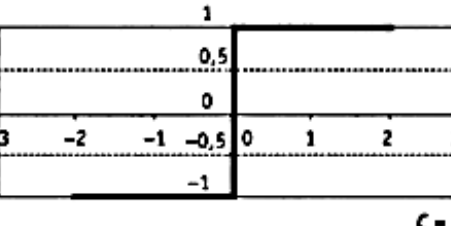
Fuente: (Matich, 2001)

c) Función de Activación

Una neurona biológica puede estar activa (excitada) o inactiva (no excitada); es decir, que tiene un “estado de activación”. Las neuronas artificiales también tienen diferentes estados de activación; algunas de ellas solamente dos, al igual que las biológicas, pero otras pueden tomar cualquier valor dentro de un conjunto determinado. La función de activación calcula el estado de actividad de una neurona; transformando la entrada global (menos el umbral, θ_i) en un valor (estado) de activación, cuyo rango normalmente va de (0 a 1) o de (-1 a 1). Esto es así, porque una neurona puede estar totalmente inactiva (0 o -1) o activa (1). La función de activación es una función de la entrada global (g_{in_i}) menos el umbral (θ_i). Las funciones de activación más comunes se detallan en la tabla 2.

Tabla 2.

Funciones de activación más en Redes Neuronales Artificiales.

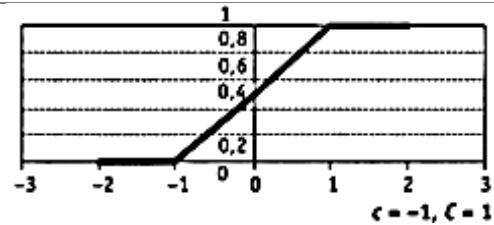
<p>Función de transferencia Lineal o</p>	
<p>Identidad. $a_j = f(Net_j(t)) = Net_j$ Rango = $(-\infty, +\infty)$</p>	
<p>Función de transferencia Escalón.</p>	
<p>$a_j = f(Net_j) = \begin{cases} 1 & \text{si } Net_j \geq c \\ 0 & \text{si } Net_j < c \end{cases}$ Rango = $\{0, 1\}$</p>	
<p>Función de transferencia Escalón.</p>	
<p>$a_j = f(Net_j) = \begin{cases} 1 & \text{si } Net_j \geq c \\ -1 & \text{si } Net_j < c \end{cases}$ Rango = $\{-1, 1\}$</p>	

CONTINÚA 

Función de transferencia Mixta.

$$a_j = f(\text{Net}_j) = \begin{cases} 1 & \text{si } \text{Net}_j \geq C \\ \text{Net}_j & \text{si } c \leq \text{Net}_j < C \\ 0 & \text{si } \text{Net}_j < c \end{cases}$$

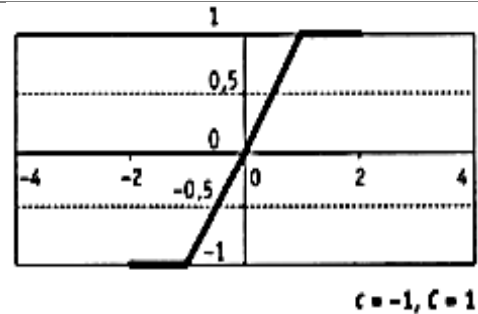
$$\text{Rango} = [0, 1]$$



Función de transferencia Mixta.

$$a_j = f(\text{Net}_j) = \begin{cases} 1 & \text{si } \text{Net}_j \geq C \\ \text{Net}_j & \text{si } c \leq \text{Net}_j < C \\ -1 & \text{si } \text{Net}_j < c \end{cases}$$

$$\text{Rango} = [-1, 1]$$

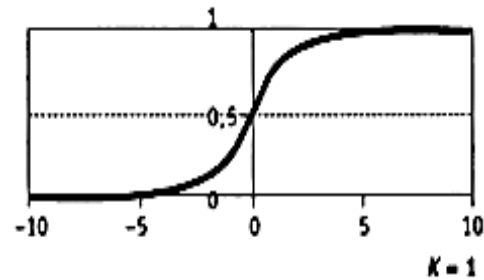


Función de transferencia Sigmoide o Logística.

$$a_j = f(\text{Net}_j) = \frac{1}{1 + e^{-k \text{Net}_j}}$$

$$\text{Rango} = (0, 1)$$

$$k > 0$$

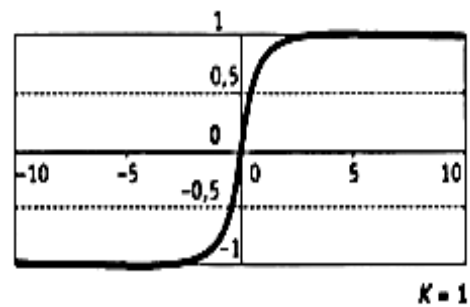


Función de activación Tangente, Tangente Hiperbólica o Tangencial.

$$a_j = f(\text{Net}_j) = \frac{e^{k \text{Net}_j} - e^{-k \text{Net}_j}}{e^{k \text{Net}_j} + e^{-k \text{Net}_j}}$$

$$\text{Rango} = (-1, 1)$$

$$k > 0$$

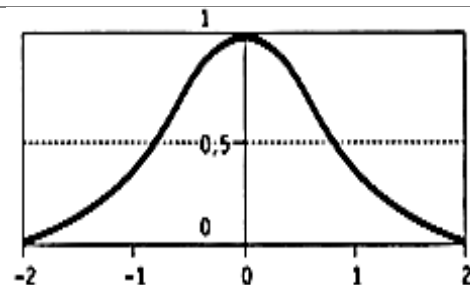


Función de transferencia Gaussiana.

$$a_j = f(\text{Net}_j) = Ae^{-B(\text{Net}_j)^2}$$

$$\text{Rango} = (0, 1)$$

$$A > 0, B > 0$$

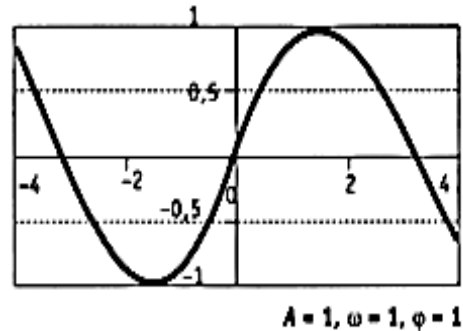


CONTINÚA

Función de transferencia Sinusoidal.

$$a_j = f(\text{Net}_j) = A \cdot \text{sen}(\omega \cdot \text{Net}_j + \varphi)$$

$$\text{Rango} = (-1, 1)$$



Fuente: (Matich, 2001)

d) Función de Salida

Finalmente, el valor obtenido de esta función es la respuesta de la neurona i (out_i); por lo tanto, la función de salida determina qué valor se transfiere a las neuronas vinculadas. Si la función de activación está por debajo de un umbral determinado, ninguna salida se pasa a la neurona subsiguiente. Normalmente, no cualquier valor es permitido como una entrada para una neurona, por lo tanto, los valores de salida están comprendidos en el rango $[0, 1]$ o $[-1, 1]$. También pueden ser binarios $\{0, 1\}$ o $\{-1, 1\}$. Dos de las funciones de salida más comunes son:

- **Ninguna:** este es el tipo de función más sencillo, tal que la salida es la misma que la entrada. Es también llamada función identidad.
- **Binaria:** 1 si la función de $\text{act}_i \geq \sum i$, donde $\sum i$ es el umbral o 0 de lo contrario.

e) Tipos de Aprendizaje

En una red neuronal lo más importante es que la misma pueda aprender a diferenciar entre lo que se le está enseñando y el resto del mundo, algunos de los métodos de aprendizaje que se detallan a continuación” (Gómez, Fernández, López, & Díaz, 2009):

Aprendizaje Supervisado: es un caso de entrenamiento con profesor y utiliza información global, se presentan dos vectores (entrada y salida deseada). La salida computada por la red se compara con la salida deseada, y los pesos de la red se modifican

en el sentido de reducir el error cometido. Se repite iterativamente, hasta que la diferencia entre salida computada y deseada sea aceptablemente pequeña.

Aprendizaje No Supervisado: es un caso de entrenamiento sin profesor y sólo usa información local durante todo el proceso de aprendizaje, es un modelo más cercano al sistema biológico, no se utiliza el vector de salida esperada, y sólo hay vectores de entrada en el conjunto de entrenamiento. El algoritmo modifica los pesos de forma que las salidas sean consistentes, es decir, que, a entradas muy parecidas, la red compute la misma salida. Las salidas se asocian a las entradas de acuerdo con el proceso de entrenamiento. El proceso extrae características, abstrayendo las propiedades colectivas subyacentes del conjunto de entrenamiento, y agrupa por clases de similitudes.

1.8 Visión Artificial

1.8.1 Definición

Es un sub-campo de la inteligencia artificial, es muy notorio observar esto en la naturaleza puesto que mientras mayor es el desarrollo de inteligencia en una especie, su capacidad visual también aumenta. La visión artificial es la capacidad de adquirir, procesar, entender imágenes digitalizadas que pueden ser usadas en diferentes propósitos.

1.8.2 Proceso de visión artificial.

El ser humano para la creación de sus inventos se inspira tanto en la naturaleza como en sí mismo, hay que considerar que a la naturaleza le ha llevado cientos y hasta miles de años de desarrollo y evolución para alcanzar lo que ahora se observa y usa de una u otra manera, así pues una de las acciones más difíciles de realizar es el reconocimiento a través de la visión por la gran cantidad de parámetros que intervienen en la misma tales como: luminosidad, intensidad de los colores, contrastes, cantidad colores, campo de visión, entre otros.

Se puede definir que la visión es un proceso sumamente complejo en el cual intervienen múltiples parámetros los mismos que se los realiza de forma ordenada permitiendo transformar las diferentes ondas lumínicas captadas, en información

procesable y entendible, en la figura 10 se observa de forma simplificada los pasos necesarios para la realización de visión artificial.

Módulo de digitalización. Es la parte del proceso en la cual la luz recibida del medio ambiente se transforma en una señal digital, la misma que se basa en una matriz formada por píxeles.

Memoria de imagen. Después de haber transformado la información visual captada del entorno y haber sido digitalizada estos datos se proceden a guardar o almacenar para su posterior análisis, procesamiento y visualización.

Procesado de imagen. En esta etapa del proceso se resuelve e interpreta los datos obtenidos del entorno, los mismos que ya han sido guardados en la memoria, esta es la parte del proceso que más energía y tiempo requiere puesto que en ella se realizan múltiples cálculos y procesos matemáticos.

Módulo de visualización. Convierte la señal digital ubicada en la memoria electrónica, de manera que el operador pueda comprender los datos obtenidos generalmente una señal de vídeo o imagen analógica que puede ser visualizada.

Módulo de entradas/salidas, comunicaciones y procesador auxiliar. Tramita la entrada sincrónica de captación de imagen y las salidas de control que actúan en los dispositivos externos en función del resultado de la inspección (Gonzalo Pajares, 2001).

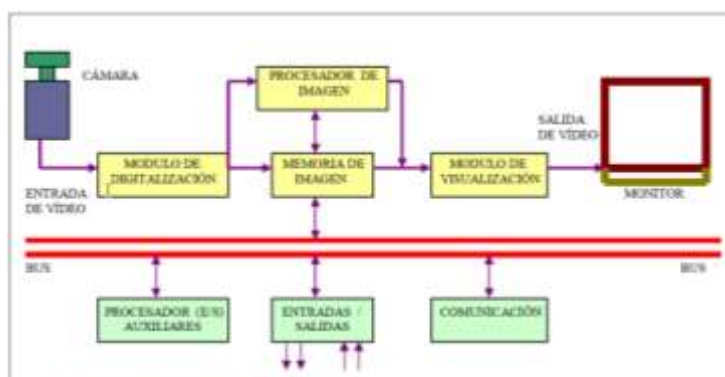


Figura 10. Proceso de visión artificial.

Fuente: (ETI, 2013)

1.8.3 Componentes de un sistema de visión artificial

Para la ejecución de un proceso tan complejo como es la visión artificial es necesario el uso de diversos elementos que permitan adquirir, procesar y guardar los datos del entorno a analizar.

Cámara.

Dentro del proceso de visión artificial, la parte más esencial es la que permite adquirir las imágenes por medio de un sensor óptico, el mismo que transforma la luz recibida en un pulso eléctrico el mismo que luego es procesado para la obtención de una imagen digital. Ver Figura 11.

Existen diferentes tipos de cámaras las mismas que se eligen dependiendo de la función que va a desempeñar, algunos de los parámetros más importantes a tener en cuenta son:

- El tipo de onda electromagnética que puedan leer (Los colores que puede captar).
- Resolución, cantidad de pixeles.
- Tipo de lente (zoom, apertura y ángulo de visión permitida).
- Tamaño de la cámara y su forma de sujeción.
- La resistencia a las inclemencias ambientales.
- Estabilidad frente al movimiento.
- Costo y mantenimiento.



Figura 11. Cámara y sus partes.

Fuente: (Melamed, 2011)

Sistema de iluminación.

Es muy importante para el sistema de visión artificial puesto que las cámaras capturan el reflejo de la luz sobre los objetos, el propósito de la iluminación es la forma en la cual se quiere que nuestra cámara vea el objeto, así pues, se tiene diferentes tipos de iluminación como por ejemplo iluminación por fluorescentes, fibra óptica, leds, natural entre otras, se descarta la iluminación natural puesto que es la que se usará en este proyecto.

- **Iluminación natural.** Es aquella que proviene del ambiente en el que se desarrolla la visión artificial, esta es una de las más arduas de usar debido al alto y variado cambio de la misma (Guilcamaigua Maisincho & Vaca Navas, 2013). Ver figura 12.



Figura 12. Luz reflejada por el sol.

Fuente: (Lechner, 2005)

Grabbers (tarjetas PC).

En las aplicaciones de visión y de análisis de imagen es necesario tomar las imágenes con la mejor calidad posible y enviarlas a la memoria del ordenador con el fin de procesarlas, examinarlas y visualizarlas. Las cámaras que se utilizarán en estos entornos presentan una serie de requisitos. Los detalles y precios de las placas de captura de imagen (Frame Grabbers) varían enormemente y por tanto se deben tener en cuenta los requerimientos técnicos de cada frame grabber para su discriminación.

Software.

Con el rápido progreso de los sistemas informáticos se consigue visualizar las imágenes en tiempo real, lo que permite realizar la mayoría de procesos en tiempos suficientemente cortos, como para que puedan resolver aplicaciones de visión en ambientes complicados, con los resultados esperados en tiempo real.

Esta evolución de dispositivos ha incitado el desarrollo de librerías de visión que puedan funcionar en entornos estándar de todo tipo de sistemas operativos como de procesadores. El pilar fundamental del software de un sistema de visión es la observación y estudio de los píxeles, obteniendo resultados tales como: desde la medida de una partícula, a la valoración o lectura de una serie de caracteres (OCR), pasando por cualquier otro proceso de visión artificial que se pueda imaginar para obtener la mayor cantidad de beneficios de una imagen. (López, 2012)

1.8.4 Procesamiento de imágenes

Es el tratamiento, el almacenamiento, transmisión, representación y manipulación de información, de imágenes digitales por medio de una computadora digital usando un software adecuado. (DeMello, 2003)

Imágenes digitales.

Las imágenes digitales pueden ser de distintos formatos. Cada una se corresponde con una extensión específica del archivo que lo contiene, así como se observa en la figura 13. Los más utilizados en la actualidad son:



Figura 13. Formatos de imágenes digitales.

Fuente: (España, 2008)

- **RAW.** Es comparable a la imagen latente que contiene la película expuesta sin revelar. (Wright, 2001) Recoge puntualmente lo que atrapa el sensor. Nada más y nada menos. Esto significa que el fotógrafo puede conseguir la mayor calidad de imagen posible. Los archivos JPG a máxima calidad ocupan algo más de 2 MB, mientras que un RAW necesita entre 5 y 6 MB además cada cámara ocupa su propio formato RAW.
- **BMP (Bitmap = Mapa de bits).** Han sido muy manejados porque fueron desarrollados para aplicaciones de Windows, la fotografía se forma usando una matriz de píxeles, este formato no sufre pérdidas de calidad y por tanto resulta adecuado para guardar imágenes que se desean manipular posteriormente, además, guarda gran cantidad de información y los archivos llegan a alcanzar tamaños mayores de los que se obtiene con el formato JPG.
- **GIF (Graphics Interchange Format = Formato de Intercambio Gráfico).** Fue creado con el objetivo de comprimir imágenes digitales reduciendo la paleta de colores a 256 colores como máximo (profundidad de color de 8 bits). Acepta escalas de menor número de colores y por ende permite optimizar el tamaño del archivo que contiene la imagen.
- **JPG-JPEG (Joint Photographic Experts Group = Grupo de Expertos Fotográficos Unidos).** A diferencia del formato GIF, acepta una gama de hasta 16 millones de colores, es el formato más común junto con el GIF para colgar imágenes en la red. La compresión JPEG puede suponer cierta pérdida de calidad en la imagen. En la generalidad de los casos esta pérdida se puede dar porque permite reducir la capacidad del archivo y su visualización es admisible.

- **TIF-TIFF (Tagged Image File Format = Formato de Archivo de Imagen Etiquetada).** Recopila retratos de una calidad atractiva al recurrir a cualquier profundidad de color de 1 a 32 bits es ideal para editar o imprimir una imagen, genera archivos muy grandes.
- **PNG (Portable Network Graphic = Gráfico portable para la red).** Es un nuevo formato que se ha difundido recientemente y alternativo al GIF, su tasa de compresión es 10% mayor que el GIF permite usar más de 256 colores. (Luna, 2012)

Pasos fundamentales en el procesamiento digital de imágenes.

Generalmente los pasos fundamentales para el procesamiento de imágenes son:

- **Adquisición de imágenes.** En esta fase se encuentran los sensores de imágenes y la capacidad para la digitalización de las señales recogidas por las cámaras.
- **Realce de la imagen.** En esta parte se procede a trastornar las características que conforman a la imagen color, pixeles, luminosidad entre otros. Con el objetivo de adquirir los datos más notables de la imagen.
- **Restauración de la imagen.** También se corresponde con la mejora del aspecto de una imagen. A diferencia del concepto de realce, el cual es individual, la restauración es equitativa, es decir, utiliza técnicas basadas en matemáticas y en probabilidad para modelar la degradación.
- **Procesamiento de imagen en color.** Este no es muy utilizado debido a que presenta una gran cantidad de datos, aunque el uso de este procesamiento permite obtener características que no se identifican en imágenes que solo se procesan en escala de grises.
- **Ondeletas y procesamiento de multiresolución.** Las ondeletas o Wavelets son las bases para simbolizar imágenes en distintas resoluciones o escalas.

- **Compresión.** Es disminuir el tamaño de la imagen, expresado en Kb para tener una forma más sencilla de almacenarla o enviarla.
- **Procesos morfológicos.** Es el grupo de equipos utilizados para extraer aquellos elementos de una imagen que son útiles en la representación y descripción de formas.
- **Segmentación.** Es un método que se utiliza para extraer o aislar del resto de la imagen los objetos para su posterior examen. Se da en dos sentidos:
 - a) **Segmentación autónoma**, la cual puede facilitar procesos subsecuentes.
 - b) **Salidas, píxeles y bordes**, que indican el límite existente entre objetos, esquinas, texturas, etc.
- **Representación y descripción.** Consiste en seleccionar particularidades que posteriormente serán definidas como bordes, inflexiones, formas internas como texturas, formas del esqueleto, etc.
- **Reconocimiento.** Permite establecer una etiqueta a un objeto, fundamentado en la información provista por los descriptores, o bien establecer un significado a un conjunto de objetos ya registrados.

El conocimiento acerca del dominio del problema está recopilado dentro del sistema de procesamiento de imágenes en forma de una base de datos. Este conocimiento puede ser tan simple como puntualizar las regiones en donde la información de interés puede ser situada o bien compleja delimitando un inventario de características obtenidas de disímiles argumentos. (Matich, 2001)

1.8.5 Detección de bordes

Las orillas de una imagen conforman detalles de grano fino que conciernen con cambios más o menos bruscos, abruptos, en la variación de luminosidad que permite comprimir notablemente la cuantía de información a procesar. (Casanova, 2012)

Detección de bordes, algoritmo de CANNY

Uno de los mecanismos afines con la localización de orillas con el uso de la primera derivada, la cual es usada porque adquiere el valor de cero en todas las zonas donde no se altera la intensidad y tiene un valor máximo en todas las transiciones de intensidad, (Rebaza, Detección de bordes mediante el algoritmo de Canny , 2011) ver figura 14. Por tanto, una permutación de intensidad se muestra como un cambio violento en la primera derivada, peculiaridad que es usada para divisar un borde y en la que se fundamentó el algoritmo de Canny. (Nonami, Kendoul, Suzuki, Wang, & Nakazawa, 2010)

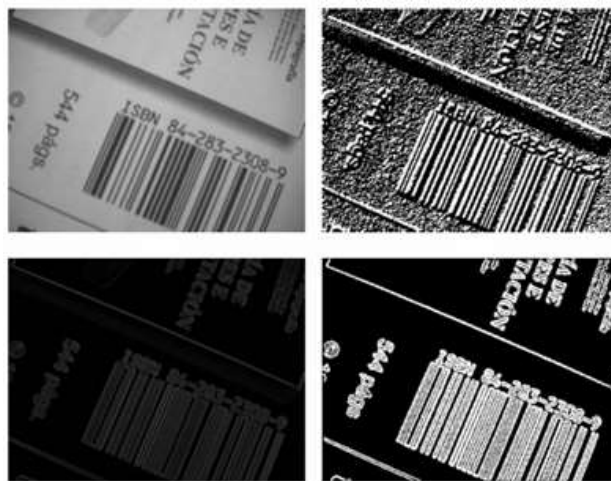


Figura 14. Resultado de aplicar el detector de bordes de Canny.

Fuente: (Nonami, Kendoul, Suzuki, Wang, & Nakazawa, 2010)

El algoritmo de Canny consiste en tres procesos:

- **Obtención del gradiente:** En este paso se calcula la dimensión y orientación del vector gradiente en cada píxel. Se ejecuta la aplicación de un filtro gaussiano a la imagen adquirida, con el objetivo de afinar la imagen e intentar eliminar el ruido existente.
- **Supresión de no máximos:** En este paso se logra la reducción del grosor de los fillos obtenidos con el gradiente, hasta conseguir bordes de un píxel de ancho. Se consideran cuatro direcciones reconocidas por las disposiciones de 0° , 45° , 90°

y 135° con relación al eje plano. Para cada píxel se halla la trayectoria que mejor se acerque a la orientación del ángulo de gradiente.

- **Histéresis de umbral:** En este paso se aplica una función de histéresis basada en dos umbrales; con este proceso se pretende reducir la posibilidad de aparición de contornos falsos. Tomar la imagen obtenida del paso anterior, tomar la orientación de los puntos de borde de la imagen y tomar dos umbrales, el primero más pequeño que el segundo. Para cada punto de la imagen se debe localizar el siguiente punto de borde no explorado que sea mayor al segundo umbral. A partir de dicho punto seguir las cadenas de máximos locales conectados en ambas direcciones perpendiculares a la normal del borde, siempre que sean mayores al primer umbral. Así se marcan todos los puntos explorados y se almacena la lista de todos los puntos en el contorno conectado.

Un mecanismo muy manejado es el algoritmo de Deriche y Cocquerez. Este algoritmo esgrime como entrada una imagen binarizada (0,1) de alrededores de un píxel de ancho. El algoritmo indaga los extremos de los alrededores francos y sigue la dirección del máximo gradiente hasta taponarlos con otro extremo abierto. (Pajares Martin-Sanz, 2010)

1.8.6 Flujo óptico

El flujo óptico puede ser determinado como el movimiento aparente de los patrones de intensidad en una imagen, la mayoría de las metodologías existentes para la evaluación del flujo óptico se puede catalogar en 4: las basadas en gradientes espacio-temporales, las basadas en comparación de regiones, las basadas en fase y las basadas en energía. (Mora & Andres, 2010)

En todas las maniobras de valoración de flujo óptico se empieza con la suposición de los niveles de gris perduran constantes ante movimientos espaciales en un tiempo determinado. (Pino, Gómez, & de Abajo, 2001)

1.8.7 Lucas Kanade

Este método asume que el flujo óptico es invariable en una región R . Donde R es una región de la imagen, y su vector de flujo óptico asociado (Rebaza, Detección de bordes

mediante el algoritmo de Canny , 2010).Es el algoritmo mediante el cual se logra obtener la comunicación deseada de dos imágenes.

Este algoritmo está pensado para calcular el flujo óptico de una secuencia de vídeo, es decir, para realizar seguimiento de objetos. Actualmente está apreciado como una de las metodologías clásicas en el procesamiento de imágenes, y ha sido aplicado y adaptado a solucionar diversos problemas. Está diseñado para funcionar con secuencias de vídeo, por lo que solo funciona para imágenes con una variación entre punto y punto muy pequeña figura 15. A cambio de este atenuante, lo que se logra son tiempos de realización muy buenos. (Riba, 2004)

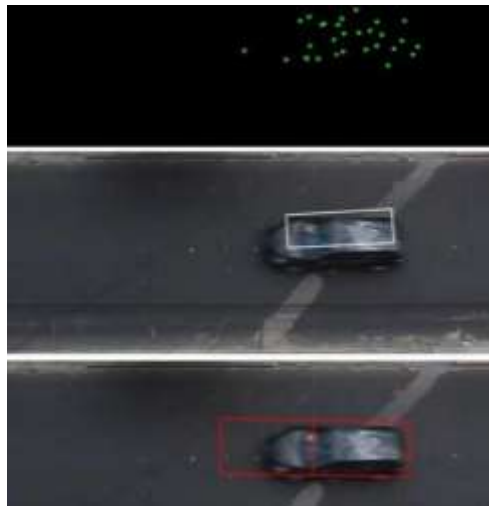


Figura 15. Puntos característicos y solución Algoritmos.

Fuente: (Mora & Andres, 2010)

1.9 Usos y aplicaciones

El campo de la inteligencia artificial ha permitido el desarrollo de diversas tecnologías que facilitan la vida del hombre, realizando complejos procesos de análisis en los que sistemas de control comunes no se pueden implementar; así también el uso de sistemas de visión artificial permiten mejorar la calidad de vida de las personas pudiendo realizar análisis complejos de imágenes y videos para la realización de distintas aplicaciones, otro elemento tecnológico que en la actualidad se está volviendo muy habitual y con grandes aplicaciones son los sistemas UAV o drones.

Es interesante observar la increíble variedad de problemas que estos elementos han solucionado trabajando de forma individual en diversos campos tales como industria, domótica, robótica, agricultura, investigación médica, etc.

Estos elementos se fusionan de formas diferentes para solucionar problemas puntuales, así como mejorar la capacidad independiente de cada uno, dando múltiples proyectos con un alto impacto, a continuación, se detallan algunos de estos usos y aplicaciones.

Forma o apariencia.

Control de la conformidad, a través del uso de visión artificial se realiza el análisis de la forma que debe tener el objeto analizado, así como también sus dimensiones deben ser las adecuadas.

Guiado predeterminado.

Se caracteriza por una situación en la que una cámara acimutal (que le permite moverse en horizontal y vertical) toma una instantánea de la escena y el sistema de visión dirige un robot. Una aplicación típica incluye tareas de paletización de objetos pesados, tales como bloques de motor y cigüeñales, pero también puede incluir trabajos mucho más ligeros tales como el embalaje de chocolates. (Zeidan, 1990)

Identificación de falsificaciones.

Con el uso de redes neuronales artificiales se puede llegar a clasificar formas que tienen ciertas variaciones, las que pueden confundir a sistemas más convencionales.

Robots automatizados y sistemas de control (visión artificial y sensores de presión, temperatura, gas, etc.).

Debido a la facilidad versatilidad y sobretodo adaptabilidad que posee una red neuronal se le facilita la interacción con el medio que lo rodea.

Monitorización en cirugías.

Con el avance de la tecnología se pueden realizar cirugías a cientos o miles de kilómetros esto se debe a la interacción que se ha logrado en los últimos años entre el humano y la máquina de forma inteligente.

Delivery

En algunos países los drones se están encargando de hacer envíos, por ejemplo, en china, la empresa de correo SF Express hace envíos a través de los pequeños helicópteros. Estados Unidos, esta vez se quedó atrás ya que la legislación actual no lo permite.

En situaciones de emergencia

Los drones se enfatizan por su efectividad en situaciones límites, especialmente en áreas que quedaron aisladas o que su acceso es muy complejo. Se demostró la importancia de su uso, por ejemplo, en zonas que fueron azotadas por desastres naturales. Su velocidad de respuesta les permite recorrer áreas enormes en muy poco tiempo. Concediendo llevar la ayuda necesaria en tiempo record, traslado de bancos de sangre o en una fase previa para evaluar la ayuda necesaria en la zona o la forma de arribo al lugar. (Juyamar, 20016)

Reconocimiento y seguimiento en el tiro al blanco.

Esta es una aplicación con un alto grado de complejidad puesto que necesita de movilidad sin arriesgar a las personas, además, de poder reaccionar de forma rápida y anticipada a las situaciones que le presente el campo de batalla.

CAPÍTULO II

EQUIPOS Y CONFIGURACIONES

2.1 Introducción

El presente capítulo detalla los dispositivos utilizados para este proyecto, su funcionamiento, configuración y características principales, mediante los cuales se espera conseguir una sinergia entre hardware y software que permita realizar el reconocimiento de imágenes desde un dron haciendo uso de las redes neuronales.

2.2 Elección del Multirotor

Una investigación de los principales multirrotores disponibles en el mercado para uso comercial, determino tres posibles drones que se adaptan a las necesidades del proyecto el “Dji Inspire Drone”, “Dji Phantom 3” y el “Quantum FVP Drone” mostrados en la figura 16, cuentan con cuatro rotores y es que un cuadricóptero presenta una gran capacidad de estabilidad en vuelo que será de mucha utilidad al momento de realizar la identificación del helipad desde el aire, además gracias al número de rotores presentes existe un equilibrio entre maniobrabilidad, control de vuelo y consumo de energía que permitirán un tiempo considerable de autonomía, así también como muchos drones de su clase tiene la capacidad de despegue y aterrizaje vertical, que permite reducir el área de despegue a un espacio compacto y cómodo para el operador, sus características se muestran en la tabla 3.



Figura 16. Elección del multirrotor, a) Dji Inspire Drone, b) Dji Phantom 3, c) Qanum Venture Drone.

Fuente: (Dji, *Dji Inspire 1*, 2014), (HobbyKing, 2015), (Dji, 2014)

Tabla 3.

Multirrotos Dji Inspire Drone, Dji Phantom 3, Qanum Venture Drone.

Drone	Dji Inspire	Dji Phantom 3	Qanum Venture
Peso Total	2935 g.	1280 g.	1490 g.
Max. Tiempo Vuelo	18 min.	20 min.	15 min.
Temperatura Operación	-10 – 40 °C	0 – 40 °C	0 – 42 °C
Max. Velocidad Subida	5 m/s	5 m/s	5 m/s
Max. Velocidad Bajada	4 m/s	3 m/s	4 m/s
Dimensiones (mm)	438x451x301	350x350x305	355x345x312
Capacidad Batería	4500 mAh.	4480 mAh.	5000 mAh.
Vibración Cámara	± 0.03°	± 0.05°	± 0.04°
Angulo Giro Pitch	-125° – +45°	-90° – +30°	± 200 °
Angulo Giro Pan	± 330 °	NA	± 230 °
Tamaño Imagen Cámara	12 Mpx.	12 Mpx.	12 Mpx.
Precio Total	\$ 2000	\$ 1400	\$ 700

Fuente: (Dji, *Dji Inspire 1*, 2014), (Dji, 2014), (HobbyKing, 2015)

En la siguiente casa de la calidad mostrada en la tabla 4, se muestran los aspectos que tienen mayor relevancia para el proyecto y serán de gran importancia al momento de elegir un multirrotor.

Tabla 4.

Matriz QFD para la elección del multirrotor.

CARACTERISTICAS TECNICAS		CORRELACION						
		OO MUY POSITIVA O POSITIVA X NEGATIVA XX MUY NEGATIVA						
REQUERIMIENTOS		EVALUACION COMPETITIVA						
		A. Quanam Drone B. Dji Inspire C. Dji Fanthom						
		1	2	3	4	5	6	7
Peso < 1500 g.	3	5	5	5	5	4	2	ABC
Tiempo de vuelo > 10 min.	3	5	5	4	2	2	3	ABC
Capacidad de la batería > 4000mah	2	3	5	5	4	3	2	ABC
Vibración de la cámara $\pm 0,04$	4	0	0	0	3	2	4	AC
Tamaño de imagen cámara < 13MP	3	0	0	0	4	1	1	ABC
Precio < 800	5	3	4	5	4	2	5	A
Disponibilidad < 1 semana	5	1	1	1	3	5	5	A
IMPORTANCIA % 100		56	65	67	88	70	88	
IMPORTANCIA RELATIVA / 10		6	7	8	9	7	9	

En la anterior matriz QFD se observa que la prioridad al momento de elegir un multirrotor se encuentran sobre diez puntos, donde la principal es la vibración de la cámara y el precio, ambos con una valoración de 9/10 puntos, seguido por la disponibilidad en el mercado ecuatoriano con 8/10 puntos, así también la autonomía de vuelo y la capacidad de la batería con 7/10 y finalmente el peso total del multirrotor con 6/10.

Tomando en cuenta las posibilidades presentadas en la tabla 3, así también lo mencionado en la casa de la calidad en la tabla 4, se ha determinado que el multirrotor que cumple con nuestras expectativas es el Quanam Venture Drone, gracias a sus características de gama media y su precio económico.

2.3 Funcionamiento

Un cuadricóptero posee cuatro rotores, ubicados en los extremos de la estructura o “Frame” la cual es simétrica adquiriendo la forma de + o X, se usa la configuración en X, “cada rotor produce un torque y un empuje con respecto a su eje de rotación, al igual que una fuerza de arrastre que se opone la dirección de giro y por tanto se logra un torque nulo alrededor del eje Z” (Romero Paredes, 2014), dispone de dos pares de hélices opuestas, de las que un par gira en el sentido de las manecillas del reloj y el otro en sentido contrario como se observa en la figura 17, de esta forma si las hélices giran a la misma velocidad,

se consigue un torque resultante igual a cero, lo que permite mantener la aeronave en el aire a una altura determinada.

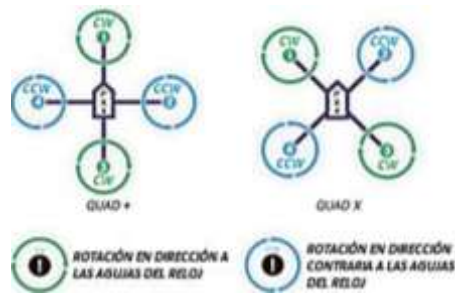


Figura 17. Disposición de hélices en un cuadricóptero.

Fuente: (Romero Paredes, 2014)

El modo “X” es el más común en modelos comerciales, se diferencia del modo “+” por su orientación de vuelo, los ejes “Roll” y “Pitch” actúan sobre los ejes del cuadricóptero como se muestra en la figura 18.



Figura 18. Ejes de giro presentes en un cuadricóptero.

Fuente: (NoRunaway, 2014)

2.4 Estructura

Mediante un diagrama de bloques mostrado en la figura 2.4 a, se define el funcionamiento convencional o manual del dron como opera normalmente mediante un piloto, mientras que su estructura avanzada se muestra en figura 2.4 b, estableciendo su funcionamiento de manera automática o mediante un computador que utiliza las redes neuronales artificiales para aproximar la aeronave al objetivo (helipad).

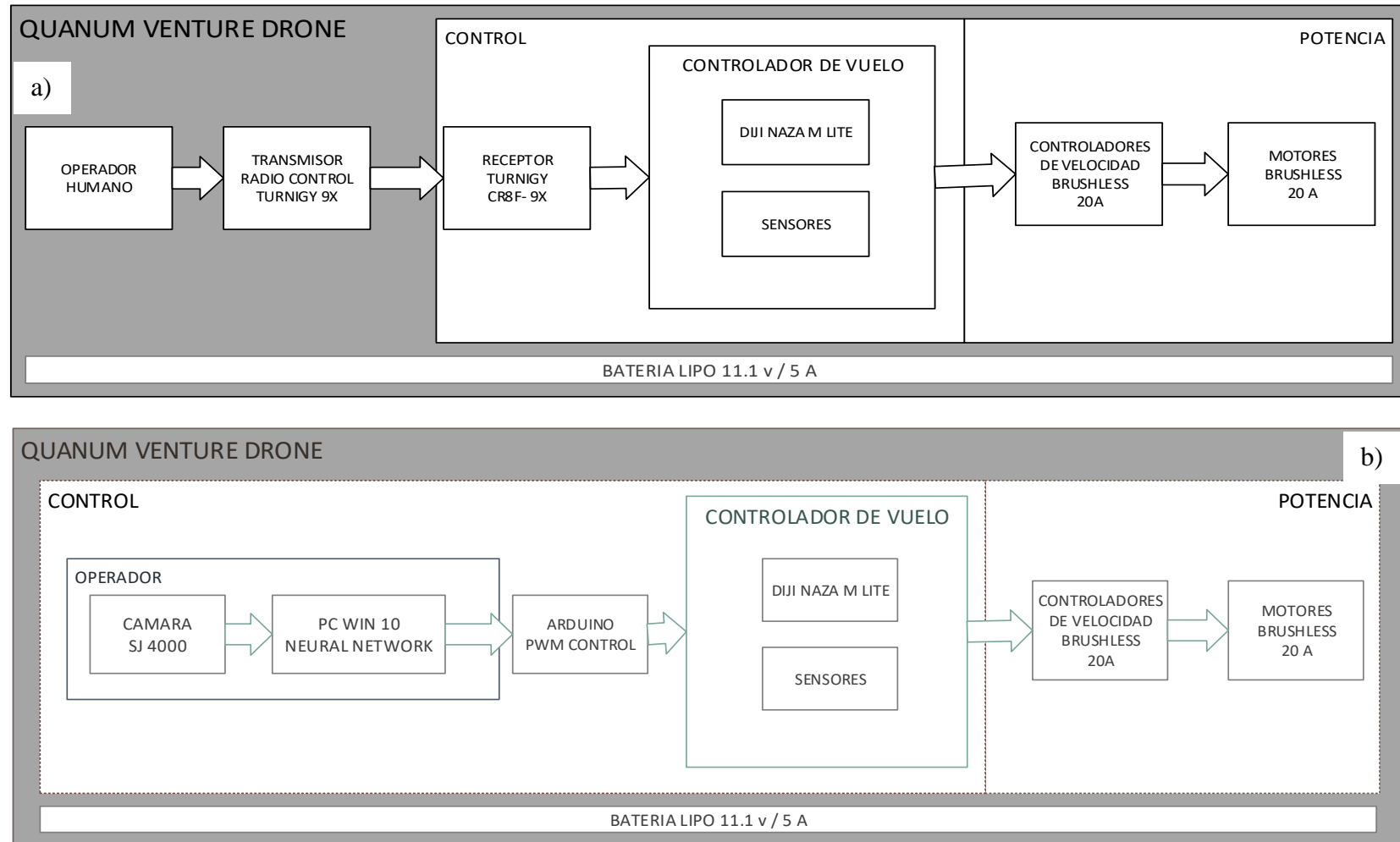


Figura 19. Qnanum Venture Drone. a) Estructura Básica. b) Estructura Avanzada.

Algunas características adicionales de la aeronave “Quantum Venture Drone” se muestran a continuación en la tabla 5.

Tabla 5.

Características adicionales del dron Quantum Venture.

Característica	Descripción
Distancia Ejes	430 mm
Peso	440 g. SN / 1050 Batería
Hélices	8 – 10 inch.
Controlador de Vuelo	Dji Naza M Lite
Distribución de Potencia	Afro 20 A ESC (Variadores de velocidad)
Motores	2213 - 935KV Multistar Motors
Montura Gimbal	45 mm cuadrado y 40 mm línea centro

Fuente: (HobbyKing, 2015)

2.5 Elección de la Emisora

Se ha tomado en cuenta dos emisoras de similares características la “Turnigy 9X” y la “FlySky TH9X”, sus principales características que se exhiben en la tabla 6.

Tabla 6.

Principales características de las emisoras Turnigy 9X y FlySky TH 9X.

Característica	Turnigy 9x	Flysky TH9X
Número de Canales	8 CH PPM / 9 CH PCM	9 CH PPM / PCM
Modos	HELI/ACRO/GLID	HELI/ACRO/GLID
Rango de Radio Frecuencia	2.4 GHz	2.4 – 2.8 GHz
Voltaje Operación	12 v	12 v
Mínimo Voltaje	8.4 v	9 v
Alcance	2 KM	2 KM

CONTINÚA 

Peso	690 g.	670 g.
Tamaño Antena	26 mm.	26 mm.
Tamaño	182x197x120 mm.	185x220x100 mm.

Fuente: (HobbyKing, Turnigy 9X, 2014), (Manualslib, 2013)

Así también se exhibe una matriz QFD en la tabla 7 sobre las necesidades que presentan el proyecto con respecto a la emisora, para así lograr una correcta elección de la misma.

Tabla 7.

Matriz QFD para la elección de la emisora de vuelo.

CARACTERISTICAS TECNICAS		CORRELACION						
		OO MUY POSITIVA O POSITIVA X NEGATIVA XX MUY NEGATIVA						
REQUERIMIENTOS		EVALUACION COMPETITIVA						
		A. Turnigy 9X B. FlySky TH9X						
		1	2	3	4	5	6	7
	PRIORIDAD (1 - 5)							
	# Canales PPM							
	Modos HELI/ACRO/GLID							
	Voltaje Operacion							
	Alcance							
	Disponibilidad Mercado							
	Precio							
Canales PPM > 7 CH.	5	5	2	1	2	4	5	AB
Modo ACRO	5	3	5	1	2	3	2	AB
Voltaje 8 -12 v	3	2	3	5	5	1	1	AB
Alcance Efectivo 50 metros	4	2	2	5	5	5	2	AB
Peso < 700 g.	3	1	1	5	3	3	2	AB
Precio < \$ 100	5	4	4	3	4	2	5	A
Disponibilidad < 1 semana	5	3	1	1	1	5	1	A
IMPORTANCIA % 100		92	80	80	89	102	82	
IMPORTANCIA RELATIVA / 10		9	8	8	9	9	9	

Así se define que nuestra prioridad sobre diez puntos para la emisora del proyecto está distribuida de la siguiente manera, con 9/10 puntos se encuentran las principales prioridades que son el número de canales, la disponibilidad en el mercado, el precio y el alcance efectivo, y con 8/10 se encuentran el modo de vuelo ACRO y el voltaje de

operación de la emisora, con ello se ha determinado que la emisora que cumple con los requerimientos del proyecto es la Turnigy 9X.

2.5.1 Transmisor Turnigy 9X

El transmisor “Turnigy 9x” (9 canales) se exhibe en la figura 20, trabaja a una frecuencia de 2.4 GHz en el espectro de radio frecuencia (RF) capaz de alcanzar distancias de hasta 2 km entre el transmisor y el receptor, este mando será utilizado por el operador humano en caso de presentarse algún mal funcionamiento en el piloto automático para así evitar un siniestro de la aeronave



Figura 20. Transmisor Turnigy 9X.

Fuente: (HobbyKing, Turnigy 9X, 2014)

El transmisor es configurable para varios tipos de aeronaves (drones), se usará el modo ACRO que es el modo de vuelo para multirrotores ya que admite el control independiente de cada canal del receptor, permitiendo un control sobre los diferentes ejes de la aeronave, “Alerón, Elevador, Acelerador y Timón”, ideal para el control de un cuadricóptero en todas direcciones, a continuación, se muestra en la figura 21. La disposición de los botones del transmisor.



Figura 21. Distribución de botones y palancas en el transmisor Turnigy 9X.

Fuente: (Pacini, Turnigy 9X Instruction Manual, 2009).

2.5.2 Configuración

Para configurar el mando Turnigy 9X se procedió de la forma descrita a continuación.

Sistema de vuelo

1. Presionar el botón “Menú” junto a la pantalla del mando.
2. Seleccionar la categoría “SYSTEM” utilizando las flechas de desplazamiento y seleccionándolo con la tecla “Menú”.



Figura 22. Modo “SYSTEM” en transmisor Turnigy 9X.

Fuente: (Pacini, Turnigy 9X Instruction Manual, 2009)

3. Escoger la categoría “MODEL SELE”, la cual permitirá guardar en uno de los ocho espacios de memoria las configuraciones a realizar,



Figura 23. Categoría selección de modo en el transmisor Turnigy 9X.

Fuente: (Pacini, Turnigy 9X Instruction Manual, 2009)

4. Elegir alguno de los espacios de memoria, presionando la tecla “MENÚ”.



Figura 24. Memoria del sistema de vuelo en el transmisor Turnigy 9X.

Fuente: (Pacini, Turnigy 9X Instruction Manual, 2009)

5. A continuación, seleccionar la opción “MODEL NAME”, para escribir el nombre que se le desee asignar a la configuración a ser realizada.



Figura 25. Nombre del usuario en el sistema de vuelo en el transmisor Turnigy 9X.

Fuente: (Pacini, Turnigy 9X Instruction Manual, 2009)

6. Escoger la categoría “TYPE SELE”, para escoger el tipo de aeromodelo a ser utilizado, en este caso se escoge la segunda opción “ACRO”, que permite controlar cada canal de manera independiente y es el que se adapta de mejor a nuestro cuadricóptero.



Figura 26. Tipo de aeronave para sistema de vuelo en el transmisor Turnigy 9X.

Fuente: (Pacini, Turnigy 9X Instruction Manual, 2009)

7. Elegir la siguiente opción “MODE UAT”, para programar el tipo de modulación que utilizar el mando, en nuestro caso, escoger la modulación por posición de pulsos “PPM”.



Figura 27. Tipo de modulación del receptor en el transmisor Turnigy 9X.

Fuente: (Pacini, Turnigy 9X Instruction Manual, 2009)

8. Finalmente, se ubica en la categoría "STICK SET", que determinara la configuración de las palancas en el mando, así se selecciona "MODEL 2", que significa Stick Derecho, ARRIBA / ABAJO: Elevador - DERECHA / IZQUIERDA: Alerones, Stick Izquierdo, ARRIBA / ABAJO: Acelerador - DERECHA / IZQUIERDA: Timón.



Figura 28. Modo de distribución de los Sticks para el sistema de vuelo en el transmisor.

Fuente: (Pacini, Turnigy 9X Instruction Manual, 2009)

Modo de vuelo

En esta configuración realizará la asignación de modos de vuelo que permite utilizar la tarjeta estabilizadora de vuelo "Dji NAZA" y se procederá de la forma siguiente

1. Presionar el botón “MENÚ”, para acceder a la categoría “SETTINGS”.



Figura 29. Categoría “SYSTEM” del modo de vuelo en el transmisor Turnigy 9X.

Fuente: (Pacini, Turnigy 9X Instruction Manual, 2009)

2. Utilizar las flechas de desplazamiento y seleccionar la opción “AUX-CH”.



Figura 30. Canales auxiliares del modo de vuelo en el transmisor Turnigy 9X.

Fuente: (Pacini, Turnigy 9X Instruction Manual, 2009)

3. Configurar los canales 5, 6, 7, 8, 9 de la siguiente manera, utilizar las teclas de desplazamiento y seleccionar cada configuración con el botón “MENÚ”.



Figura 31. Configuración de canales 5,6,7,8,9 en el transmisor Turnigy 9X.

Fuente: (Pacini, Turnigy 9X Instruction Manual, 2009)

- Presionar la tecla “EXIT”, para regresar al menú anterior donde se debe seleccionar la categoría “PROG-MIX”, que permitirá configurar los modos de vuelo presentes en la tarjeta controladora “Dji NAZA”.



Figura 32. Programación de la mezcla de canales del modo de vuelo en el transmisor.

Fuente: (Pacini, Turnigy 9X Instruction Manual, 2009)

- Elegir la opción “MIX1” y configurarla de la siguiente manera, utilizar las flechas de desplazamiento y el botón “MENÚ”.



Figura 33. Programación del MIX1 del modo de vuelo en el transmisor Turnigy 9X.

Fuente: (Pacini, Turnigy 9X Instruction Manual, 2009)

- Seleccionar la opción “MIX2” y configurarla de la siguiente manera, utilizar las flechas de desplazamiento y el botón “MENÚ”.



Figura 34. Programación del MIX 2 del modo de vuelo en el transmisor Turnigy 9X.

Fuente: (Pacini, Turnigy 9X Instruction Manual, 2009)

7. Escoger la opción “MIX3” y configurarla de la siguiente manera, utilizar las flechas de desplazamiento y el botón “MENÚ”.



Figura 35. Programación del MIX 3 del modo de vuelo en el transmisor Turnigy 9X.

Fuente: (Pacini, Turnigy 9X Instruction Manual, 2009)

2.6 Receptor Turnigy 9X

El receptor “Turnigy CR8F – 9X”, se muestra en la figura 36, dispone de 9 canales que emiten señales de control en modulación por ancho de pulso (PWM), promocionales a las señales enviadas desde el transmisor por el operador (humano).



Figura 36. Receptor Turnigy 9X.

Fuente: (HobbyKing, Turnigy 9X, 2014)

2.7 Elección de la placa de control

Se ha tomado en cuenta tres posibles tarjetas de control arduino, “Arduino NANO”, “Arduino MEGA 2560”, “Arduino UNO”, se muestran sus principales características en la Tabla 8.

Tabla 8.

Características de las placas Arduino NANO, MEGA 2560 y UNO.

Característica	Arduino Nano	Arduino Mega	Arduino Uno
Microcontrolador	ATmega 168 / 328	ATmega 2560	ATmega 328P
Voltaje de operación	7 – 12 V	7 – 12 V	7 – 12 V
Pines Digitales (I/O)	14 Pines (6 PWM)	54 Pines (14PWM)	14 (6 PWM)
Pines Analógicos	8	16	6
Corriente DC Pin (I/O)	40 mA	40 mA	20 mA
Memoria Flash	16KB / 32KB	256 KB	32 KB
SRAM	1 KB	8 KB	2 KB
EEPROM	512 Bytes / 1 KB	4 KB	1 KB
Velocidad Reloj	16 MHz	16 MHz	16 MHz

CONTINÚA 

# Canales Seriales	1 TTL Serial	3 TTL Serial	1 TTL Serial
Peso	5 g.	30 g.	25 g.
Tamaño	45 x 18 x 15 mm.	102 x 55 x 11 mm.	69 x 54 x 12 mm.

Fuente: (Arduino, 2015), (Arduino, Arduino MEGA 2560, 2015), (Arduino, Arduino NANO, 2013)

Así también se identifica una matriz QFD en la tabla 9, donde se identifican las necesidades del proyecto con respecto a la tarjeta de control.

Tabla 9.

Matriz QDF para la elección de la tarjeta de control.

CARACTERISTICAS TECNICAS REQUERIMIENTOS	PRIORIDAD (1 - 5)	CORRELACION												
		○○ MUY POSITIVA ○ POSITIVA X NEGATIVA XX MUY NEGATIVA												
		EVALUACION COMPETITIVA												
		# Pines Digitales (I/O)	# Pines Analógicos	# Puertos Seriales TTL	Corriente DC por PIN	Disponibilidad Mercado	Precio	A. Arduino NANO B. Arduino MEGA 2560 C. Arduino UNO						
								1	2	3	4	5	6	7
Pines Digitales PWM > 10	5	5	1	1	4	1	3	B						
Pines Analogicos > 2	3	3	5	2	3	1	3		ABC					
Puertos Serial TTL > 2	5	3	3	5	4	3	3			B				
Corriente DC por PIN > 25 mA.	4	5	4	4	5	3	2				AB			
Peso < 40 g.	3	4	2	2	0	2	2					ABC		
Precio < \$ 30	2	3	2	3	2	2	5						ABC	
Disponibilidad < 1 semana	3	1	1	1	1	5	1							ABC
IMPORTANCIA % 100		90	64	67	76	60	66							
IMPORTANCIA RELATIVA / 10		9	7	7	8	6	7							

Así se identifica la principal prioridad del proyecto con respecto a la placa de control con 9/10 puntos es el número de pines digitales PWM más de diez, seguido de la cantidad de corriente suministrada por cada pin (I/O) mayor a 25 mA con 8/10 puntos, así también el número de pines analógicos, el número de puertos seriales TTL y el precio con 7/10 puntos, y finalmente la disponibilidad en el mercado con 6/10, de lo

cual se puede decir que la placa escogida para el control del proyecto es el Arduino MEGA 2560.

2.7.1 Arduino Mega 2560

La tarjeta arduino “MEGA 2560” se exhibe en la figura 37, permitirá emular las señales PWM de control emitidas por el receptor “Turnigy CR8F–9X” hacia la tarjeta estabilizadora de vuelo “Dji Naza M Lite”, tomando el control de los 5 primeros canales se muestran en la tabla 8, para así conseguir posicionar la aeronave sobre el objetivo (helipad) y lograr su posterior aterrizaje, las características del arduino se muestran en la tabla 2.7.

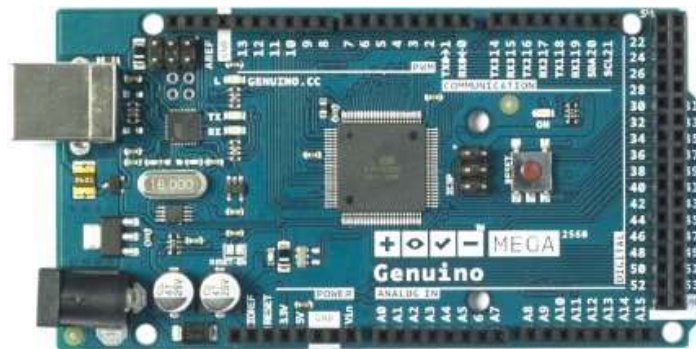


Figura 37. Arduino Mega 2560.

Fuente: (Arduino, Arduino MEGA 2560, 2015)

Tabla 10.

Distribución de canales de control de la aeronave Quantum Venture Dron.

Canal	Identificador	Función
1	Aileron (Alerón)	Controla los movimientos del eje “Pitch”, permitiendo mover la aeronave de izquierda a derecha.

CONTINÚA 

2	Elevator (Elevador)	Controla los movimientos del eje “Roll”, permitiendo mover la aeronave de adelante hacia atrás.
3	Throttle (Acelerador)	Controla la velocidad de los cuatro rotores a la vez, consiguiendo de este modo que la aeronave se eleve a la altura (Eje Z) deseada.
4	Rudder (Timón)	Controla los movimientos del eje “Yaw”, permitiendo girar la aeronave desde una posición estacionaria.
5	Flight Mode (Modo de Vuelo)	Controla los modos de vuelo presentes en la tarjeta estabilizadora “Dji Naza M Lite”

Tabla 11.

Características Arduino Mega 2560.

Característica	Descripción
Microcontrolador	ATmega 2560
Voltaje de operación	5V
Voltaje Alimentación (Recomendado)	7-12 V
Voltaje Alimentación (limite)	6-20 V
Pines Digitales (I/O)	54 Pines (14 Pines PWM)
Pines Analógicos	16
Corriente DC por Pin (I/O)	40 mA
Corriente DC por Pin de 3.3V	50 mA
Memoria Flash	256 KB
Velocidad Reloj	16 MHz
Número de Canales Seriales	3 TTL Serial
Consumo	200 mA/h
Peso	30 g.

Fuente: (Arduino, Arduino MEGA 2560, 2015)

2.8 Elección del tipo de Relés

Para escoger que tipo de relés usar se han considerado dos, del tipo mecánico (accionamiento de contactos mediante bobina) el módulo de relés "SONGLE SRD – 05VD", y del tipo solido (accionamiento eléctrico mediante semiconductores) el módulo "OMROM SSR G3MB – 202P", sus principales características se encuentran la tabla 12.

Tabla 12.

Módulos de relés SONGLE SRD-05VD y OMROM SSR 63MB-202P.

Característica	Songle SRD-05VD	Omrom SSR G3MB-202P
Voltaje de activación	4.5 – 5.5 VDC	4 – 6 VDC
Corriente de activación	0.25 – 1 A	0.1 – 2 A
Voltaje de carga	110 – 250 VAC / 30 VDC	100 – 240 VAC
Corriente de carga	10 A	2 A
Impedancia de entrada	70 Ω \pm 10 %	440 Ω \pm 20 %
Tamaño	750 x 47 x 27 mm.	56 x 52 x 32 mm.
Peso	60 g.	45 g.

Fuente: (Songle, 2012), (OMRON, 2013)

Así también se identifica las necesidades del proyecto con respecto al módulo de relés en una matriz QFD que se exhibe en la tabla 13.

Tabla 13.

Matriz QFD para la elección del módulo de relés.

CARACTERISTICAS TECNICAS		CORRELACION						EVALUACION COMPETITIVA							
		○○ MUY POSITIVA ○ POSITIVA X NEGATIVA XX MUY NEGATIVA						A. SONGLE SRD-05VD B. OMROM SSR G3MB-202P							
REQUERIMIENTOS		PRIORIDAD (1 - 5)	Voltaje de Activación	Corriente de Activación	Voltaje de carga	Corriente de carga	Disponibilidad Mercado	Peso	1	2	3	4	5	6	7
Voltaje de activación < 5,5 V		4	5	4	2	1	1	3	AB						
Corriente de Activación < 40 mA		4	4	5	2	1	1	3		AB					
Voltaje de carga > 15 VDC		3	4	3	5	3	2	2			AB				
Corriente de carga > 1 A		3	4	4	2	5	3	3				AB			
Peso < 50 g.		3	2	2	3	3	3	5					A		
Interferencia magnética mínima		5	2	3	5	5	4	4						A	
Disponibilidad < 1 semana		3	2	1	1	1	5	3							AB
IMPORTANCIA % 100			82	81	74	69	67	83							
IMPORTANCIA RELATIVA / 10			8	8	7	7	7	8							

De lo descrito anterior en la matriz QFD la prioridad para el proyecto es la interferencia magnética, así también el voltaje de activación, corriente de activación y el peso con 8/10 puntos, no menos importante la disponibilidad en el mercado, el voltaje de carga y la corriente de carga con 7 /10 puntos, así se determinó que el módulo de relés de estado sólido OMROM SSR G3MB-202P se adapta mejor a las necesidades del proyecto.

2.8.1 Módulo de relés OMROM SSR G3MB-202P

El cambio “Mando - Computador” y será realizado mediante 2 módulos de relés de estado sólido que se muestra en la figura 38, llamados así por el hecho de que usan transistores y diodos para realizar el cambio de estado de abierto a cerrado, gracias a esta característica, se evita el uso de bobinas que generan campos magnéticos e

interferencias en las señales de control, las conexiones se muestran en la figura 39, las características del módulo se muestran en la tabla 14.



Figura 38. Módulo de relés OMRON SSR G3MB-202P.

Fuente: (OMRON, 2013)

Tabla 14.

Características técnicas de los relés de estado sólido.

Característica	Descripción
Voltaje Alimentación	5 V DC
Corriente Max. Alimentación	160 mA
Voltaje de Control	3.3 – 5 V
Voltaje Max. Salida	250 V
Corriente Max. Salida	2 A
Tamaño	57 x 55 x 25 mm
Consumo	11 mA/h por Relé
Peso	45 g.

Fuente: (OMRON, 2013)

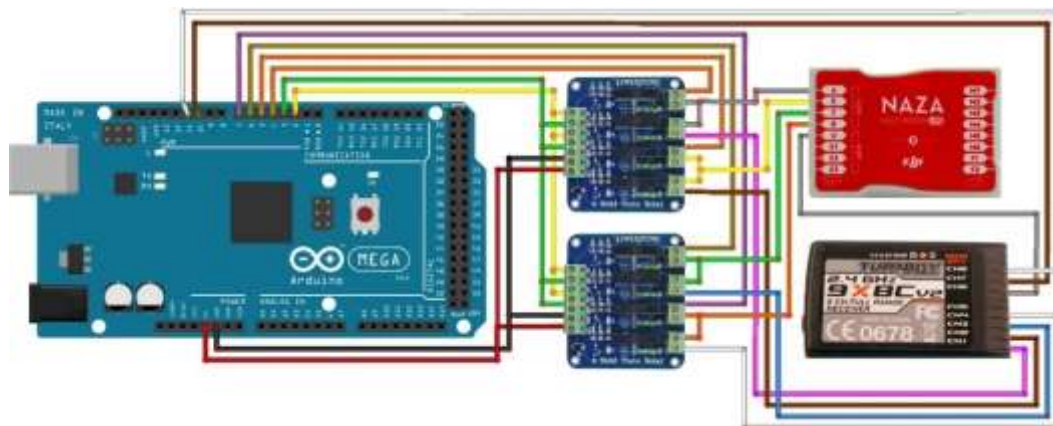


Figura 39. Conexiones a través de los relés de estado sólido.

2.9 Elección del módulo GPS

Se ha considerado dos módulos GPS de similares características, el “GP3906-TLP” y el “GY-GPS6MV2”, sus especificaciones técnicas se muestran en la tabla 15.

Tabla 15.

Módulos GPS GP3906-TLP y GY-GPS6MV2.

Característica	GP3906-TLP	GY-GPS6MV2
Voltaje Nominal	3.3 VDC \pm 5 %	2.7 – 5 VDC
Pila	3 – 4 VDC	1.4 – 3.6 VDC
Chip	ARM7EJ-S	Ublox NEO-6M
Canales	66 CH.	50 CH.
Tipo de datos	UART 8 Bits, NP, Stop Bit	UART 8 Bits, NP, Stop Bit
Comunicación	Serial TTL (4800 – 9600 bps)	Serial TTL (4800 – 152000 bps)
Tiempo de adquisición	de 10 nS. RMS	30 nS. RMS

CONTINÚA 

Altura Max.	18 000 m.	50 000 m.
Velocidad Max.	515 m/s.	500 m/s.
Tamaño	32x42x8 mm	25x35x10 mm
Peso	30 g.	25 g.

Fuente: (ADH-Tech, 2014), (U-Blox, 2014)

También se establece las necesidades del proyecto en relación con el módulo GPS, se muestran en la tabla 16.

Tabla 16.

Matriz QFD para la elección del módulo GPS.

CARACTERISTICAS TECNICAS		PRIORIDAD (1 - 5)	CORRELACION											
			EVALUACION COMPETITIVA											
REQUERIMIENTOS		Voltaje de alimentación	Tipo de comunicación	Tiempo de adquisición	Pila de respaldo	Disponibilidad Mercado	Precio	A. GP3906-TLP B. GY-GPS6MV2						
								1	2	3	4	5	6	7
Voltaje de alimentación < 5,5 VDC		4	5	1	3	2	2	3	AB					
Comunicación serial TTL		4	2	5	4	1	1	1		AB				
Tiempo de adquisición < 50 nS.		5	2	1	5	1	3	3			AB			
Pila integrada para respaldo		4	3	1	1	5	1	2				AB		
Disponibilidad < 1 semana		5	1	2	2	1	5	4					B	
Precio < \$ 50		3	3	2	3	2	2	5						B
Peso < 30 g.		2	1	1	1	1	4	1						AB
IMPORTANCIA % 100			66	51	78	50	70	76						
IMPORTANCIA RELATIVA / 10			7	5	8	5	7	7						

Considerando las necesidades del GPS para el proyecto explicadas en la matriz QFD, se ha determinado que el módulo GPS a ser usado es el GY-GPS6MV2, ya que la prioridad con 8/10 puntos es el tiempo de adquisición de datos, seguido de la disponibilidad en el mercado, el precio y el voltaje de alimentación con 7 /10 puntos.

2.9.1 Módulo GPS GY-GPS6MV2

Para realizar un mejor control sobre la posición del dron y su posible ruta de vuelo, se dispuso de un módulo GPS “GY-GPS6MV2” que se exhibe en la figura 40, su comunicación se realiza por un canal serial independiente, razón la que se hizo uso de un arduino “MEGA 2560” con tres puertos seriales, las conexiones se exponen en la figura 41, así también sus características son manifestadas en la tabla 17.



Figura 40. Módulo GPS GY-GPS6MV2.

Fuente: (U-Blox, 2014)

Tabla 17.

Características técnicas módulo GPS GY-GPS6MV2.

Característica	Descripción
Rango Voltaje Operación	3.3 – 5 V
Tipo Antena	Cerámica
Modo Batería	Ahorro
Indicador Señal	LED
Tamaño Antena	25 x 25 mm
Tamaño Módulo	25 x 35 mm
Velocidad Serial por Defecto	9600 bps
Consumo	72 mA / h
Peso	25 g.

Fuente: (U-Blox, 2014)

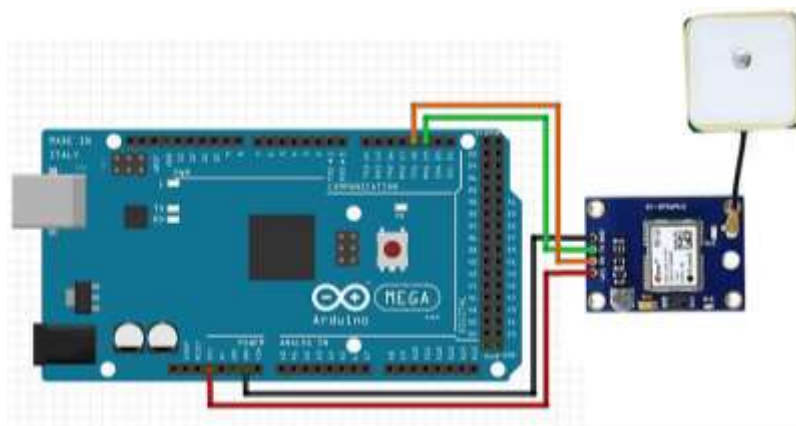


Figura 41. Conexiones del Módulo GPS GY-GPS6MV2

2.10 Elección del sensor de altura

Se ha considerado tres posibilidades para medir la distancia desde el suelo hasta el dron (altura), los sensores ultrasónicos “MB1003”, “HCSR04” y el sensor barométrico de presión atmosférica “BMP180”, sus características se muestran en la tabla 18.

Tabla 18.

Características técnicas de los sensores MB103, HCSR04 y BMP180.

Característica	MB1003	HCSR04	BMP180
Voltaje nominal	2.5 – 5.5 VDC	5 VDC	1.6 – 3.6 VDC
Corriente nominal	25 mA.	15 mA.	3 – 32 uA.
Rango	30 – 500 cm.	2 – 400 cm.	1 – 5000 m
Tiempo pulso	100 uS.	10 uS.	NA
Lectura de datos	Análogo / Digital PWM / Seria TTL	Análogo / Digital PWM	I2C
Perturbaciones	Temperatura	Temperatura	Temperatura, Luz, Humedad

CONTINÚA 

Tamaño	15 x 15 x 10 mm.	45 x 20 x 15 mm.	10 x 5 x 5 mm.
Peso	35 g.	30 g.	5 g.

Fuente: (MaxBotix, 2015), (Elec freaks, 2015), (BOSCH, 2013)

Así también se consideró las prioridades del sensor de distancia para el proyecto, las que se muestran en la tabla 19.

Tabla 19.

Matriz QFD para la elección del sensor de altura.

CARACTERISTICAS TECNICAS		CORRELACION						EVALUACION COMPETITIVA							
		OO MUY POSITIVA O POSITIVA X NEGATIVA XX MUY NEGATIVA						A. MB1003 B. HCSR04 C. BMP 180							
REQUERIMIENTOS		PRIORIDAD (1 - 5)	Rango de medida	Voltaje de alimentación	Corriente de alimentación	Perturbaciones	Disponibilidad Mercado	Precio	1	2	3	4	5	6	7
Rango de medida entre 10 - 300 cm.		5	5	4	4	3	3	4	B						
Voltaje de alimentación < 5,5VDC		4	4	5	5	2	1	1		ABC					
Corriente de alimentación < 25mA.		4	4	5	5	3	1	1			ABC				
Perturbaciones mínimas		5	3	1	1	5	4	4				B			
Disponibilidad < 1 semana		5	3	1	1	4	5	4					BC		
Precio < \$ 30		3	3	2	2	4	4	5						BC	
Peso < 30 g.		3	1	1	1	1	4	1							BC
IMPORTANCIA % 100			99	79	79	95	92	86							
IMPORTANCIA RELATIVA / 10			10	8	8	10	9	9							

Se determinó que la principal prioridad del proyecto para el sensor de altura es el rango de medida y la capacidad de resistir perturbaciones con 10/10 puntos, seguido de la disponibilidad en el mercado con 9/10 puntos, el sensor HCSR04 cumple con lo necesario por lo que será el encargado de medir la altura del proyecto.

2.10.1 Sensor Ultrasónico HC- SR04

El control de altura de la aeronave se realizó mediante un controlador PID del cual se hablará más adelante, haciendo uso del sensor ultrasónico “HC-SR04” que se muestra en la figura 42, sus características se exponen en la tabla 20 y sus conexiones son manifestadas en la figura 43.



Figura 42. Sensor Ultrasónico HC-SR04.

Fuente: (Elecfreaks, 2015)

Tabla 20.

Características Técnicas Sensor HC – SR04

Característica	Descripción
Voltaje de Alimentación	5 V DC
Corriente de Alimentación	15 mA
Frecuencia	40 Hz
Rango Max.	4 m
Rango Min.	2 cm
Angulo Medición	15 °
Señal Entrada Trigger	10 uS. pulso TTL
Señal Salida Echo	Señal de entrada TTL y rango de proporción
Dimensiones	45 x 20 x 15 mm
Consumo	30 mA/h

CONTINÚA 

Peso 9 g.

Fuente: (ElecFreaks, 2015)

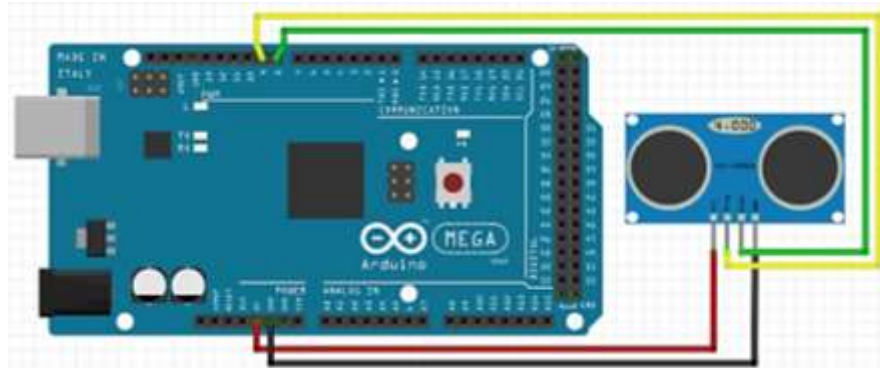


Figura 43. Conexiones realizadas para el Sensor HC – SR04.

2.11 Elección del computador

Se ha considerado dos dispositivos de similares características, el computador “Intel STCK1A32WFC” y la “Raspberry Pi 3, sus principales especificaciones técnicas se muestran en la tabla 21.

Tabla 21.

Características técnicas de la Intel STCK1A32WFC y la Raspberry Pi 3.

Característica	Intel STCK1A32WFC	Raspberry Pi 3
Suministro voltaje	5 VDC	5 VDC
Suministro corriente	2 A	1 A
Procesador	Intel® Atom™ Processor Z3735F (2M Cache, up to 1.83 GHz)	4× ARM Cortex-A53, 1.2GHz
RAM	1 Gb. DDR3L-1333	1GB LPDDR2 (900 MHz)

CONTINÚA 

Puertos USB	1 USB 3.0	4 USB 2.0
Sistema Operativo	Windows 10	Raspbian / Ubuntu / Windows iot
Tamaño	103 x 37 x 12 mm	85 x 55 x 20 mm.
Peso	54 g.	60 g.

Fuente: (Intel, 2015), (Raspberry, 2015)

De igual forma se han establecido las necesidades del proyecto con respecto al computador en la tabla 22.

Tabla 22.

Matriz QFD para la elección del computador.

CARACTERISTICAS TECNICAS		CORRELACION						
		OO MUY POSITIVA O POSITIVA X NEGATIVA XX MUY NEGATIVA						
REQUERIMIENTOS		EVALUACION COMPETITIVA						
		A. Intel STCK1A32WFC B. Raspberry Pi 3						
PRIORIDAD (1 - 5)		1	2	3	4	5	6	7
	Suministro Voltaje							
	Suministro Corriente							
	Procesador							
	Sistema Operativo							
	Disponibilidad Mercado							
	Precio							
Suministro de voltaje < 7 VDC	4	5	4	3	3	2	2	AB
Suministro de corriente < 2A	4	4	5	3	3	2	2	AB
Velocidad de Procesador > 1 Ghz.	5	4	4	5	2	3	3	AB
SO comatible con Visal Studio 2013	5	1	1	5	5	2	2	A
Disponibilidad < 1 semana	3	2	2	3	3	5	3	AB
Precio < \$ 150	2	2	2	5	4	3	5	AB
Peso < 50 g.	3	2	2	2	2	4	1	AB
IMPORTANCIA % 100		77	77	99	82	74	63	
IMPORTANCIA RELATIVA / 10		8	8	10	8	7	6	

Tomando a consideración las prioridades del proyecto, con 10/10 puntos se tiene el tipo de procesador compatible con Visual Studio 2013, seguido del sistema operativo, el suministro de voltaje y corriente con 8/10 puntos, además de la disponibilidad en el mercado con 7/10 puntos y el precio con 6/10 puntos, así se ha determinado que el computador que cumple los requerimientos es la PC Intel STCK1A32WFC.

2.11.1 PC Intel STCK1A32WFC

El computador “PC Intel STCK1A32WFC” se exhibe en la **¡Error! No se encuentra el origen de la referencia.** figura 44, cuenta con las características necesarias para lograr un procesamiento adecuado de la red neuronal en el programa de “Visual Studio”, permitirá la identificación del objetivo “Helipad” ubicándolo en coordenadas X y, enviando así esta información para que la tarjeta arduino “Mega 2560” genere las señales PWM necesaria para posicionar el dron en el lugar de aterrizaje.



Figura 44. PC Intel STCK1A32WFC

Fuente: (Intel, 2015)

A continuación, se muestran las características técnicas tabla 23, que posee este computador, las que permitirán un procesamiento fluido y eficiente en consumo de energía.

Tabla 23.

Características técnicas PC Intel STCK1A32WFC.

Característica	Descripción
Sistema Operativo	Windows 8

CONTINÚA 

Procesador	Intel Atom Z3735F quad-core a 1,33 GHz
Memoria	2 GB (1.333 MHz DDR3)
Conectividad	Wi-Fi 802.11n (Realtek RTL8723BS) + BT 4.0
Puertos	1 salida HDMI, 1 USB 2.0, 1 MicroSD, 1 USB
Tamaño	103 x 37 x 12 mm
Consumo	2000 mA/h
Peso	54 g.

Fuente: (Intel, 2015)

Las conexiones entre el computador, el arduino Mega 2560 y la cámara SJ4000, se muestran en la figura 45, haciendo uso de un HUB para conexiones USB, se consiguió incrementar la capacidad de puertos USB del computador.

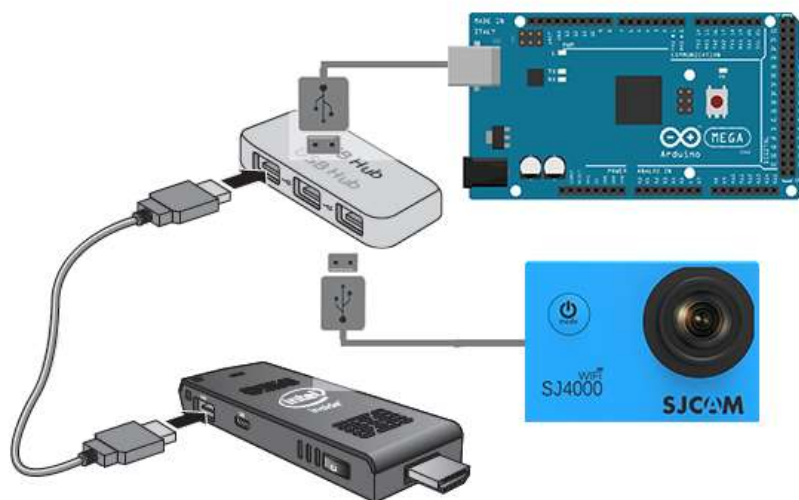


Figura 45. Conexiones PC Intel STCK1A32WFC.

2.12 Elección de la cámara

Se ha considerado dos tipos de cámaras de similares especificaciones, la “GoPro Hero 3” y la “SJCAM 4000”, sus características técnicas se muestran en la tabla 24.

Tabla 24.

Características técnicas de las cámaras GoPro Hero 3 y SJCAM 4000.

Características	GoPro HERO 3	SJCAM4000
Tamaño de Imagen	11 Mpx.	12 Mpx.
Tipo de sensor	CMOS f/2.8	CMOS
Angulo de la lente	160 °	170 °
Capacidad Batería	1050 mA.	900 mA.
ZOOM	2X	4X
Tamaño	56 x 28 x 42 mm.	58 x 30 x 40 mm.
Peso	74 g.	65 g.

Fuente: (GoPro, 2014), (SJCAM, 2015)

Además, se ha tomado en cuenta las necesidades del proyecto con respecto a la cámara, estas se muestran en la Tabla 25.

Tabla 25.

Matriz QFD para la elección de la cámara.

CARACTERÍSTICAS TÉCNICAS	PRIORIDAD (1 - 5)	Tamaño de imagen	ZOOM	Angulo de la lente	Conexión PC	Disponibilidad Mercado	Precio	CORRELACION						
								1	2	3	4	5	6	7
								OO MUY POSITIVA O POSITIVA X NEGATIVA XX MUY NEGATIVA						
								EVALUACION COMPETITIVA						
								A. GoPro HERO 3 B. SJCAM 4000						
REQUERIMIENTOS								1	2	3	4	5	6	7
Tamaño de imagen < 13 Mpx.	3	5	4	3	2	3	3	AB						
ZOOM > 2X	4	4	5	3	2	2	2		B					
Angulo del lente > 150 °	3	4	2	5	2	3	3			AB				
Conexión PC - USB	5	1	3	1	5	3	3				AB			
Disponibilidad < 1 semana	4	2	2	1	3	5	3					AB		
Precio < \$ 150	3	3	3	5	3	3	5						AB	
Peso < 70 g.	4	1	3	3	4	4	1							B
IMPORTANCIA % 100		69	82	72	82	86	72							
IMPORTANCIA RELATIVA / 10		7	8	7	8	7	7							

2.12.1 Cámara SJ4000

La cámara para la adquisición de imágenes es la SJ4000 se muestra en la figura 46 podrá captar la información en unidades de pixeles ya que posee una lente gran angular de 170° para captar una vista panorámica permitiéndonos ubicar con mayor facilidad el objetivo (helipad) dentro de la pantalla, en la figura 47, se muestra sus principales características, la comunicación cámara–pc se realizará mediante comunicación mini-USB de la cual se hablará más adelante.



Figura 46. Cámara SJ4000.

Fuente: (SJCAM, 2015)



Figura 47. Características técnicas de la cámara DJ4000.

Fuente: (SJCAM, 2015)

2.13 Elección de la tarjeta controladora de vuelo

Se ha considerado dos placas controladoras de vuelo, la “Naza M Lite” y la “APM 2.8”, sus características de muestran en la tabla 26.

Tabla 26.

Controladoras de vuelo NAZA M Lite y APM 2.8.

Característica	NAZA M Lite	APM 2.8
Multirrotores soportado	I4, X4, I6, X6, IY6, Y6	H4, X4, I6, X6, IY6, Y6
Transmisor soportado	PCM 2.4GHz	PCM 2.4GHz
Voltaje operación	MC: 4.8-5v Regulador: 7.2-26v	MC: 3.6-5.5v Regulador: 7-12v
Modos de vuelo	Attitude, Manual, GPS	Manual, Attitude, Hold
Protección bajo voltaje	Vuelta casa y aterrizaje	Vuelta casa y aterrizaje
Módulo GPS	GPS/COMPASS High R.	GPS/COMPASS Basic.
Estabilidad	± 1 m.	± 2 m.
Tamaño	MC: 45.5x31.5x18.5 GPS: φ46 x 9 mm.	MC: 65x45x35 GPS: φ52 x 15 mm.
Peso	MC: 25g. GPS: 21.3g.	MC: 35g. GPS: 25g.

Fuente: (Dji, NAZA-M LITE, 2014), (HobbyPro, 2014).

Además, se estableció las necesidades que presenta el proyecto con respecto a la tarjeta controladora de vuelo que se exhiben en la tabla 27.

Tabla 27.

Matriz QFD para la elección de la tarjeta controladora de vuelo.

CARACTERISTICAS TECNICAS		CORRELACION													
		EVALUACION COMPETITIVA													
REQUERIMIENTOS		PRIORIDAD (1 - 5)	Multirotor soportado	Transmisor soportado	Modos de vuelo	Estabilidad GPS	Disponibilidad Mercado	Precio	A. NAZA M Lite B. APM 2.8						
									1	2	3	4	5	6	7
Multirotor X4	5	5	3	1	4	2	2	2	AB						
Transmisor 2.4 GHz.	5	3	5	3	2	2	2	2		AB					
Modo GPS (Posicion - Altura)	5	3	2	5	4	3	3	3			A				
Estabilidad < 1 m	5	3	2	5	5	2	2	2				A			
Disponibilidad < 1 semana	4	2	2	2	2	5	3	3					A		
Precio < \$ 300	3	2	2	4	4	2	5	5						AB	
Peso < 70 g.	3	1	1	1	1	4	1	1							AB
IMPORTANCIA % 100			87	77	93	98	83	75							
IMPORTANCIA RELATIVA / 10			9	8	9	10	8	8							

De la matriz anterior se define que la principal prioridad del proyecto relacionado con la tarjeta controladora de vuelo es, con 10/10 puntos la estabilidad GPS, seguido con 9/10 puntos el tipo de multirotor soportado, el modo de vuelo GPS con compensación de posición y altura, además la disponibilidad en el mercado, el tipo de transmisor y el precio con 8/10 puntos, así se puede decir que la tarjeta controladora de vuelo será la NAZA M Lite.

2.13.1 Controladora de vuelo Dji Naza M Lite

La tarjeta estabilizadora de vuelo “Dji Naza M Lite” que se muestra en la figura 48, funcionará como apoyo para el control de vuelo realizado por la red neuronal, ya que proporciona correcciones necesarias para mantener la aeronave lo más estable posible, permitiendo así que la identificación del objetivo (helipad) mediante visión

artificial se realice la manera más eficiente posible, logrando el posicionamiento y posterior aterrizaje.



Figura 48. Tarjeta estabilizadora de vuelo Dji Naza M Lite.

Fuente: (Dji, NAZA-M LITE, 2014)

La tarjeta “Dji Naza M Lite” cuenta con un módulo GPS que se exhibe en la figura 49, que proporciona dirección y ubicación geográfica al multirrotor, este módulo es propio de la tarjeta controladora, razón por la cual no se puede obtener datos que pueden ser usados para establecer una ruta, es por ello que se optó por utilizar el módulo GPS GY-GPS6MV2, el cual será el encargado de brindar los datos de posicionamiento global para que la tarjeta arduino MEGA 2560 establezca la ruta previamente programada hacia le helipad, también cuenta con un indicador led, que alerta sobre el modo de vuelo y nivel de batería, además de poseer un puerto mini-USB para comunicación entre la placa y la PC.



Figura 49. Módulo Dji GPS e Indicador Led.

Fuente: (Dji, NAZA-M LITE, 2014)

La distribución de canales de control de entrada a la tarjeta se muestra en la figura 50, donde se puede apreciar los 5 canales principales que serán objeto de nuestra investigación, implementando las respectivas señales de control a través de la tarjeta arduino y la red neuronal artificial.

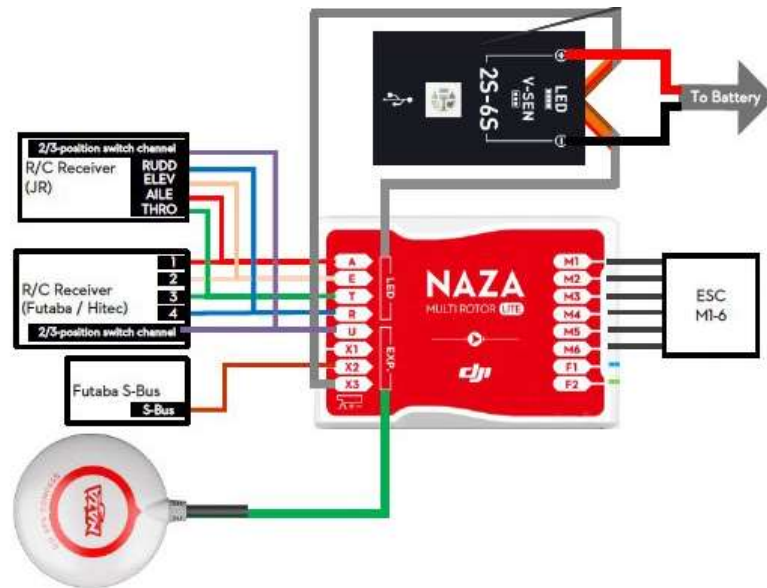


Figura 50. Conexiones de la tarjeta estabilizadora de vuelo Dji Naza M Lite.

Fuente: (Dji, NAZA-M LITE User Manual, 2014)

De la misma manera se encuentran los canales de control de salida que permiten enviar las respectivas señales hacia los variadores de velocidad Afro ESC de 20^a, que controlaran los 4 motores de la aeronave.

Las características de la tarjeta controladora “Dji Naza M Lite”, se muestran a continuación en la tabla 28.

Tabla 28.**Características técnicas de la controladora de vuelo Dji Naza M Lite.**

Característica	Descripción
Multirotores	I4, X4, I6, X6, IY6, Y6
Salida ESC	400 Hz frecuencia de actualización
Transmisor	PCM o 2.4GHz (mínimo 4 canales)
Voltaje de trabajo	MC: 4.8-5v Regulador: 7.2-26v
Consumo de Energía	MAX: 1.5W/0.3A/5V MIN: 0.6W/0.12A/5V
Temperatura	-10°C – 50°C
Software	Windows XP/7/8/10
Dimensiones (mm)	MC: 45.5x31.5x18.5 Regulador: 32.2x21.1x7.7mm GPS: ϕ 46mm x 9
Consumo	200 mA/h
Peso	MC: 25g. Regulador: 20g. GPS: 21.3g.

Fuente: (Dji, NAZA-M LITE, 2014)

2.13.2 Configuración

El procedimiento para la configuración de la tarjeta controladora de vuelo Dji Naza M Lite se realizó de la siguiente manera.

- 1 Conectar el multirotor al computador, encender el mando, energizar el dron y acceder al software de configuración “Dji Naza M Lite Assistant”.
- 2 Acceder a la pestaña “BASIC”, en la opción “AIRCRAFT” donde se debe elegir el tipo de multirotor, la tarjeta controladora de vuelo “Dji Naza M Lite” puede controlar drones de hasta 6 rotores, seleccionar don de cuatro rotores en configuración “X”.

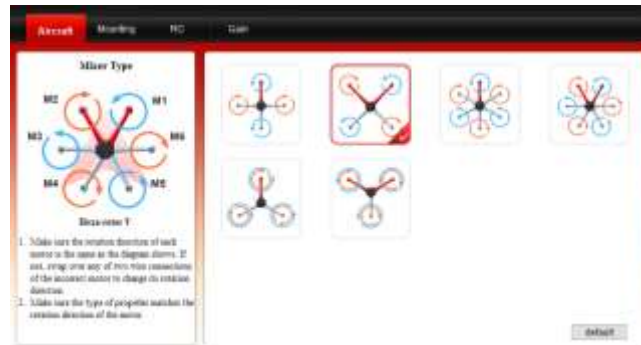


Figura 51. Selección del tipo de multirrotor la tarjeta estabilizadora de vuelo Dji Naza.

Fuente: (Dji, NAZA-M LITE User Manual, 2014)

- 3 Elegir la siguiente pestaña “MOUNTING”, donde se debe escribir las distancias de los ejes “X, Y, Z”, que establecen la posición del módulo “Dji GPS” con respecto a la placa controladora de vuelo “Dji Naza M Lite”.

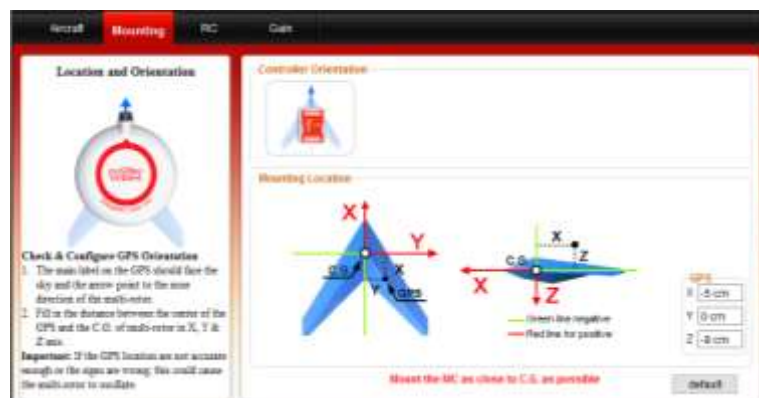


Figura 52. Posición del GPS con respecto a la tarjeta estabilizadora de vuelo Dji Naza.

Fuente: (Dji, NAZA-M LITE User Manual, 2014)

- 4 Seleccionar la siguiente categoría “RC”, aquí se escoge el tipo de receptor utilizado, elegir en “RECEIVER TYPE”, la opción “TRADITION”.

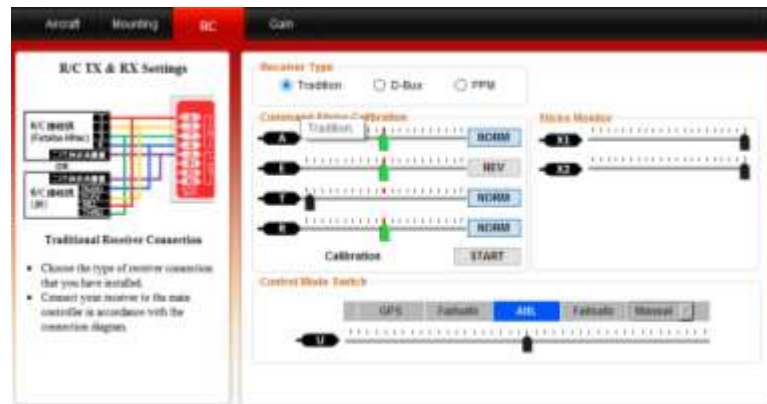


Figura 53. Elección del tipo de receptor usado para la tarjeta de vuelo Dji Naza.

Fuente: (Dji, NAZA-M LITE User Manual, 2014)

- 5 En la misma pestaña, realizar la calibración de los sticks (Palancas) del mando, presionar el botón “START” y mover los sticks de derecha a izquierda y de arriba abajo hasta conseguir que el indicador se quede en el centro, para terminar la calibración presionar el botón “FINISH”.

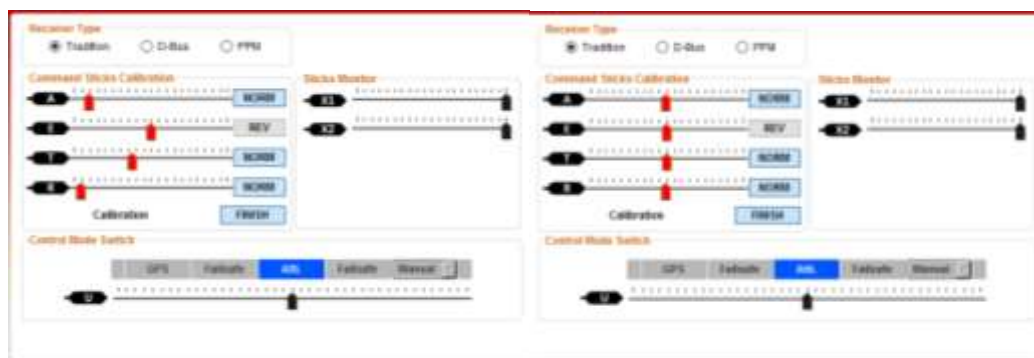


Figura 54. Calibración de Sticks en la tarjeta estabilizadora de vuelo Dji Naza M.

Fuente: (Dji, NAZA-M LITE User Manual, 2014)

- 6 Finalmente, en la parte inferior se muestran los diferentes tipos de vuelo “ATTI, MANUAL, GPS” que ofrece la controladora “Dji Naza M Lite”, asignados al stick “F MODE” del mando.

2.13.3 Modos de Vuelo

MODO DE VUELO ATTI (Attitude)

El modo de vuelo ATTI permite un mejor control sobre la aeronave ya que la controladora de vuelo brinda ajuste y correcciones de altura.

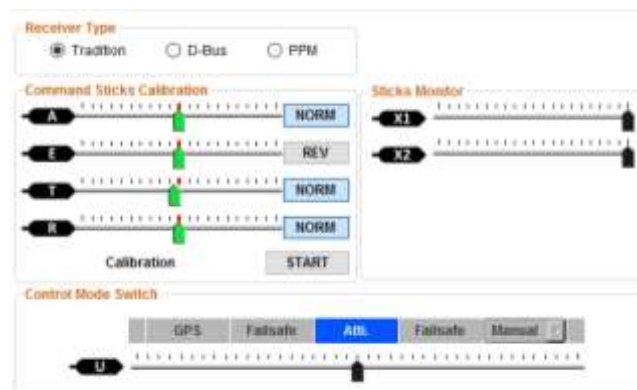


Figura 55. Modo de vuelo ATTI en la tarjeta estabilizadora de vuelo Dji Naza M.

Fuente: (Dji, NAZA-M LITE User Manual, 2014)

MODO DE VUELO MANUAL

El modo de vuelo manual, esta establecido para pilotos con capacidades de vuelo superiores y avanzadas, la controladora de vuelo no brinda correcciones ni ajustes en la trayectoria del multirotor.

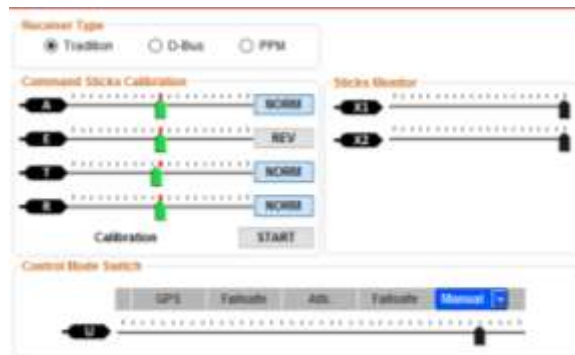


Figura 56. Modo de vuelo Manual en la tarjeta estabilizadora de vuelo Dji Naza M.

Fuente: (Dji, NAZA-M LITE User Manual, 2014)

MODO DE VUELO GPS

El modo de vuelo GPS, brinda al multirrotor ajustes y correcciones de posición, velocidad y altura, por lo que es el modo de vuelo mas seguro que posee la controladora “Dji Naza M Lite”.

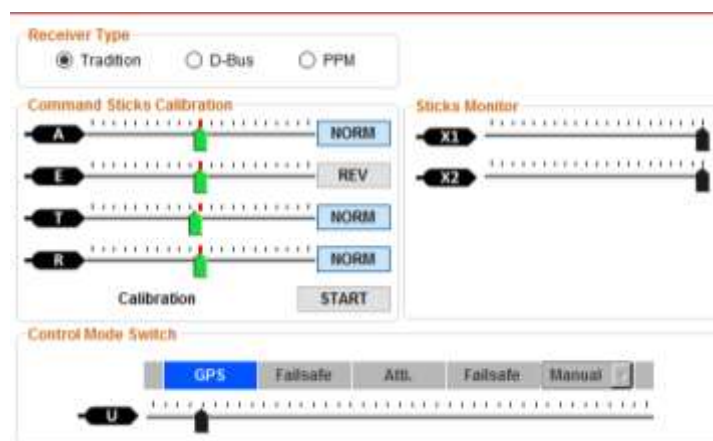


Figura 57. Modo de vuelo GPS en la tarjeta estabilizadora de vuelo Dji Naza M.

Fuente: (Dji, NAZA-M LITE User Manual, 2014)

El modo de vuelo de nuestra elección, es el modo “GPS” ya que brindará al rotor la estabilidad necesaria para la realización del proyecto.

2.14 Elección de los variadores de velocidad

Se ha considerado dos variadores de velocidad de similares especificaciones, el “UBEC ESC 20A” y el “AFRO ESC 20A”, sus características técnicas se muestran en la tabla 29.

Tabla 29.

Variadores de velocidad UBEC ESC 20 A y AFRO ESC 20 A.

Característica	UBEC ESC 20 A	AFRO ESC 20 A
Voltaje de alimentación	5 VDC	5 VDC
Corriente de alimentación	3 A	3 A
Batería soportada	2 – 4 S Lipoly	2 – 4 S Lipoly
Corriente constante	20 A	20 A
Corriente arranque	25 A	30 A
Tipo de motor	Brushless	Brushless
Tamaño	54 x 26 x 11 mm.	45 x 25.7 x 10.5 mm
Peso	30 g.	22.8 g.

Fuente: (HobbyKing, Hobby King 20A ESC 3A UBEC, 2014), (HobbyKing, Afro ESC 20A Motor Speed Controller, 2014)

Así también, se muestran las necesidades del proyecto en relación a los variadores de velocidad en la tabla 30.

Tabla 30.

Matriz QFD para la elección de los variadores de velocidad.

CARACTERISTICAS TECNICAS		CORRELACION						EVALUACION COMPETITIVA							
		OO MUY POSITIVA O POSITIVA X NEGATIVA XX MUY NEGATIVA						A. UBEC ESC 20 A B. AFRO ESC 20 A							
REQUERIMIENTOS		PRIORIDAD (1 - 5)	Voltaje de Alimentación	Corriente de Alimentación	Corriente constante	Corriente de arranque	Disponibilidad Mercado	Peso	1	2	3	4	5	6	7
Voltaje de alimentación < 5,5 V		4	5	4	3	3	3	3	AB						
Corriente de Alimentación < 4 A		4	4	5	4	4	3	2		AB					
Corriente constante > 18 A		4	3	3	5	2	3	3			AB				
Corriente de arranque > 25 A		4	4	4	3	5	3	4				B			
Disponibilidad < 1 semana		3	1	1	3	3	5	3					B		
Peso < 30 g.		5	1	1	3	1	2	5						B	
Tipo de motor Brushless		3	2	3	3	3	4	3							AB
IMPORTANCIA % 100			78	81	93	79	85	91							
IMPORTANCIA RELATIVA / 10			8	8	9	8	8	9							

En la matriz anterior se identifican las prioridades que debe poseer el variador de velocidad, con 9/10 puntos se encuentra el peso y la corriente constante y con 8/10 puntos la disponibilidad en el mercado, la corriente de arranque, el voltaje y corriente de alimentación, así se puede decir que el variador de velocidad que cumple los requerimientos es el AFRO ESC 20 A.

2.14.1 Variadores de velocidad Afro ESC 20 A

Los controladores de velocidad se muestran en la figura 58, poseen una alta eficiencia con un diseño MOSFET de canal N, con una entrada controlada por señal PWM de hardware sin fluctuaciones y una respuesta uniforme que permiten un control de velocidad muy eficiente.



Figura 58. Variadores de velocidad Afro ESC 20A.

Fuente: (HobbyKing, Afro ESC 20A Motor Speed Controller, 2014)

Las características de los variadores de velocidad “Afro ESC 20 A”, se muestran a continuación en la tabla 31.

Tabla 31.

Características técnicas de los variadores de velocidad Afro ESC 20^a.

Característica	Descripción
Consumo de Corriente	20A c.c.
Rango Voltaje	2-4s Li-Poly
Circuito Eliminator Batería (BEC)	0.5 A Linear
Frecuencia Entrada	1 KHz
Firmware	afro_nfet.hex
Cables/Conectores Descarga	16 AWG / Macho 15mm
Cables/Conectores Motor	16 AWG / Hembra 15mm
Peso	22.8 g.
Tamaño	45 x 25.7 x 10.5 mm
Consumo	500 mA/h

Fuente: (HobbyKing, Afro ESC 20A Motor Speed Controller, 2014)

2.15 Elección de los motores

Se ha considerado dos tipos de motores de similares características, el “Multistar Mt2213-935kv” y el “Turnigy D2836/9-950kv”, sus características se muestran en la tabla 32.

Tabla 32.

Motores Multistar Mt2213-935kv y Turnigy D2836/9-950kv.

Característica	Multistar Mt2213-935kv	Turnigy D2836/9-950kv
Rango de voltaje	2 – 4 S Lipoly	2 – 4 S Lipoly
RPM	950kv	935kv
Corriente max.	23.2 A	20 A
Potencia max.	243 W	200 W
Resistencia interna.	0.07 ohm	0.18 ohm
Diámetro del eje	4 mm.	3 mm.
Tamaño	28 x 36 mm.	28 x 26 mm.
Peso	70 g.	55 g.

Fuente: (HobbyKing, MT2213-935KV MultiStar Motor, 2014), (HobbyKing, Turnigy D2836/9 950KV Brushless Outrunner Motor, 2014)

También se identifican las características que deben poseer los motores del proyecto, se muestran en la tabla 33.

Tabla 33.

Matriz QFD para la elección de los motores.

CARACTERISTICAS TECNICAS	PRIORIDAD (1 - 5)	CORRELACION						EVALUACION COMPETITIVA						
		Voltaje de Alimentación	Corriente de Alimentación	Potencia	Diametro Eje	Disponibilidad Mercado	Peso	A. Multistar Mt2213-935kv B. Turnigy D2836/9-950kv						
REQUERIMIENTOS								1	2	3	4	5	6	7
Voltaje de 2-4 S lipoly	3	5	4	4	1	3	4	AB						
Corriente >= 20 A	3	4	5	4	1	3	2		AB					
Ptencia >= 200 W	4	3	3	5	1	3	3			AB				
Diametro >= 3mm.	5	1	1	1	5	1	1				AB			
Disponibilidad < 1 semana	5	1	1	3	1	5	3					A		
Peso < 60 g.	5	1	1	3	1	2	5						A	
RPM > 900kv	4	2	3	3	1	4	3							AB
IMPORTANCIA % 100		62	66	91	49	86	87							
IMPORTANCIA RELATIVA / 10		6	7	9	5	7	9							

Se ha tomado a consideración que la prioridad para los motores es, con 9/10 puntos el peso y la potencia, con 7/10 puntos la disponibilidad en el mercado y la corriente soportada, así se determinó que le tipo de motor que satisface los requerimientos del proyecto es el Multistar Brushless Mt2213-935kv.

2.15.1 Motores Multistar Brushless Mt2213-935kv

Los motores a usar se muestran en la figura 59, están diseñados para maximizar el rendimiento del multirrotor, cuentan con imanes de alta gama “45SH”, rodamientos “NMB” y con 14 polos precisamente equilibrados para un funcionamiento suave, se han construido en estructura de aluminio para su fácil y rápida instalación.



Figura 59. Motores Brushless MT2213-935kv.

Fuente: (HobbyKing, MT2213-935KV MultiStar Motor, 2014)

Las características de los motores “MT2213-935KV Multistar Motor”, se muestran a continuación en la tabla 34.

Tabla 34.

Características técnicas de los motores Brushless MT2213-935kv.

Característica	Descripción
KV (RPM/V)	935
Celdas LIPO	2-4s
Potencia Max.	200 W
Corriente Max. (10s)	15 A
Resistencia Interna	0.18 Ω
Polos	14
Dimensiones	ϕ 28 x 26 mm
Eje	ϕ 3mm
Hélice	10x4.5 (3s) / 8x4.5 (4s)
Consumo	10 000 mA/h
Peso	55 g.

Fuente: (HobbyKing, MT2213-935KV MultiStar Motor, 2014)

2.16 Hélices 8x4.5 (CW/CCW)

Las hélices están determinadas según el tipo de motor así que se usaran las que se muestran en la figura 60, son de fibra de carbono, poseen un estándar de rotación “1 pc/1pc RH” que nos permitirán elevar el multirotor, así como controlar su maniobrabilidad, su dirección de rotación “CW (horario) / CCW (anti horario)” como se muestra en la figura 60.

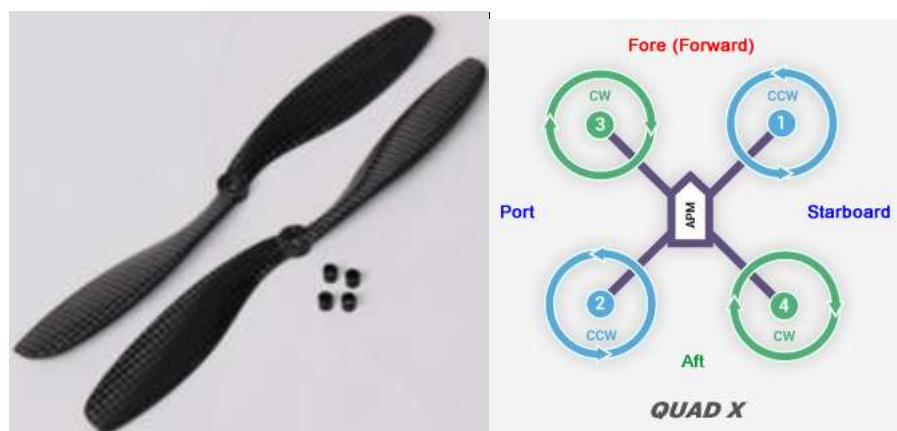


Figura 60. Hélices 8x4.5 y su sentido de giro en el multirotor.

Fuente: (HobbyKing, Propeller 10x4.5 CW/CCW, 2014)

Las características de las “Hélices 8x4.5 CW/CCW”, se muestran a continuación en la tabla 35.

Tabla 35.

Características técnicas de las hélices 8x4.5.

Característica	Descripción
Tamaño	8 x 4.5
Peso	6g. cada una

Fuente: (HobbyKing, Propeller 10x4.5 CW/CCW, 2014)

2.17 Elección de la batería

Se ha realizado un análisis sobre el consumo de energía de cada dispositivo electrónico que será utilizado en el proyecto, el mismo se muestra en la tabla 36.

Tabla 36.

Análisis de consumo de energía del proyecto.

Dispositivo	Consumo
Arduino MEGA 2560	200 mA/h
Receptor Turnigy 9X	90 mA/h
Módulo de Relés OMROM SSR 63MB-202P	11 mA/h x 6 Relés = 66 mA/h
Módulo GPS GY-GPS6MV2	72 mA/h
Sensor ultrasónico HCSR04	30 mA/h
PC Intel STCK1A32WFC	2 000 mA/h (Batería Independiente)
Cámara SJCAM4000	900 mA/h (Batería Independiente)
Controladora de vuelo NAZA M LITE	200 mA/h
Variadores de velocidad AFRO ESC 20 A	500 mA/h x 4 ESC = 2 000 mA/h
Motores Multistar Brushless Mt2213-935kv	10 000 mA/h x 4 Motores = 40 000 mA/h
TOTAL	42 658 mA/h

Se ha considerado la máxima capacidad permisible por el dron “Quantum Venture” para la batería, la cual es de 5200 mA, así se ha realizado una aproximación del tiempo de vuelo.

$$\frac{1}{42\,658} \frac{h}{mA} \times \frac{5\,200\,mA}{1} \times \frac{60\,min}{1\,h} = 7.3139\,min$$

Se ha optado por tres baterías LIPO de similares especificaciones, las baterías “ZOP POWER”, “MULTISTAR” y “TCB”, sus características técnicas se muestran en la tabla 37.

Tabla 37.

Baterías ZOP POWER, MULTISTAR y TCB.

Característica	ZOP POWER	Multistar	TCB
Capacidad	5200 mAh	5200 mAh	5200 mAh
Voltaje	11.1 V	11.1 V	11.1 V
Numero de celdas	3 s	3 s	3 s
Descarga continua	30 C	10 C	25 C
Corriente de carga	1 C (5 A)	1 C (5 A)	1 C (5 A)
Dimensiones	155x 48x 25.5 mm	106 x 44 x 34 mm.	155 x 48 x 25 mm
Peso	375 g.	325 g.	410 g.

Fuente: (Banggood, 2015), (HobbyKing, Multistar High Capacity 3S 5200mAh Multi-Rotor Lipo Pack, 2014), (AliExpress, 2013)

Además, se ha identificado las necesidades del proyecto para la batería, la que se muestran en la tabla 38.



Figura 61. Batería LIPO ZOP POWER 5000mah.

Fuente: (Banggood, 2015)

Posee 3 celdas (3s), que almacenan el voltaje de 11.1 V, así como una corriente de 5 A, suficientes para impulsar la aeronave por los cielos, sus principales características se muestran en la tabla 39.

Tabla 39.

Características técnicas de la batería LIPO ZOP Power 5000mah.

Características	Descripción
Capacidad	5200 mAh
Voltaje	11.1 V
Numero de celdas en serie	3 s
Descarga continua nominal máxima	25 C
Corriente nominal de carga recomendada	1 C (5 A)
Conector de balance hembra	1.5 mm
Conector de descarga hembra	4 mm
Dimensiones	25.5 x 48 x 155 mm
Peso	375 g.

Fuente: (Banggood, 2015)

CAPÍTULO III

PROGRAMACIÓN E IMPLEMENTACIÓN

3.1 Introducción

En el presente capítulo se describen de forma detallada y concisa los procedimientos realizados para la programación de los sistemas autónomos como la generación de señales PWM, el despegue, control de altura, tuta GPS, posicionamiento neuronal artificial y aterrizaje de la aeronave, así también del ensamble e implementación de los componentes descritos en el capítulo anterior, la sinergia entre ellos y la programación para alcázar la realización del proyecto. El capítulo se dividirá en dos aspectos principales, la programación en “Arduino” que contendrá todo lo relacionado con el hardware de control y la programación en “Visual Studio” con lo relacionado al procesamiento de imágenes y redes neuronales artificiales.

3.2 Programación en Arduino

3.2.1 Señales de control PWM

Inicialmente se ha realizado una investigación sobre el tipo de señales que describe el receptor “Turnigy 9X”, para realizar el control de los diferentes ejes que se muestran en la figura 62, donde el eje ”ROLL” confiere el movimiento “Derecha/Izquierda” al multirotor y le fue asignado el nombre “ALERON” o alerón, correspondiente al canal 1, el eje “PITCH” responde al movimiento “Adelante/Atrás” del multirotor, establecido al canal 2 y denominado “ELEVATOR” o elevador, el eje “YAW”, confiere el movimiento “ROTATORIO” del multirotor en su propio eje, corresponde al canal 4 y lleva la denominación “RUDDER” o timón, finalmente el eje “Z”, llamado “THROTTLE” o acelerador conlleva al multirotor el movimiento “Arriba/Abajo” y está asignado al canal 3 del receptor.

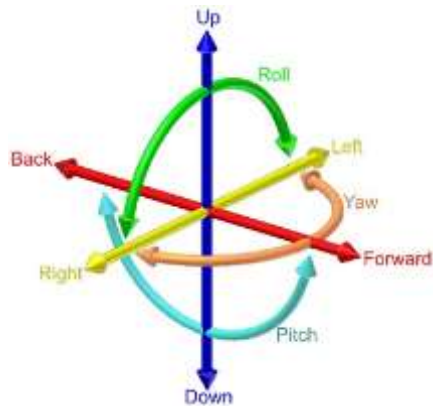


Figura 62. Ejes y movimientos de un multirrotor.

Cada canal posee una señal de control mediante un tren de pulsos, así el control de cada eje se lo realiza mediante la modulación por ancho de pulso o PWM, así se describe las características de cada canal observadas en el osciloscopio a continuación.

ALERÓN/CANAL #1

Se muestra en la figura 63 el ciclo completo de la señal PWM del canal #1/Alerón y a continuación sus características en niveles mínimos, medios y máximos.

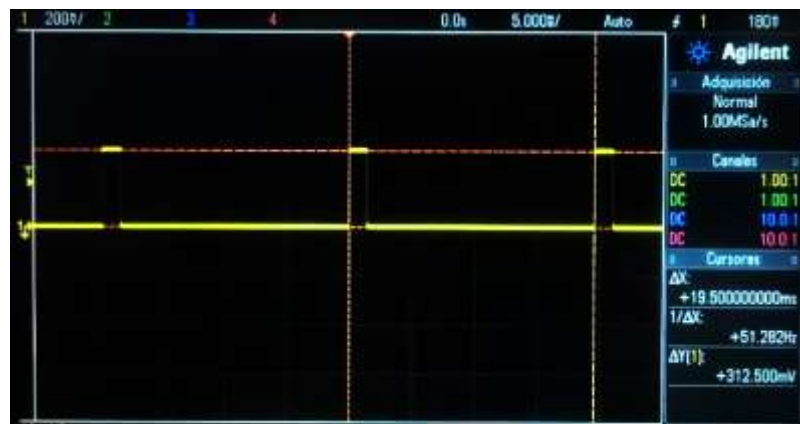


Figura 63. Señal PWM Canal # 1 / Alerón.

Considerando que se investigó las características de las señales PWM en datasheet del equipo “Receptor Turnigy 9X” y no se consiguió estos valores, se optó por obtener

estos datos de forma experimental usando el osciloscopio donde se identificó un periodo de 19500 μS , con el mismo se puede calcular la frecuencia de la señal PWM.

$$T = 19500 \mu\text{S}$$

$$f = \frac{1}{T}$$

$$f = \frac{1}{0,0195 \text{ s}}$$

$$f = 51,282 \text{ Hz}$$

MÍNIMO

Para obtener el nivel mínimo del canal 1 o Alerón se procedió a ubicar la palanca del mando “Turnigy 9X” en la posición de la figura 64.b y se visualizó la señal PWM en el osciloscopio, la cual se muestra en la figura 64.a.



Figura 64. Posición mínima del Stick Alerón b) Señal PWM Canal # 1 / Alerón nivel mínimo (Stick 100% izquierda).

Considerando un periodo de 19500 μS y un ancho de pulso de 1060 μS medidos en el osciloscopio, se obtiene el valor del ciclo de trabajo a continuación.

$$t_{min} = 1060 \mu S$$

$$D_{min} = 100 * \frac{t}{T}$$

$$D_{min} = 100 * \frac{1060 \mu S}{19500 \mu S}$$

$$D_{min} = 5,43 \%$$

MEDIOS

Para obtener el nivel medio del canal 1 o Alerón se procedió a ubicar la palanca del mando “Turnigy 9X” en la posición de la figura 65 y se visualizó la señal PWM en el osciloscopio, la cual se muestra en la figura 65.a.



Figura 65. a) Posición mínima del Stick Alerón b) Señal PWM Canal # 1 / Alerón nivel medio (Stick 50%).

Considerando un periodo de 19500 μS y un ancho de pulso de 1440 μS medidos en el osciloscopio, se obtiene el valor del ciclo de trabajo a continuación.

$$t_{med} = 1440 \mu S$$

$$D_{med} = 100 * \frac{t}{T}$$

$$D_{med} = 100 * \frac{1440 \mu S}{19500 \mu S}$$

$$D_{med} = 7,38 \%$$

MÁXIMOS

Para obtener el nivel máximo del canal 1 o Alerón se procedió a ubicar la palanca del mando “Turnigy 9X” en la posición de la figura 3.5.b y se visualizó la señal PWM en el osciloscopio, la cual se muestra en la figura 3. 5.a.



Figura 66. a) Posición media del Stick Alerón b) Señal PWM Canal # 1 / Alerón nivel máximo (Stick 100% derecha).

Considerando un periodo de 19500 μS y un ancho de pulso de 1880 μS medidos en el osciloscopio, se obtiene el valor del ciclo de trabajo a continuación.

$$t_{max} = 1880 \mu S$$

$$D_{max} = 100 * \frac{t}{T}$$

$$D_{max} = 100 * \frac{1880 \mu S}{19500 \mu S}$$

$$D_{max} = 9,64 \%$$

Así, se define que el rango del ciclo de trabajo para el canal 1 / Alerón es desde 5,43% mínimo hasta 9,64% máximo, y con un rango de ancho de pulso de 1060 μ S hasta 1880 μ S.

ELEVADOR/CANAL #2

Se muestra en la figura 67 el ciclo completo de la señal PWM del canal #2/Elevador y a continuación sus características en niveles mínimos, medios y máximos.

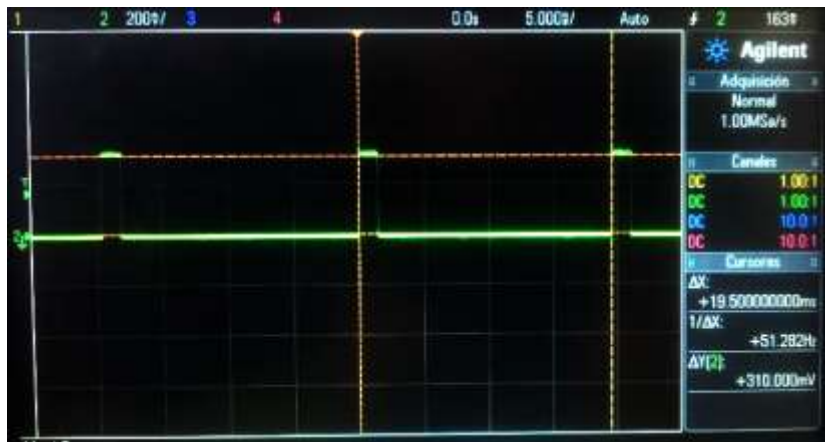


Figura 67. a) Posición máxima del Stick Alerón b) Señal PWM Canal # 2 / Elevador.

Considerando un periodo de 19500 μ S, observado en el osciloscopio se puede decir que la frecuencia de la señal PWM es de 51,282 Hz al igual que en el canal anterior.

MÍNIMO

Para obtener el nivel mínimo del canal 2 o Elevador se procedió a ubicar la palanca del mando “Turnigy 9X” en la posición de la figura 68.b y se visualizó la señal PWM en el osciloscopio, la cual se muestra en la figura 68.a.



Figura 68. a) Posición mínima del Stick Elevador b) Señal PWM Canal # 2 / Elevador nivel mínimo (100% Abajo).

Considerando un periodo de 19500 μS y un ancho de pulso de 1440 μS medidos en el osciloscopio, se obtiene el valor del ciclo de trabajo a continuación.

$$t_{min} = 1040 \mu S$$

$$D_{min} = 100 * \frac{t}{T}$$

$$D_{min} = 100 * \frac{1040 \mu S}{19500 \mu S}$$

$$D_{min} = 5,33 \%$$

MEDIOS

Para obtener el nivel medio del canal 2 o Elevador se procedió a ubicar la palanca del mando “Turnigy 9X” en la posición de la Figura 69.b y se visualizó la señal PWM en el osciloscopio, la cual se muestra en la Figura 69.a.



Figura 69. a) Posición medio del Stick Elevador b) Señal PWM Canal # 2 / Elevador nivel medio (50 %).

Considerando un periodo de $19500 \mu\text{S}$ y un ancho de pulso de $1460 \mu\text{S}$ medidos en el osciloscopio, se obtiene el valor del ciclo de trabajo a continuación.

$$t_{med} = 1460 \mu\text{S}$$

$$D_{med} = 100 * \frac{t}{T}$$

$$D_{med} = 100 * \frac{1460 \mu\text{S}}{19500 \mu\text{S}}$$

$$D_{med} = 7,48 \%$$

MÁXIMOS.

Para obtener el nivel medio del canal 2 o Elevador se procedió a ubicar la palanca del mando “Turnigy 9X” en la posición de la figura 70.b y se visualizó la señal PWM en el osciloscopio, la cual se muestra en la figura 70.a.



Figura 70. a) Posición máxima del Stick Elevador b) Señal PWM Canal # 2 / Elevador nivel máximo (100% arriba).

Considerando un periodo de $19500 \mu\text{S}$ y un ancho de pulso de $1880 \mu\text{S}$ medidos en el osciloscopio, se obtiene el valor del ciclo de trabajo de $9,64\%$ al igual que el canal anterior.

Así, se define que el rango del ciclo de trabajo para el canal 2 / Elevador es desde $5,33\%$ mínimo hasta $9,64\%$ máximo, y con un rango de ancho de pulso de $1060 \mu\text{S}$ hasta $1880 \mu\text{S}$.

ACELERADOR/CANAL #3

Se muestra en la figura 71, el ciclo completo de la señal PWM del canal #3/Acelerador y a continuación sus características en niveles mínimos, medios y máximos.

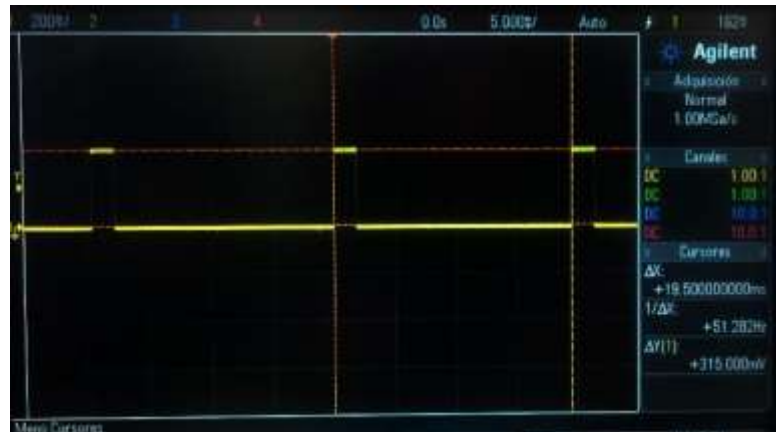


Figura 71. Señal PWM Canal # 3 / Acelerador.

Considerando un periodo de $19500 \mu\text{s}$, observado en el osciloscopio se permite decir que la frecuencia de la señal PWM es de $51,282 \text{ Hz}$ al igual que en el canal anterior.

MÍNIMO.

Para obtener el nivel mínimo del canal 2 o Elevador se procedió a ubicar la palanca del mando “Turnigy 9X” en la posición de la figura 72.b y se visualizó la señal PWM en el osciloscopio, la cual se muestra en la figura 72.a.



Figura 72. a) Posición mínima del Stick Acelerador b) Señal PWM Canal # 3 / Acelerador nivel mínimo (100% Abajo).

Considerando un periodo de $19500 \mu\text{S}$ y un ancho de pulso de $1040 \mu\text{S}$ medidos en el osciloscopio, se obtiene el valor del ciclo de trabajo a continuación.

$$t_{min} = 1040 \mu\text{S}$$

$$D_{min} = 100 * \frac{t}{T}$$

$$D_{min} = 100 * \frac{1040 \mu\text{S}}{19500 \mu\text{S}}$$

$$D_{min} = 5,33 \%$$

MEDIOS.

Para obtener el nivel medio del canal 2 o Elevador se procedió a ubicar la palanca del mando “Turnigy 9X” en la posición de la Figura 73.b y se visualizó la señal PWM en el osciloscopio, la cual se muestra en la Figura 73.a.



Figura 73. a) Posición media del Stick Acelerador b) Señal PWM Canal # 3 / Acelerador nivel medio (50%).

Considerando un periodo de $19500 \mu\text{S}$ y un ancho de pulso de $1480 \mu\text{S}$ medidos en el osciloscopio, se obtiene el valor del ciclo de trabajo a continuación.

$$t_{med} = 1480 \mu S$$

$$D_{med} = 100 * \frac{t}{T}$$

$$D_{med} = 100 * \frac{1480 \mu S}{19500 \mu S}$$

$$D_{med} = 7,58 \%$$

MÁXIMOS.

Para obtener el nivel máximo del canal 2 o Elevador se procedió a ubicar la palanca del mando “Turnigy 9X” en la posición de la figura 74.b y se visualizó la señal PWM en el osciloscopio, la cual se muestra en la Figura 74.a.



Figura 74. a) Posición máxima del Stick Acelerador b) Señal PWM Canal # 3 / Acelerador nivel máximo (100% Arriba).

Considerando un periodo de 19500 μs y un ancho de pulso de 1880 μs medidos en el osciloscopio, se obtiene el valor del ciclo de trabajo de 9,64% al igual que el canal anterior. Así, se define que el rango del ciclo de trabajo para el canal 2 / Acelerador es desde 5,33% mínimo hasta 9,64% máximo, y con un rango de ancho de pulso de 1060 μs hasta 1880 μs.

Para conseguir controlar los ejes del multirrotor se ha emulado mediante la tarjeta de control “Arduino MEGA 2560” las señales PWM originales del “Receptor Turnigy 9X” anteriormente descritas, como se muestra en la figura 75 y en la programación del anexo A.

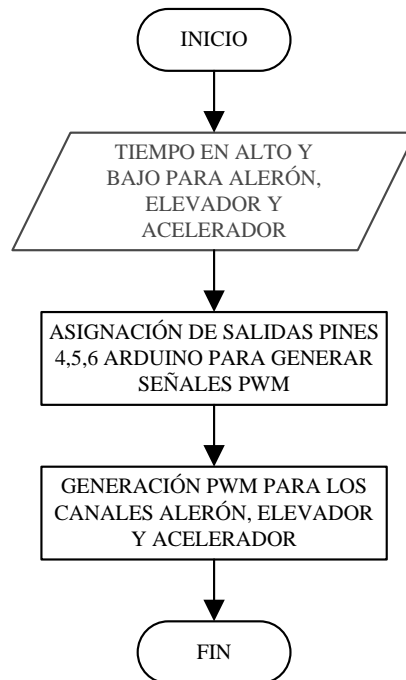


Figura 75. Diagrama de flujo programación de señales PWM.

De esta manera se genera señales PWM mediante la tarjeta “Arduino MEGA 2560”, a través de sus pines 4,5 y 6, con las características de sus homólogas procedentes del “Receptor Turnigy 9X”, para realizar el control de desplazamiento según los ejes del multirrotor, hay que mencionar que se ha generado las señales PWM mediante los tiempos en alto y en bajo de cada canal, se ha utilizado la instrucción “digitalwrite(H/L)” para la activación de la señal digital en los pines de la tarjeta arduino y la instrucción “delayMicroseconds(Time)” para establecer el ancho de pulso en microsegundos, el canal 4 correspondiente al “Rudder/Timón” y canal 5 establecido como “Modo de Vuelo” no se utilizan, ya que el modo de vuelo establecido es

“GPS/Atti” que permite mantener el multirrotor en una posición y a una altura establecida mientras realiza las actividades pertinentes en el aire relacionadas al proyecto, así también no se requiere el control del eje “YAW”, ya que el multirrotor permanecería con la nariz en una misma dirección sin cambio alguno.

Así también se ha efectuado un sistema de cambio entre las señales originales que envía el aparato de recepción y transmisión “Turnigy 9X” y sus similares generadas por la tarjeta “Arduino MEGA 2560”, con el objetivo de permitir un funcionamiento a partir de control manual (operador humano) y un control autónomo por medio de las señales creadas por el algoritmo de visión artificial (entre la tarjeta arduino y la “PC Intel”), como se muestra en el flujograma de la figura 76 y en el código de programación del anexo B.

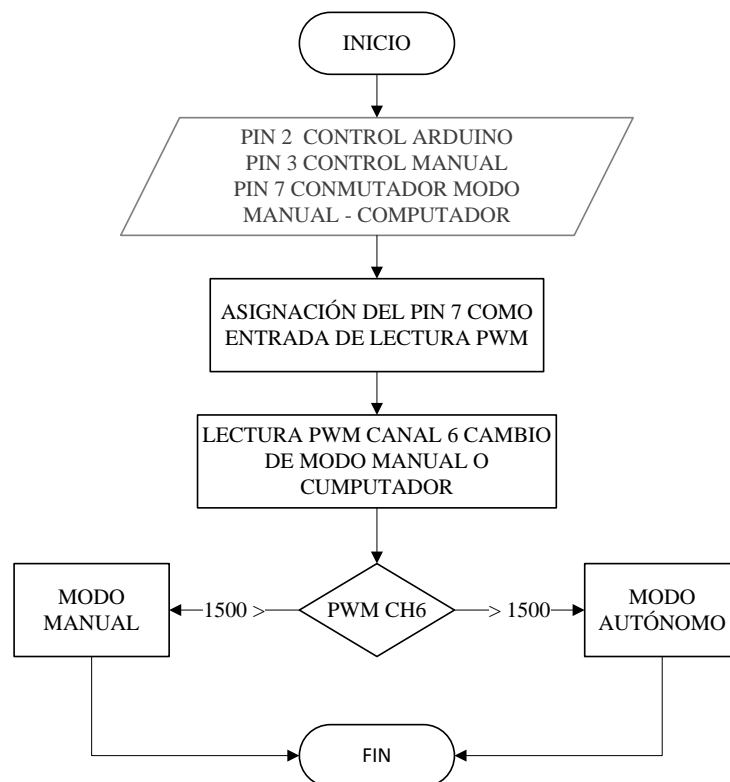


Figura 76. Diagrama de flujo de la programación para el control de los módulos de relé para el cambio entre el codo manual y el autónomo.

Para llevar a cabo la tarea de intercambiar entre el control manual y autónomo se utilizó los relés de estado sólido “OMROM SSR G3MB-202P”, debido a que cada relé actúa como interruptor, se utilizó un arreglo que permite juntar dos relés y convertirlos en un conmutador para las señales PWM originales y las emuladas por la tarjeta arduino que tienen en común la conexión hacia la tarjeta controladora de vuelo, como se muestra en la figura 77, permitiendo así que las señales lleguen a la tarjeta controladora de vuelo “Naza M Lite”, se utilizó 6 relés que intercambian las señales PWM entre el “Receptor Turnigy 9X” y las señales generadas a través de la programación anterior mediante la placa “Arduino MEGA 2560”.

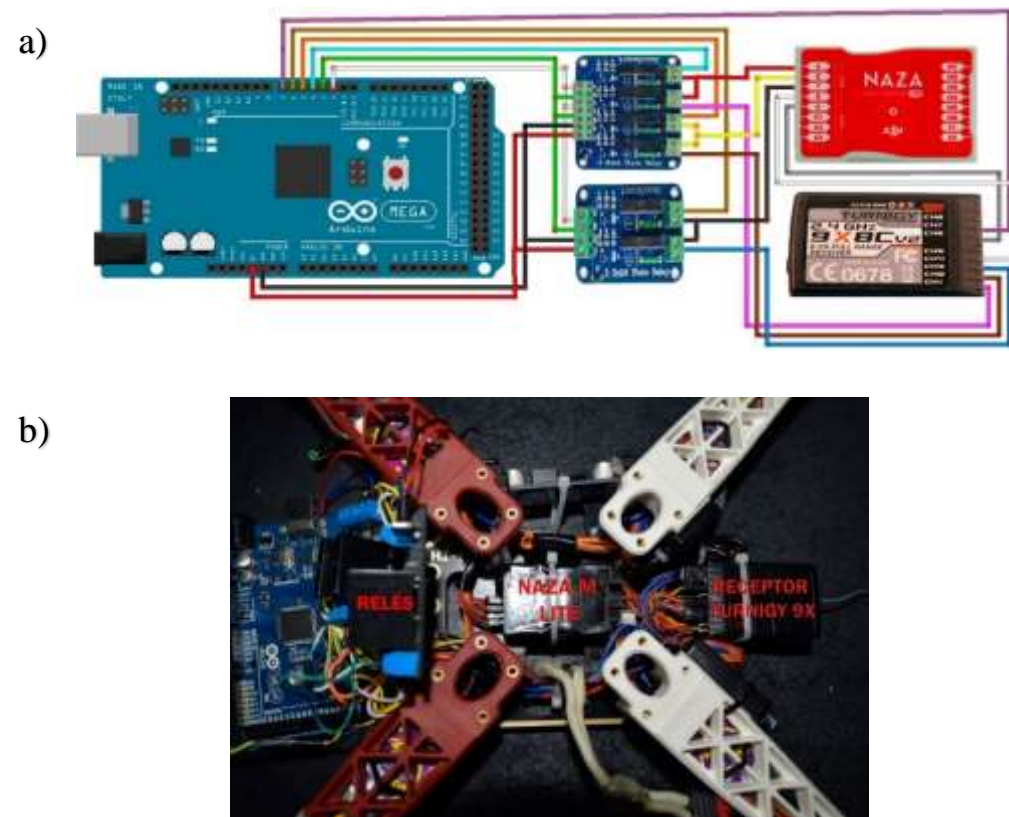


Figura 77. a) Conexiones entre el módulo de relés, la tarjeta arduino y la controladora de vuelo b) Implementación del sistema de relés.

3.2.2 Sensor ultrasónico HC-SR04

El sensor ultrasónico “HC-SR04” ofrece tres formas de obtener sus datos de distancia, entre las cuales se encuentran la comunicación TTL, lectura por ancho de pulso PWM o lectura análoga, para el proyecto se ha optado por la lectura análoga, realizada como se muestra en la figura 78 y en la programación del Anexo C.

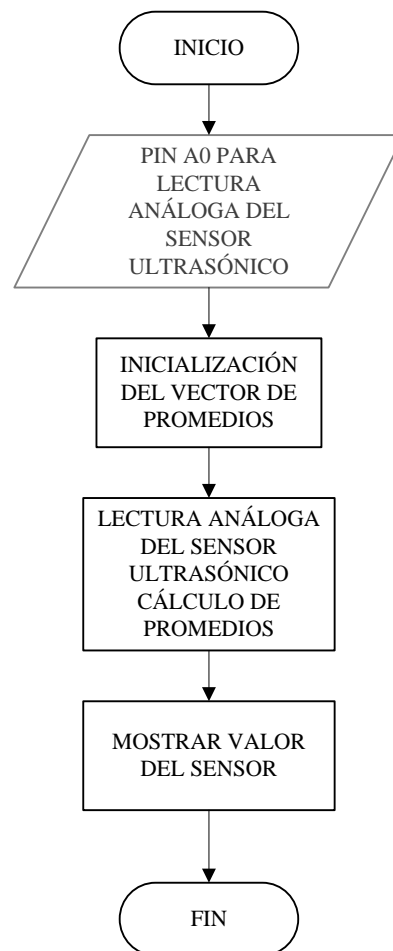


Figura 78. Diagrama de flujo para la lectura del sensor ultrasónico HC-SR04.

Donde, se ha realizado la lectura de los datos mediante el comando “analogRead (A0)” y un cálculo de promedios de lecturas análogas realizadas del sensor ultrasónico, mediante un vector que recoge diez muestras de las lecturas en un lapso de tiempo, se

ha considerado este número de muestras como valor óptimo para evitar interrumpir el desarrollo de procesos que se ejecutan durante el sistema autónomo como se muestra a continuación en la tabla 40.

Tabla 40.

Cálculo de promedios del sensor ultrasónico para medir la altura del dron.

$t = 1 \mu S$	Vector de Muestras	Promedio
V0	30 mm	$P = \frac{\sum_{n=0}^9 V_n}{9}$
V1	32 mm	
V2	31 mm	$P = \frac{315}{9}$
V3	30 mm	$P = 35 \text{ mm}$
V4	33 mm	
V5	35 mm	
V6	33 mm	
V7	31 mm	
V8	30 mm	
V9	30 mm	

Donde, para obtener una lectura más eficiente de datos de la altura real del dron, se ha implementado un sistema de promedios mediante un vector que recoge 10 muestras en un lapso de tiempo de $1\mu S$, y se calcula de forma convencional el promedio de esas lecturas, así mientras avanza el tiempo en el programa, los valores del vector continúan cambiando para cálculos posteriores.

Así también, para obtener una mayor fidelidad de los datos de distancia cuando el multirrotor se encuentre en el aire, se ha realizado un acondicionamiento de datos para adquirir una lectura en milímetros, la que se muestra en la figura 79.

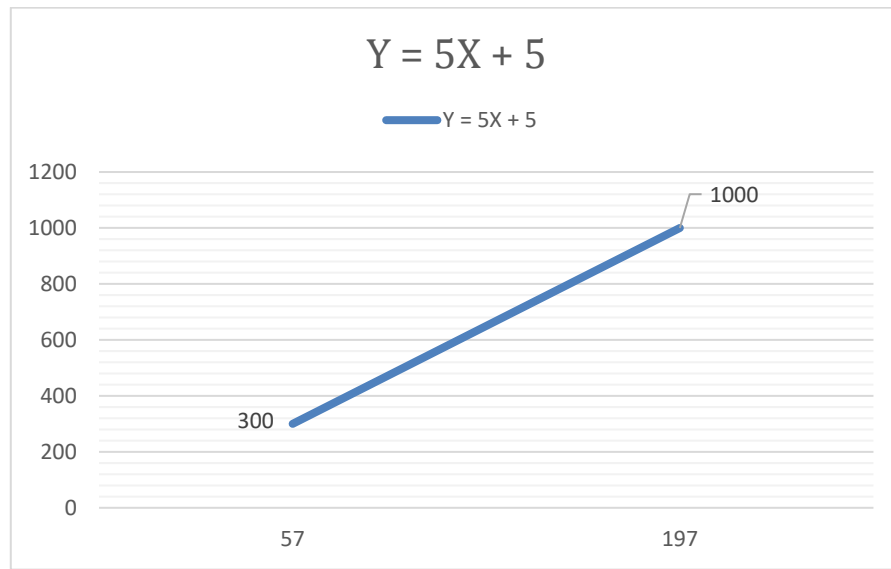


Figura 79. Grafica de acondicionamiento de datos del sensor ultrasónico HC-SR04

$$m = \frac{Y_2 - Y_1}{X_2 - X_1} \qquad Y = mx + b$$

$$m = \frac{1000 - 300}{197 - 57} \qquad 300 = 5(57) + b$$

$$m = 5 \qquad b = 15$$

$$Y = 5X + 15$$

$$\mathbf{Altura = 5 * (Lectura A0) + 15}$$

Así también, su implementación se muestra en la figura 80, donde se ha utilizado el pin A0 para la lectura de los datos análogos y su correspondiente alimentación de 5VDC.

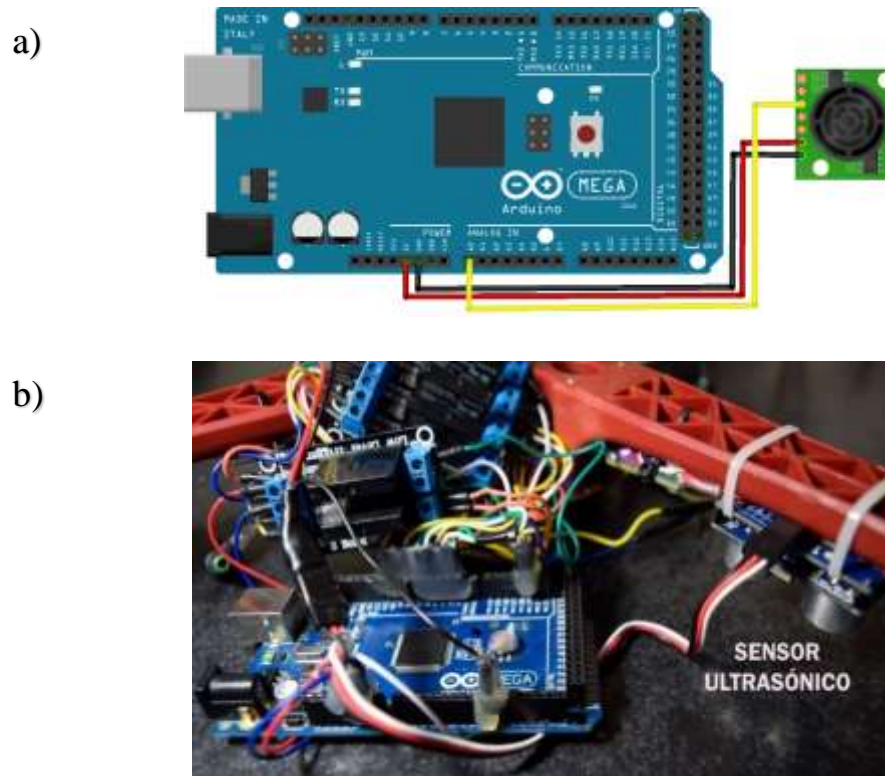


Figura 80. a) Conexiones del sensor ultrasónico HC – SR04 b) Implementación del sensor ultrasónico HC-SR04 en el multirrotor.

3.2.3 Módulo GPS GY-GPS6MV2

El módulo “GPS GY-GPS6MV2” ha sido elegido para obtener la posición del multirrotor y poder establecer una ruta de donde se encuentra la zona del helipad, para luego pasar al procesamiento e identificación de imágenes mediante la red neuronal artificial y posterior aterrizaje, su programación se detalla en la figura 81 y en la programación del anexo D.

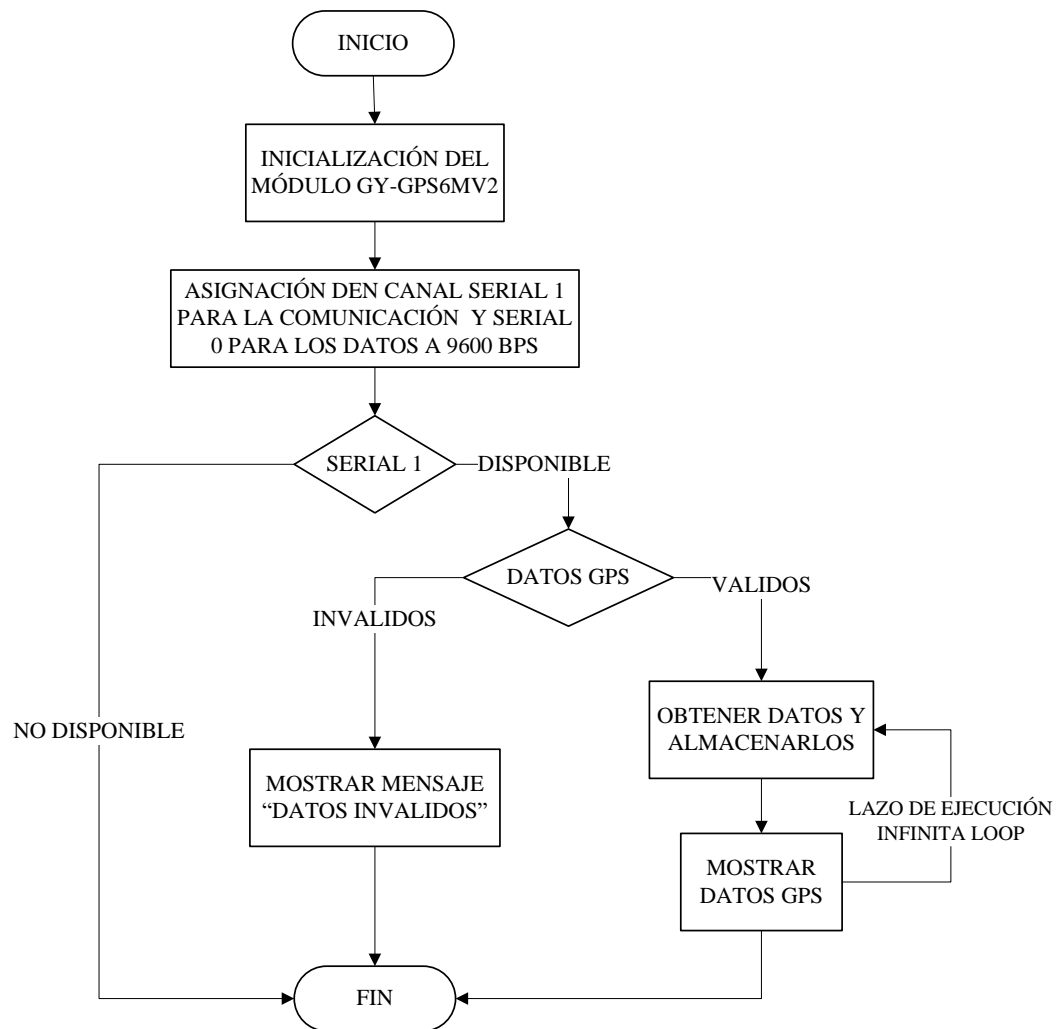


Figura 81. Diagrama de flujo de la programación para obtener los datos del módulo GPS GY-GPS6MV2.

Donde, los datos proporcionados por el módulo GPS se encuentran en coordenadas decimales, serán de gran utilidad cuando se establezca la ruta que el multirrotor debe alcanzar, la implementación del módulo GPS GY-GPS6MV2 se muestra en la figura 82.

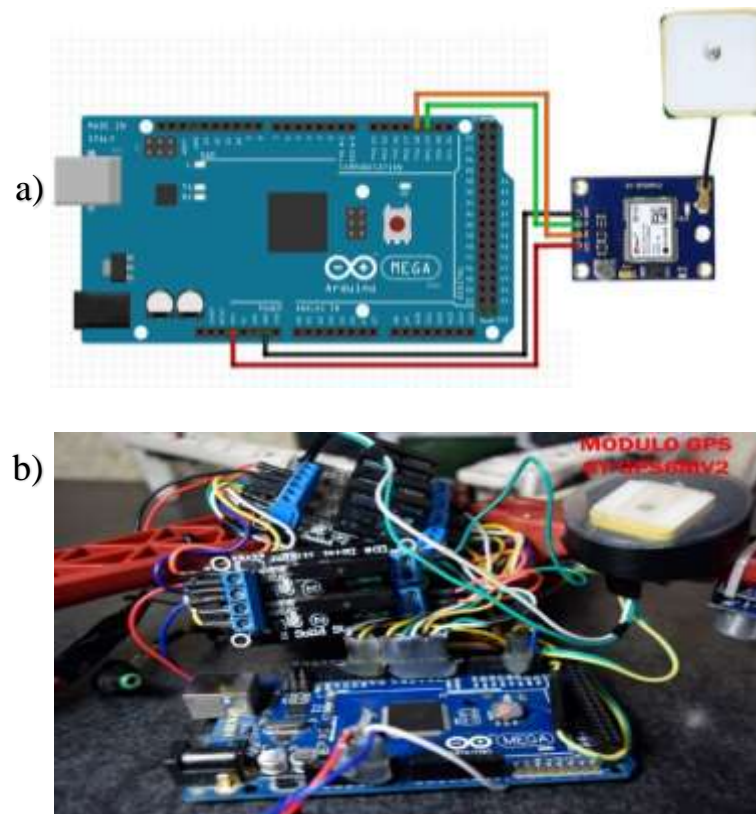


Figura 82. a) Conexiones del módulo GPS-GY6MV2 b) Implementación del módulo GPS GY-GPS6MV2 en el multirrotor.

3.2.4 Comunicación Arduino – PC (Red Neuronal Artificial)

Para poder controlar la dirección en la que se mueve el multirrotor una vez que ha alcanzado las coordenadas GPS, se ha establecido una serie de comandos mediante la comunicación serial en el canal 1 del arduino “MEGA 2560”, mediante la cual en la programación en Visual Studio, que es donde se desarrolló la red neuronal artificial de la cual se hablara más adelante, podrá enviar estos comando para dirigir hacia donde se debe mover el multirrotor para alcanzar la posición del helipad, como se muestra en la figura 83, se detalla su programación en el Anexo E.

Donde, se ha utilizado un vector llamado “inData [i]” para el almacenamiento de los caracteres (códigos) enviados desde le computador a través de la red neuronal

artificial, en la figura 84 se detalla la trama serial correspondiente al envío de un código de dirección para el posicionamiento del multirotor, los significados de cada código de direccionamiento se detallan a en la tabla 41.

Tabla 41.

Códigos de posicionamiento neuronal PC – Arduino, canales Alerón y Elevador.

Códigos de dirección del multirotor hacia el helipad		
	Código	Dirección
ALERÓN	A	20 % Derecha
	B	15 % Derecha
	C	10 % Derecha
	D	5 % Derecha
	E	Estacionario
	F	5 % Izquierda
	G	10 % Izquierda
	H	15 % Izquierda
	I	20 % Izquierda
	ELEVADOR	J
K		15 % Adelante
L		10 % Adelante
M		5 % Adelante
N		Estacionario
O		5 % Atrás
P		10 % Atrás
Q		15 % Atrás
R		20 % Atrás
BUSCADOR		5
ATERRIZAJE	T	Activa el aterrizaje autónomo

Así, si el multirrotor por ejemplo se encuentra muy alejado a la izquierda/atrás del helipad, la red neuronal artificial enviara el código “AJ” para que el multirrotor se mueva hacia la dirección derecha/adelante hasta alcanzar la posición del helipad, enviando el código “EN”, que significa mantener la posición en la que se encuentra para su posterior aterrizaje.

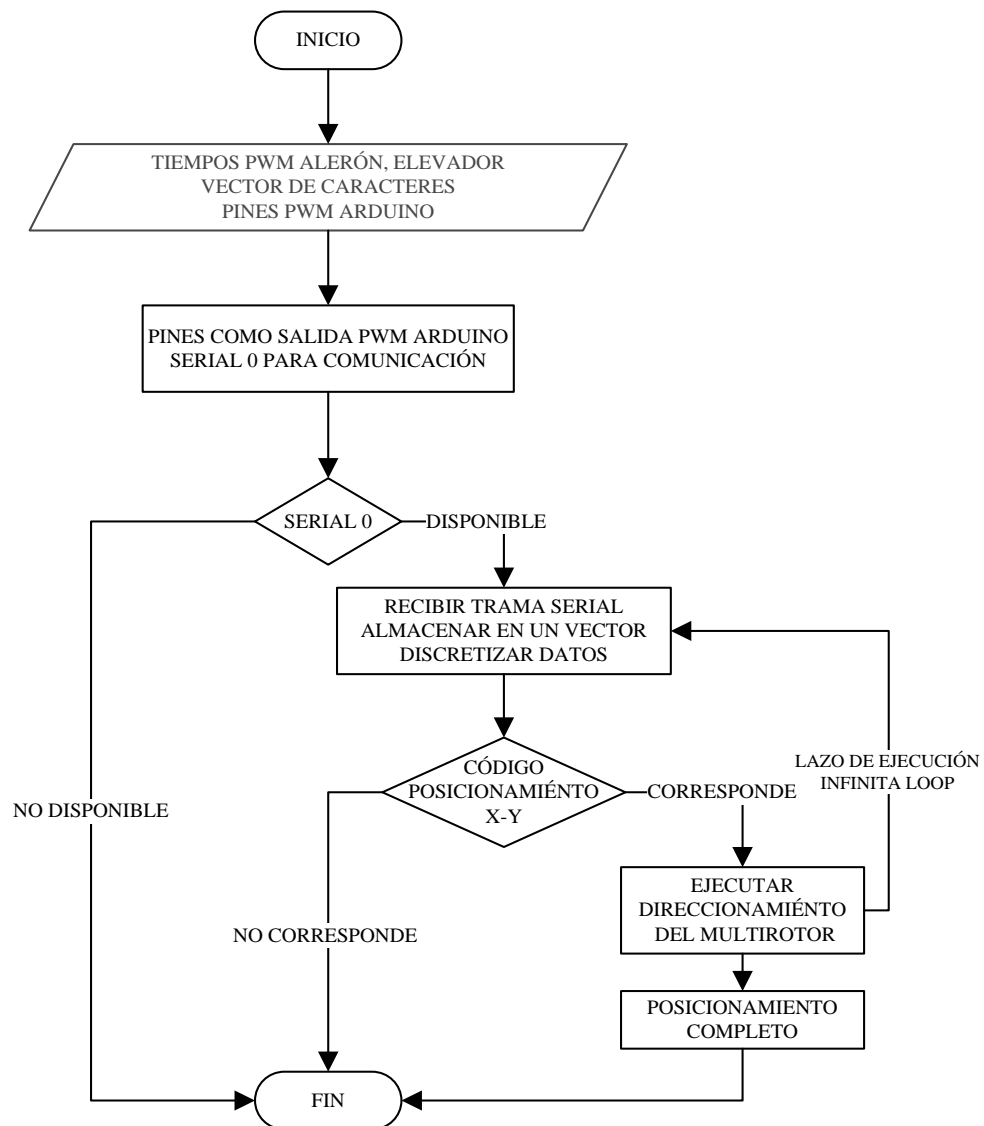


Figura 83. Programación para la comunicación Arduino – PC

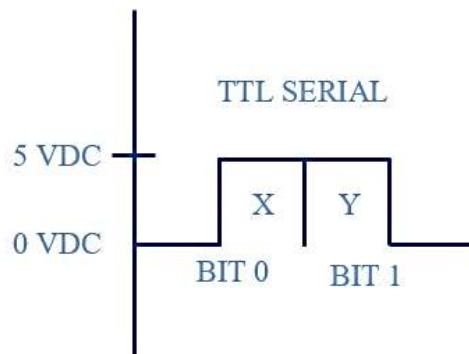


Figura 84. Trama serial de la comunicación arduino – red neuronal artificial.

3.2.5 Secuencia Autónoma

La secuencia autónoma establecida para que el multirrotor realice la misión se ha implementado así, el dron realiza el despegue, seguido de un control de altura, prosigue la ruta GPS establecida hasta alcanzar la zona donde posiblemente se encuentra el helipad, se realiza la identificación mediante la red neuronal y los métodos de visión artificial que permiten identificarlo, luego el multirrotor se ubica sobre el helipad ya identificado y se realiza el aterrizaje.

Para ello la red neuronal envía el código “T” a través del puerto serial, indicando a la tarjeta arduino que se completó el reconocimiento y el multirrotor se encuentra sobre el punto de aterrizaje, como se muestra en la figura 85, la programación se detalla en el anexo F.

La secuencia autónoma se ha dividido en rutinas que se ejecutan una a continuación de la otra, desactivando la anterior para que no se vuelva a ejecutar por error.

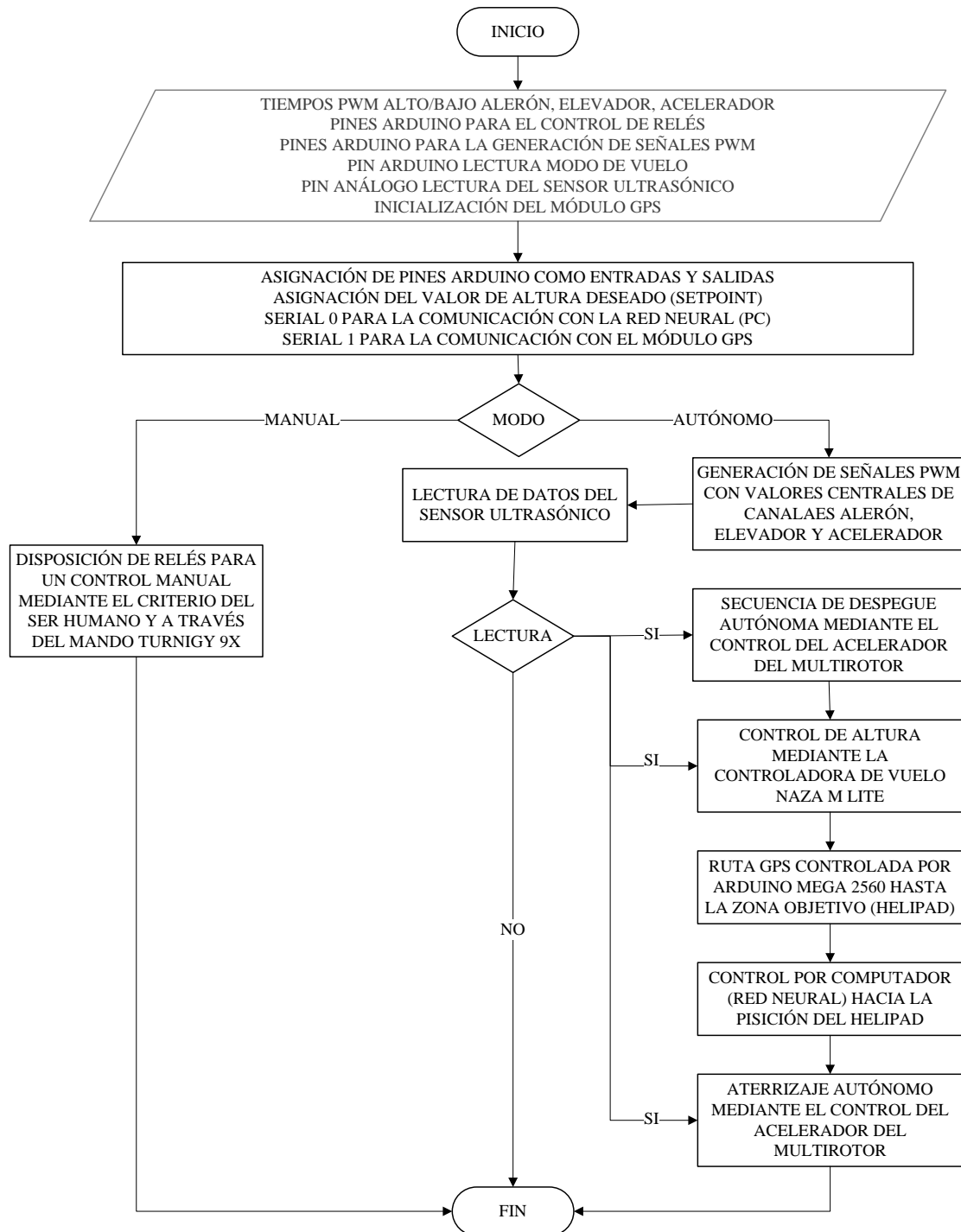


Figura 85. Secuencia autónoma.

c) Control de Altura

Una vez realizado el despegue, la sub rutina “Control_Altura()” se encarga de mantener el valor del acelerador de la aeronave en un valor de tiempo en alto de 1450 μ S, cuando alcance el setpoint de altura que es 2500 mm (2.5m), ello significa que los motores se mantendrán girando a una misma velocidad constante y el multirrotor no aumentará ni disminuirá su altura, esto gracias al controlador PID interno de la controladora de vuelo “NAZA M Lite”, al ejecutarse esta acción se desactiva la subrutina anterior de despegue, como se muestra en la figura 87 y en la programación del anexo I.

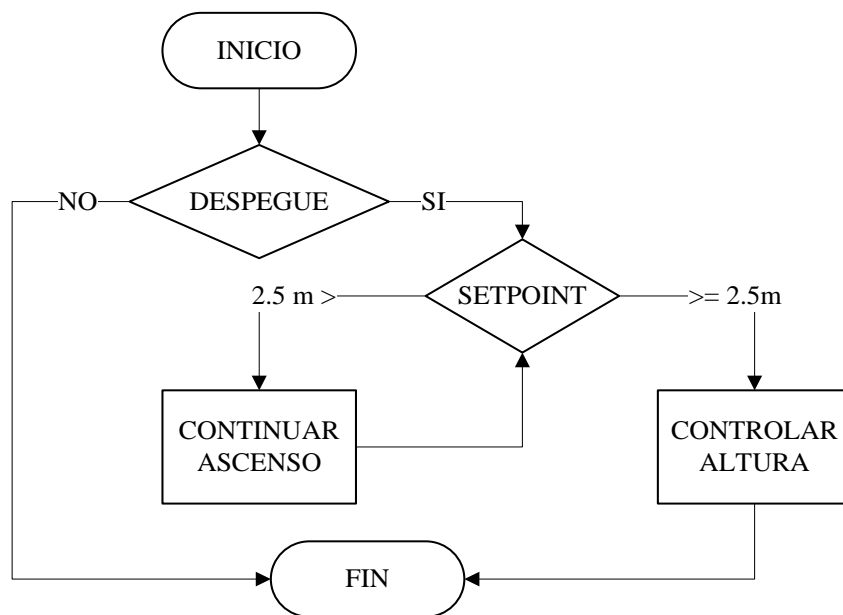


Figura 87. Diagrama de flujo para el control de altura del multirrotor.

d) Ruta GPS

Una vez establecido el control de altura, la tarjeta “Arduino MEGA2560” compara la posición actual del multirrotor y la posición objetivo de la zona de aterrizaje y calcula la distancia (error) tanto en latitud (X), como en longitud (Y), permitiendo el desplazamiento del multirrotor en los ejes “ROLL” y “PITCH”, gracias al control de los canales “Alerón” y “Elevador” mediante los tiempos en alto y en bajo, para esta tarea se ha establecido un valor de un 20% de aceleración de cada canal, como se muestra en la figura 88 y en la programación del anexo J.

e) Comunicación con la Red Neuronal Artificial (PC)

Al completarse la ruta GPS, se permite el acceso al control por computador mediante la red neuronal artificial programada en “Visual Studio”, como se explicó anteriormente en la figura 83, haciendo uso de los códigos que se observa en la tabla 3.2, la tarjeta “Arduino MEGA 2560” envía el código “5” que hace que se active el buscador de la red neuronal, para así localizar el helipad mediante la identificación de imágenes

Entonces la red posiciona el multirrotor sobre el helipad y cuando la posición es la correcta, se envía el código “T” desde el computador, que activa la secuencia de aterrizaje, como se muestra en la programación del anexo K.

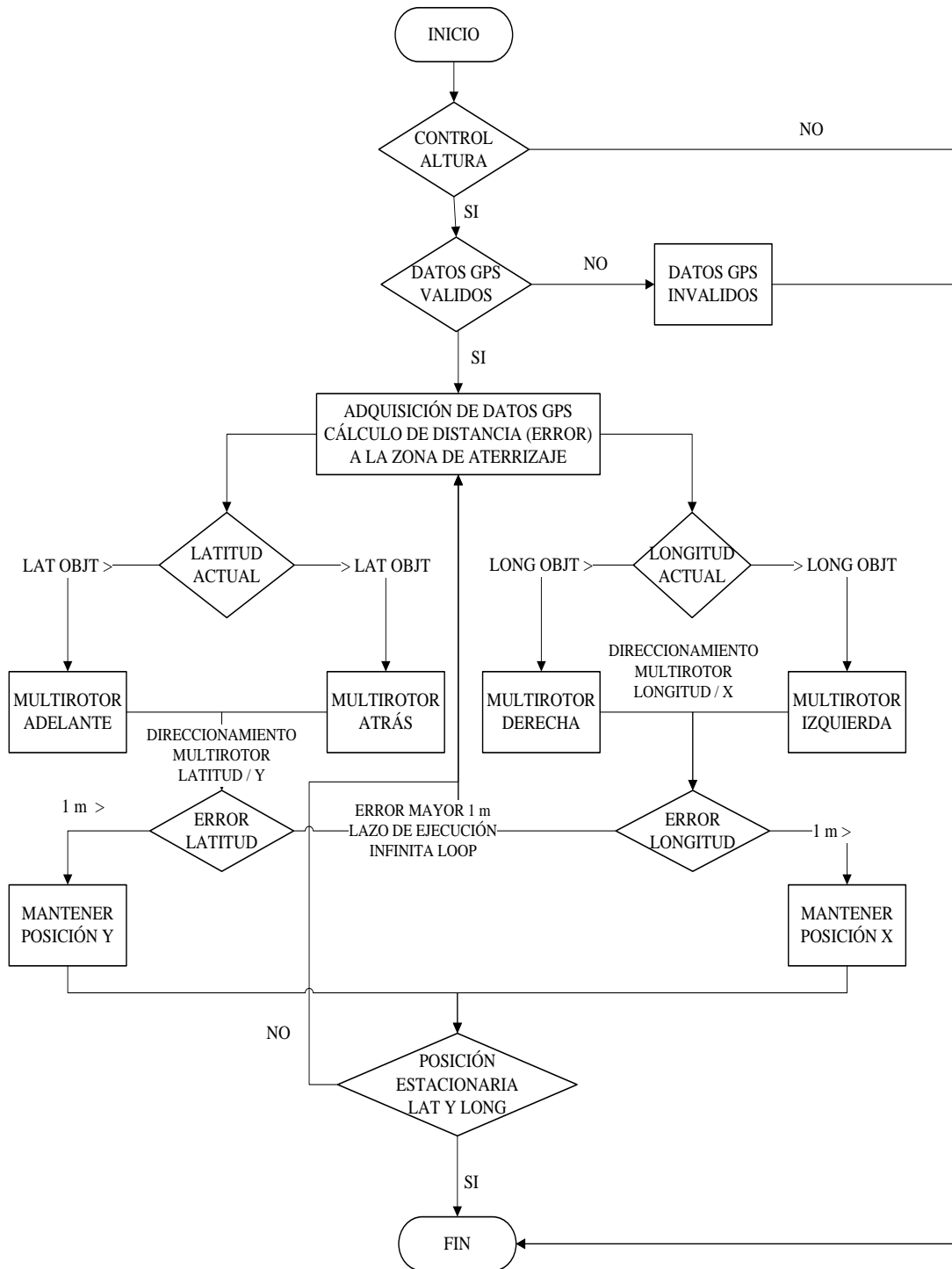


Figura 88. Ruta GPS.

f) Aterrizaje

Cuando el multirrotor se ha aproximado al helipad previamente identificado y la distancia de error en la pantalla de la RNA posee un error menor a 35 píxeles, entonces envía el código “T” desde el computador, que ejecuta la secuencia de aterrizaje autónomo, mediante el control del tiempo en alto del acelerador, con un valor de 1399 μ S correspondiente al 44 % de aceleración, obtenido de pruebas experimentales en modo manual, observando el porcentaje de activación del Stick del canal de aceleración, hasta encontrar el valor óptimo suficiente para que la aeronave realice un descenso controlado, una vez que se ha ejecutado esta secuencia se desactiva la comunicación con la red neuronal y los valores de los canales “Alerón” y “Elevador” permanecen en sus valores centrales para conseguir un descenso netamente vertical, así cuando el sensor ultrasónico detecta una altura menor a 30 mm se apagan los motores, completando el proceso, como se muestra en la figura 89, y en la programación en el anexo L.

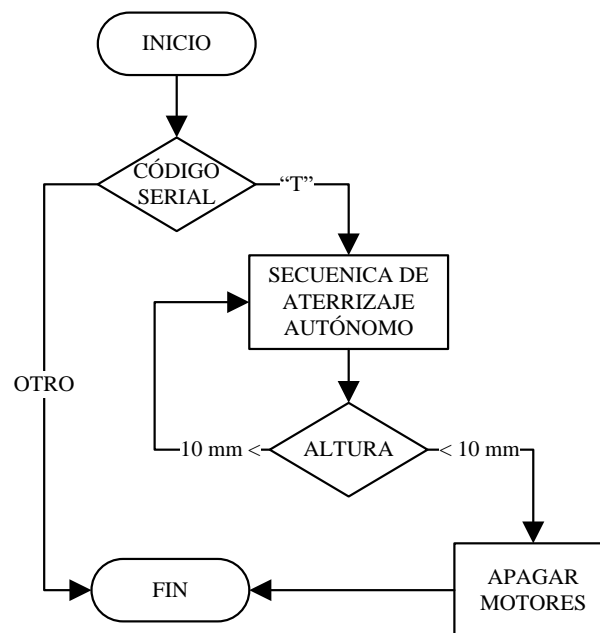


Figura 89. Diagrama de flujo de la secuencia autónoma de aterrizaje.

3.3 Programación de la red neuronal artificial.

Para la realización de la búsqueda se usará un sistema basado en redes neuronales artificiales, desarrollado por el doctor Oscar Chang, Este sistema se alimentará de una imagen que ingresará a la red neuronal artificial permitiendo identificar las características de la misma y de esta manera poder reconocer los patrones deseados, así también, poder identificar un grupo de píxeles para ser rastreados por el sistema de Lucas Kanade.

Lucas Kanade es un predictor de flujo óptico, que identifica los bordes de una imagen, en este caso lo que se encuentre dentro de la ROI, se trasfiere a la imagen capturada desde la cámara para que siga el desplazamiento del helipad, esto se puede realizar ya que el movimiento de píxeles entre imágenes capturadas de la cámara tiene una variación constante con el uso del predictor se ubica el centro de la imagen capturada que ha identificado como el objeto de interés y permite saber de qué parte de la imagen atrapada por la cámara se encuentra.

Para el desarrollo de software se han realizado diferentes programas, el primero es un entrenador para la primera RNA, El segundo es del entrenador de la segunda RNA, el tercero es el que unirá las dos redes neuronales y permitirá además incorporar el sistema de Lucas Kanade y enviar los datos necesarios al arduino que es la tarjeta encargada de la comunicación entre el computador y el DRONE.

Para el desarrollo de los distintos programas se realiza una ROI de 100 x 100 píxeles la misma que captura una parte de la imagen para alimentar a las redes neuronales artificiales, esta ROI recibe datos binarizados (0 1).

3.3.1 Entrenador de la primera red neuronal.

El primer programa es un entrenador para la primera red neuronal artificial, este programa permite entrenar a la red neuronal artificial para identificar la forma de la figura deseada sin importar algunas variaciones tales como el tamaño, giro de la imagen, posición con respecto al centro de la ROI, la misma que dispara una neurona de salida indicando la posición. Hay que considerar que esta red neuronal artificial se confunde con facilidad entre figuras y el ruido, lo que ocasiona un seguimiento de cualquier figura con una densidad significativa de píxeles.

Usando las librerías de OpenCV, se captura una imagen de tipo JPG o también puede capturar una imagen en tiempo real desde una cámara, esta imagen contendrá ejemplos de los objetivos o figuras con las que se desea entrenar, para esta primera red neuronal artificial es mejor usar una imagen que una captura de video en tiempo real, puesto que esto permite tener una mejor base de datos, admitiendo que la imagen sea más estable y de esta manera también evite el ruido que afecta al entrenamiento. Ver figura 91

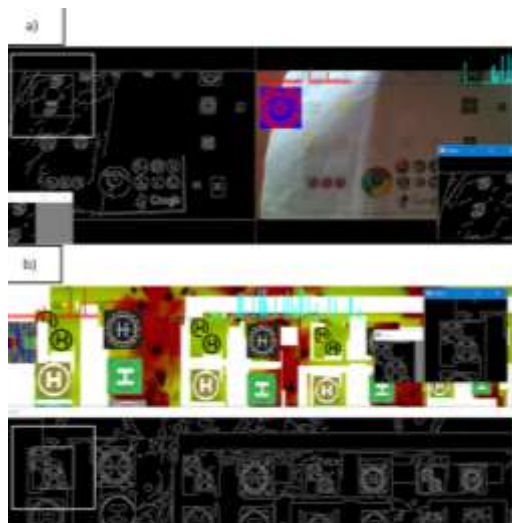


Figura 90. Adquisición de imágenes usando openCV A) adquisición en tiempo real. B) adquisición desde una base de datos.

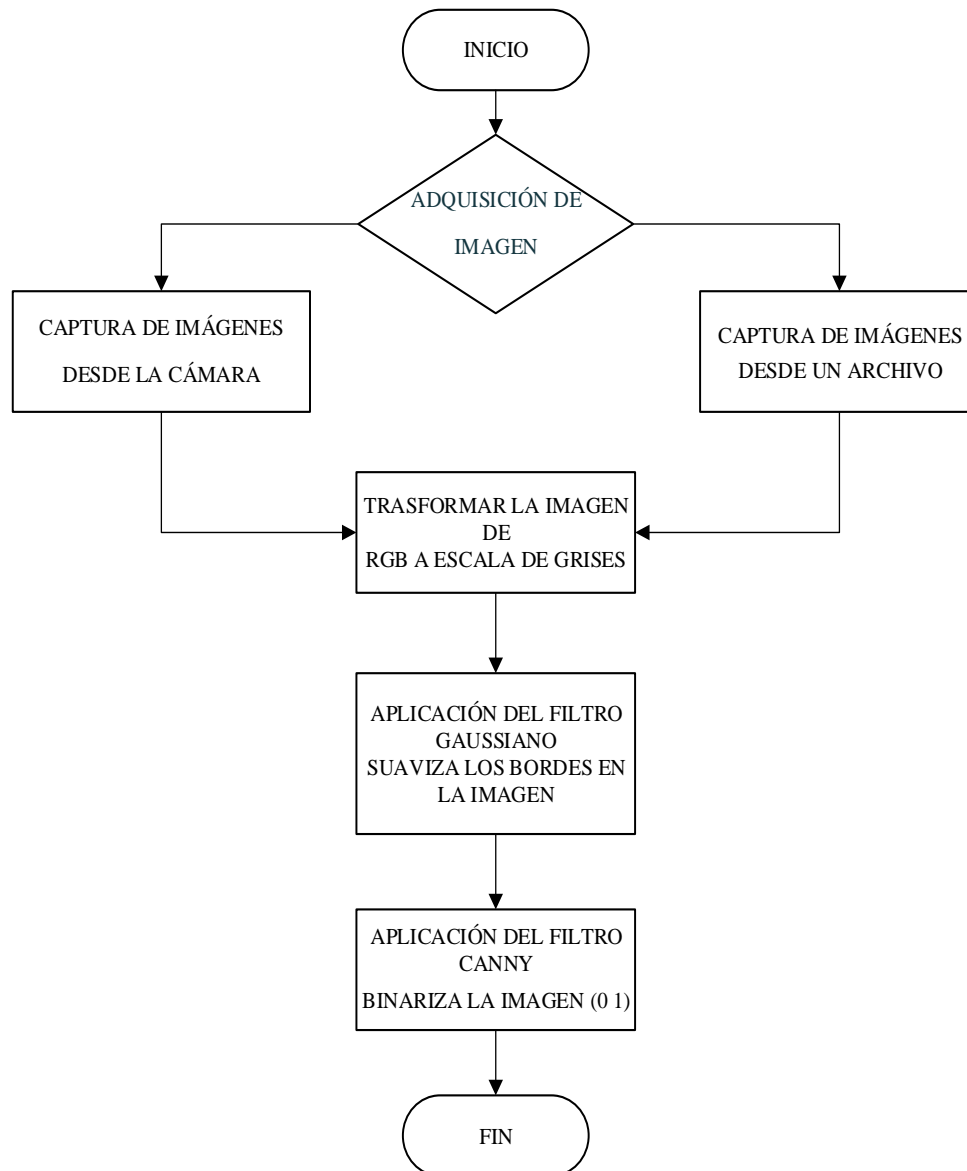


Figura 91. Diagrama de flujo, secuencia de binarización de imagen.

Luego de la adquisición de la imagen, se procede a implementar el algoritmo de CANNY, este algoritmo permite reducir la información que alimenta la red y mejora de esta manera la velocidad en los procesos, así como también los resultados finales; CANNY realiza la binarización de la imagen (imagen en blanco y negro donde solo

aparecen los bordes de la escena), en la figura 91, se observa el diagrama de flujo donde se captura la imagen de la cámara o un archivo y se procede a aplicar el filtro de CANNY. Ver figura 90.

Para el entrenamiento de la RNA se considera la captura de una imagen en la que se aplica el algoritmo de CANNY, esta imagen de 200 x 200 píxeles es la entrada de la ROI, donde se realiza desplazamientos del helipad en “x” e “y” con una variación de 1 píxel desde el centro hasta 21 píxeles hacia la derecha, arriba, izquierda y abajo como se muestra en la figura 92, así mismo se utiliza una Roi de 100 x100 píxeles ubicada en el centro de la imagen de 200 x 200, desde donde se observa como el helipad cambia de posición en “x” e “y”, así la Red neuronal artificial aprenderá a reconocer diversas formas de helipuertos para una mejor identificación del mismo.



Figura 92. En la sub ventana de color verde se observa la imagen capturada y girada (200x200 pixeles) mientras que en la ventana de color blanco se visualiza la imagen ROI (100x100 pixeles capturada de la imagen de 200 x 200).

En este programa se procede a implementar una red neuronal artificial de tres capas, una capa de entrada con 10 000 neuronas, una capa escondida 100 neuronas y 42 neuronas en la capa de salida, esto permite tener un rango de compresión de la información que ingresa y tener un buen resultado a la salida, para facilitar la

implementación de la red neuronal se usan los encabezados facilitados por el Dr. Oscar Chang.

Para el entrenamiento de la RNA se usa el método de la propagación hacia atrás de errores, tomando como ejemplo la imagen capturada por la ROI de 100 x 100 y para la salida, las 42 posiciones en que se puede mover a la imagen, 21 en “x” y 21 en “y” Ver tabla 42, lo primero que se realiza para ingresar a la red es tomar cada uno de los puntos de la ROI (imagen de 100 x 100) y lo ingresa en un vector que está formado por los valores de cada fila una a continuación de la otra hasta completar las 10 000, estos valores se los ingresa a la primera capa de la red neuronal artificial.

Tabla 42.

Relación entre las neuronas de salida y la posición del centro de la ROI.

posiciones ROI en “x”																				
10	9	8	7	6	5	4	3	2	1	0	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10
neuronas de salida																				
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
posiciones ROI en “y”																				
10	9	8	7	6	5	4	3	2	1	0	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10
neuronas de salida																				
22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42

La salida de la red (42 neuronas) la compara con la posición de la imagen ROI, se procede al entrenamiento automático hasta conseguir que la salida de la red active las neuronas que representan la posición de la imagen en “x” e “y” obsérvese la tabla 3.3 y como se muestra en la figura 93, para el entrenamiento se lo realiza primero sin cambiar ningún parámetro, luego cambiando un solo parámetro, sea este el tamaño o su orientación (giro) y luego con todos los parámetros que intervienen en este entrenamiento, es necesario el incremento de variaciones en los parámetros para evitar

los mínimos locales producidos por el ingreso de demasiados datos de una sola vez.
Ver figura 93.

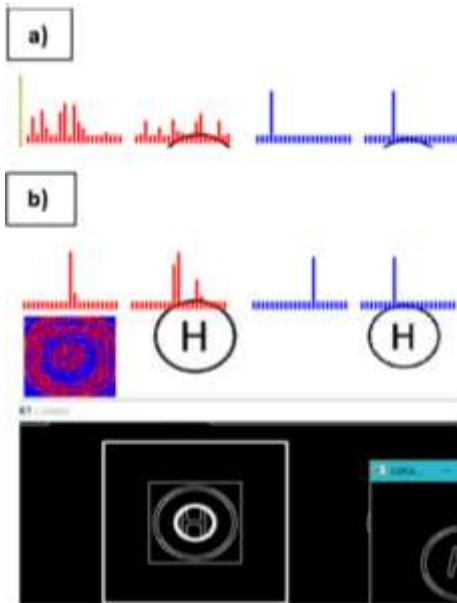


Figura 93. Entrenamiento de la RNA salidas vs posición de la imagen (targets)
A) muestra la respuesta de las neuronas sin entrenar (listones rojos) y en B)
muestra a la red entrenada (Listones rojos).

En la figura 93 se observa la posición de la imagen ROI con listones de color azul y la salida de la RNA con barras de color rojo, en la parte A se observa la reacción obtenida antes del entrenamiento, mientras que en la parte B se observa como las salidas de la red (listones de color rojo) se asemejan a la posición de la imagen ROI (listones de color azul), al término del entrenamiento las salidas de la red copia casi con precisión la posición de la imagen sin importar variaciones en tamaño, cambios en la forma y posición angular.

Para realizar este seguimiento se usa la neurona de salida ganadora (barra de color roja de mayor altura), para posicionar la ROI (“x_roi” e “y_roi”), el entrenamiento de la red neuronal artificial hace que los pesos de conexión entre neuronas cambien, estos valores se almacenan, ya que, es el conocimiento adquirido por la red neuronal artificial.

Para que el sistema de identificación sea más eficiente, se le ha entrenado la red neuronal artificial con ejemplos libres de ruido y perturbaciones, en las imágenes objetivos (helipuertos), para ello se realizó un programa entrenador como se muestra en el flujo grama de la figura 94.

El entrenamiento ha llevado varias horas de aprendizaje, ya que, para conseguir un amplio conocimiento, fue necesario enseñarle valor aproximado de ejemplos, así la capacidad de inferencia de la red neuronal artificial en el mundo se incrementa notablemente, identificando al objetivo (helipad) en ambientes totalmente desconocidos para la misma.

Hasta este punto se ha logrado que la red neuronal artificial mueva la ROI por la imagen principal en busca del helipad, la RNA puede seguir un objeto (figura) con gran velocidad de respuesta y eficiencia en el procesamiento de datos, pero presenta facilidad de confusión con otros objetos similares.

En la figura 95 se observa la imagen de entrada a la derecha, mientras que a la izquierda se puede ver la imagen después de haber sido aplicado CANNY, también se puede visualizar la posición de la ROI y la pre ROI(imagen de 200 por 200 pixeles que se usa para el entrenamiento), en la esquina inferior izquierda se visualiza la ROI que alimenta a la red neuronal artificial, y sobre la imagen a colores se ve las salidas de la RNA como: barras en color rojo reaccionando a los datos capturados por la ROI

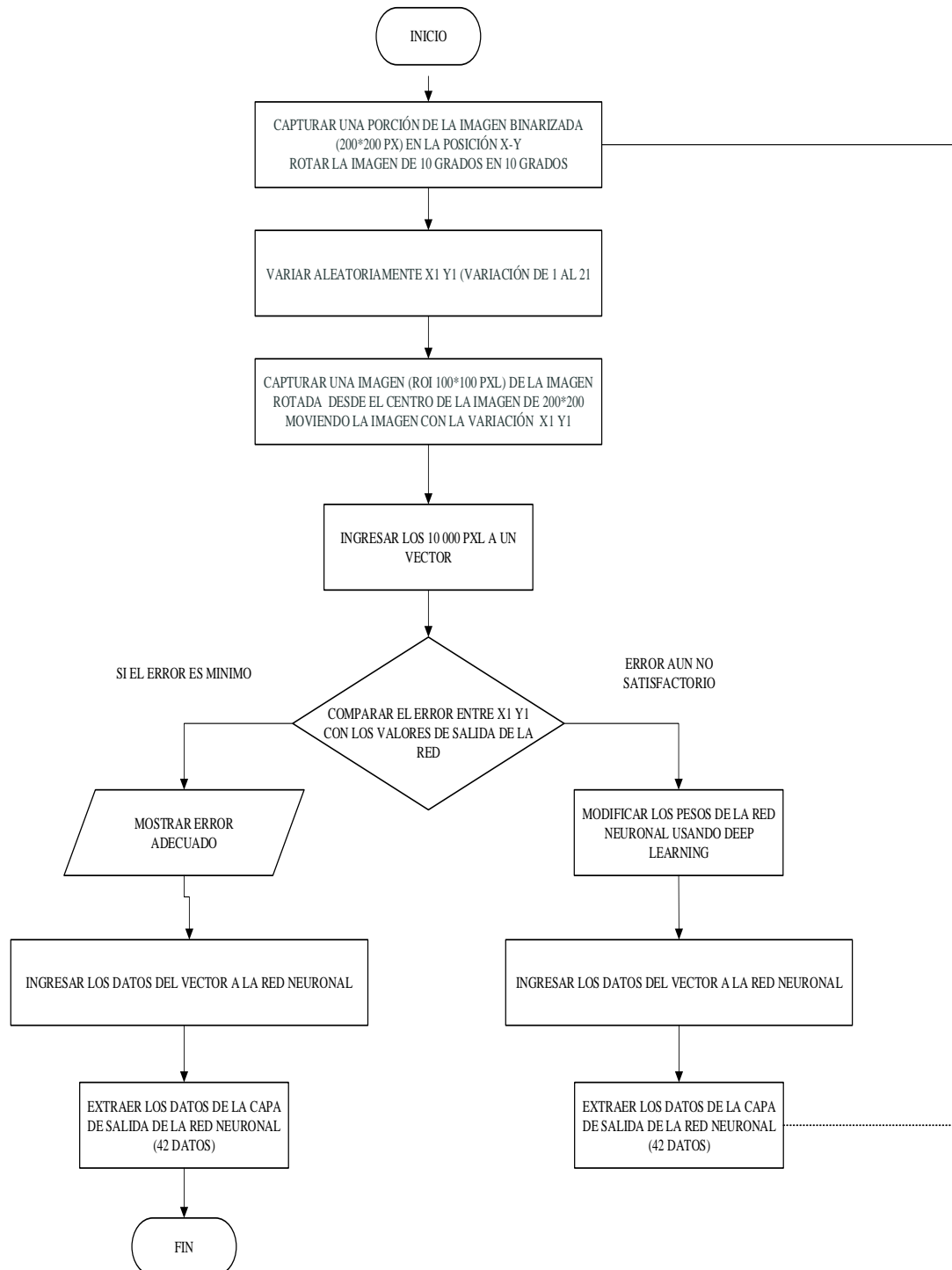


Figura 94. Diagrama de flujo, secuencia de entrenamiento de la primera red neuronal artificial.



Figura 95. Entrador de la primera red.

3.3.2 Entrenador de la segunda red neuronal.

Este programa está diseñado para el entrenamiento de la segunda red neuronal artificial (RNA), como se explica a continuación y se muestra en el flujograma de la figura 96, esta RNA se usa para analizar el patrón de movimiento de la figura dentro de la ROI, si es la figura deseada (helipad) genera un patrón en la matriz de alimentación, que la segunda red neuronal artificial identifica como el objeto de interés, evitando el ruido y reduciendo la confusión con patrones de imágenes que no sean helipad, pero que se puedan llegar a parecer. Para esto se alimenta a la segunda red neuronal artificial con los datos obtenidos de la primera red neuronal artificial en intervalos de tiempo.

Al igual que en el programa anterior, primero se recibe una imagen en tiempo real o de una base de datos, como se muestra en la figura 90, se implementa la primera red neuronal artificial, a continuación, se procede a generar una segunda red neuronal artificial similar a la primera, usando nombres distintos para las variables y así evitar confusiones entre las distintas redes neuronales artificiales.

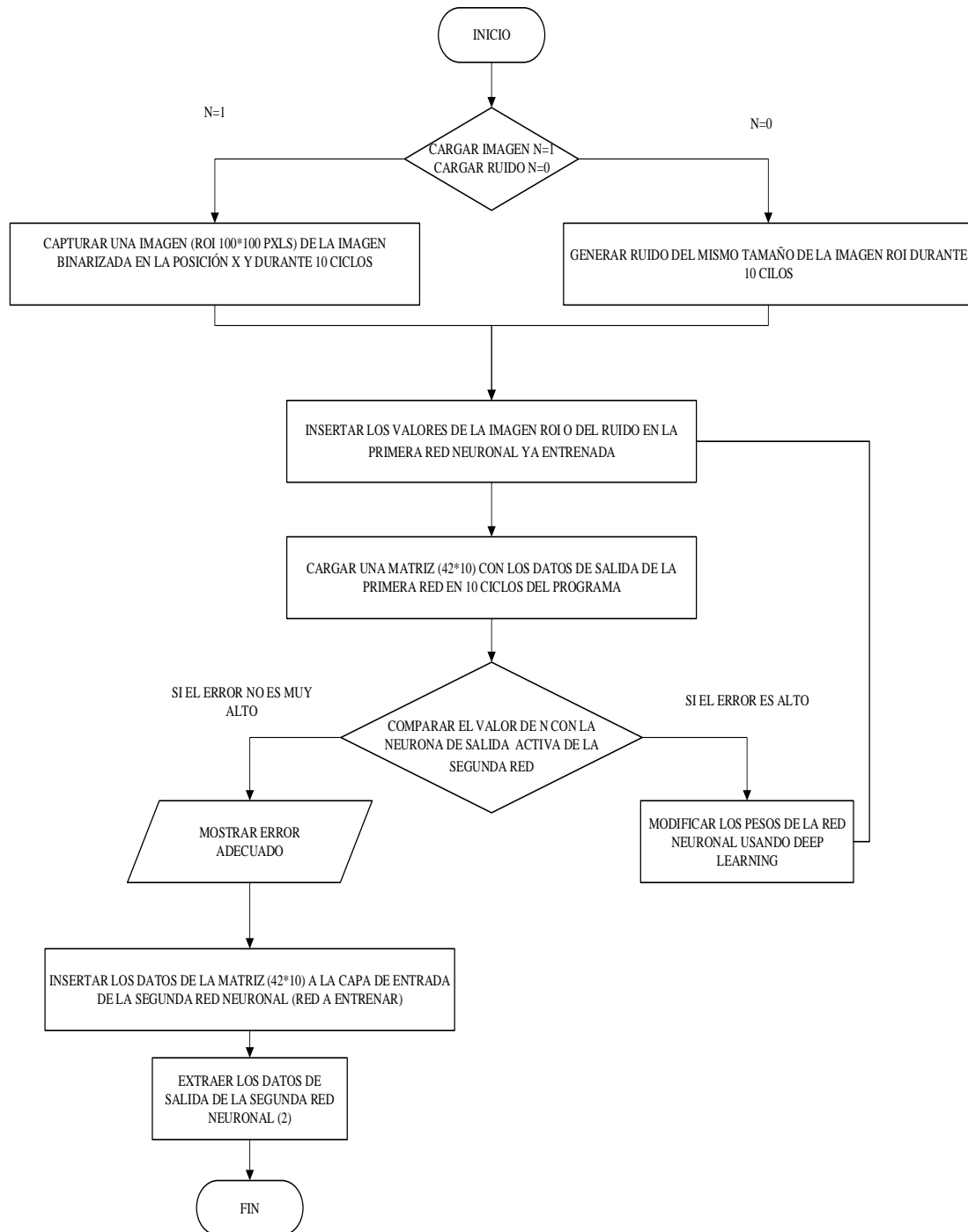


Figura 96. Diagrama de flujo de entrenamiento segunda red neuronal artificial.

En la segunda red neuronal artificial se usa 420 neuronas de entrada, 100 neuronas de capa escondida y 2 neuronas en la capa de salida, así también, se usa el método de entrenamiento de Backpropagation para alimentar la capa de entrada, en esta ocasión se usa la respuesta de la capa de salida de la primera red neuronal; generando así una matriz de espacio tiempo, que se va llenando fila a fila con los valores de la capa de salida de la primera red neuronal artificial y este dato a su vez pasa a alimentar cada una de las 420 neuronas de entrada de la segunda red neuronal artificial. Ver figura 97. Las barras de color rojo muestran la salida de la primera red neuronal artificial y la matriz de color blanco muestra los datos que se ingresa a la segunda red neuronal artificial.

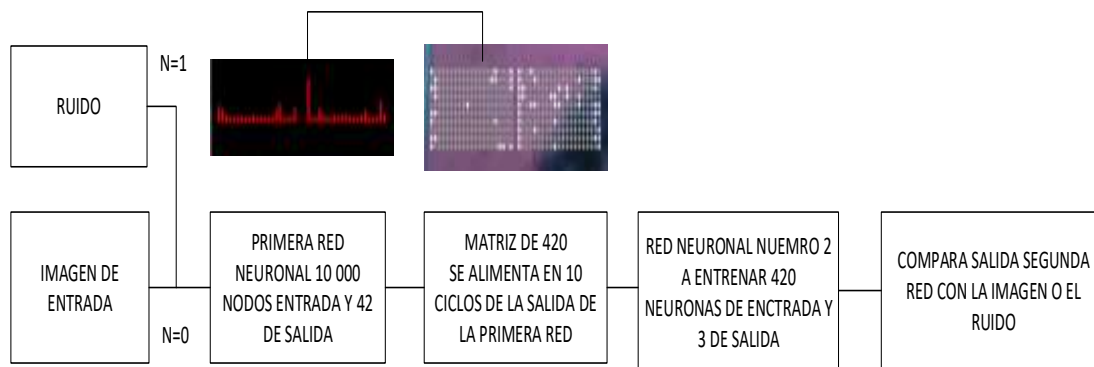


Figura 97. Proceso de alimentación segunda red neuronal artificial.

Luego de la realización de algunas pruebas (Ver tabla 43), el mejor número de neuronas de capa escondida sería 100; se decide trabajar con 100 neuronas, ya que, se obtienen resultados muy buenos y no se genera problemas de velocidad en el procesamiento de datos, así pues, al usar un número menor de neuronas en la capa oculta no se obtienen buenos resultados.

Tabla 43.

Relación entre las neuronas de salida y la posición del centro de la ROI.

Número de nodos de la segunda capa (entrenamiento de 20 minutos)	Inferencia (usando un ejemplo dibujado a mano y sin ruido)			Velocidad de procesamiento (prueba en modo debug medición por cronómetro)
	Aciertos	Pruebas 20 Correctos	Incorrectos	
200	95 %	19	1	+0.50 segundos
100	95 %	19	1	+0.25 segundos
45	90 %	18	2	base

Para la capa de salida se tienen dos neuronas, las mismas que, nos permiten identificar entre el ruido (imágenes falsas) y verdaderas (helipad), además para mejorar el entrenamiento se implementa un sistema de control manual, que permita realzar los bordes localizados en el algoritmo de CANNY (aumentarlos o disminuirlos según la cantidad de luz presente en el medio) y es necesario el uso de este filtro, puesto que, a diferencia de la primera red neuronal artificial, en el entrenamiento de la segunda red neuronal artificial, el efecto al entrenar es mejor, si se realiza usando el patrón de la salida de la primera red neuronal artificial de una imagen de un helipad real sin tanto ruido. Ver figura 98.

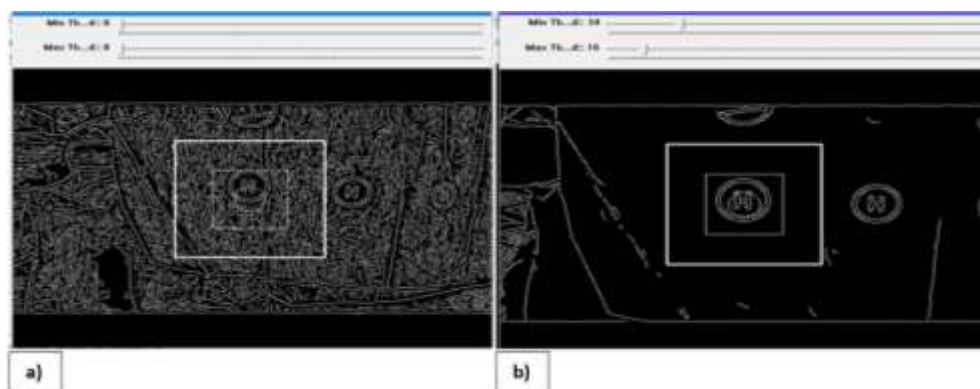


Figura 98. Control de CANNY y mejoramiento de detección de características, A) paso de todos los bordes y B) filtrado de los bordes menos contrastantes.

En la figura 99 se visualiza la imagen capturada desde la cámara junto a la salida de la primera RNA (42 listones rojos) la matriz de alimentación de la segunda red neuronal artificial en color blanco, así también, la ventana de CANNY se observa en la parte superior el control del threshold.



Figura 99. Entrador de la segunda red neuronal artificial.

3.3.3 Desarrollo del programa buscador.

En este programa se une todo el conocimiento adquirido en los programas entrenadores, lo cual permite que las dos redes neuronales artificiales encuentren la figura de interés, en un entorno real, evitando la mayor interferencia de ruido posible para con esto poder usar un método de seguimiento de trayectorias y permitir también guiar al dron a su lugar de aterrizaje. El programa se implementa usando openCV, para la adquisición de imágenes en tiempo real, una cámara, la misma que captura imágenes desde el dron para identificar y buscar el objetivo en la superficie. (Terreno, campo). Así también, se implementa la captura de un área de 100 x 100 píxeles (ROI), a la cual se le aplica el algoritmo de CANNY, que luego alimenta con estos 10 000 datos a la primera red neuronal artificial, que está conformada por 10 000 neuronas de entrada, 100 neuronas escondidas y 42 neuronas de salida, además de que contiene los valores (pesos) obtenidos del primer programa de entrenamiento; también se implementa un control manual de los parámetros (threshold) de el algoritmo de Canny para mejorar el reconocimiento. Luego se implementa una segunda red neuronal artificial con 420

neuronas de entrada, alimentadas de la misma manera que en el programa de entrenamiento de la segunda red neuronal artificial, conformada por 100 neuronas escondidas y 2 neuronas de salida, que se utiliza para identificar el objeto de interés (helipad) mientras se desplaza dentro de la ROI. Al ya tener implementado el algoritmo de CANNY y las dos redes neuronales artificiales en cascada, a continuación, se implementa un buscador que mueve la ROI (imagen de 100 x 100 píxeles) dentro de la imagen capturada desde la cámara, permitiendo analizar y buscar el objeto de interés (helipad) de forma autónoma, mientras la ROI se mueve, una de las dos neuronas de salida de la segunda red neuronal artificial informa que en la ROI existe un helipad. Para identificar el helipad durante la búsqueda, se usa el valor obtenido a la salida de la segunda red neuronal artificial, que identifica el helipad luego de tratar de centrarse sobre el mismo, actúa el algoritmo de Lucas Kanade, si es un helipad el tercer listón de color rojo se eleva Ver figura 100 A. y de no ser un helipad se mantiene abajo Ver figura 100 B. Si no identifica el patrón, sigue buscando en la imagen, en el caso de identificar el objeto de interés se activa el predictor Lucas Kanade, Ver figura 102 el cual analiza los contornos existentes en el área donde se encuentra la ROI y procede a tomar el control sobre el movimiento de la ROI, así como también permite saber la posición en la que se encuentra la ROI, tomando como referencia el centro de la imagen.

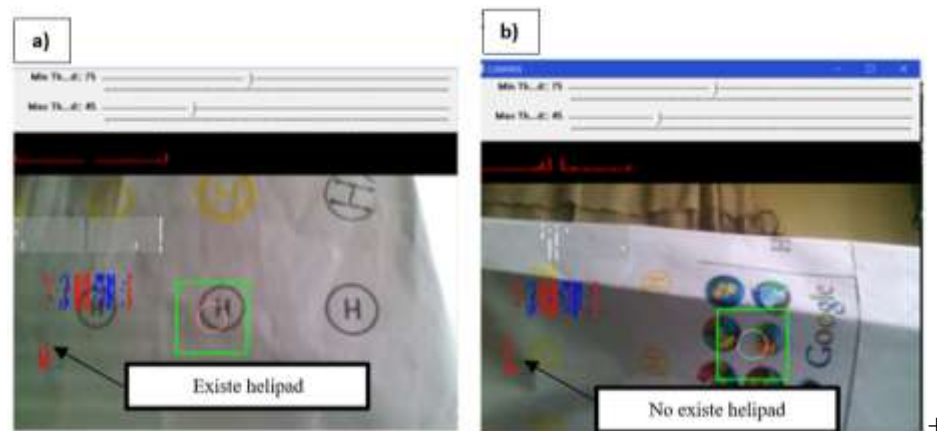


Figura 100. Detección entre helipad y ruido A existe helipad B no existe.

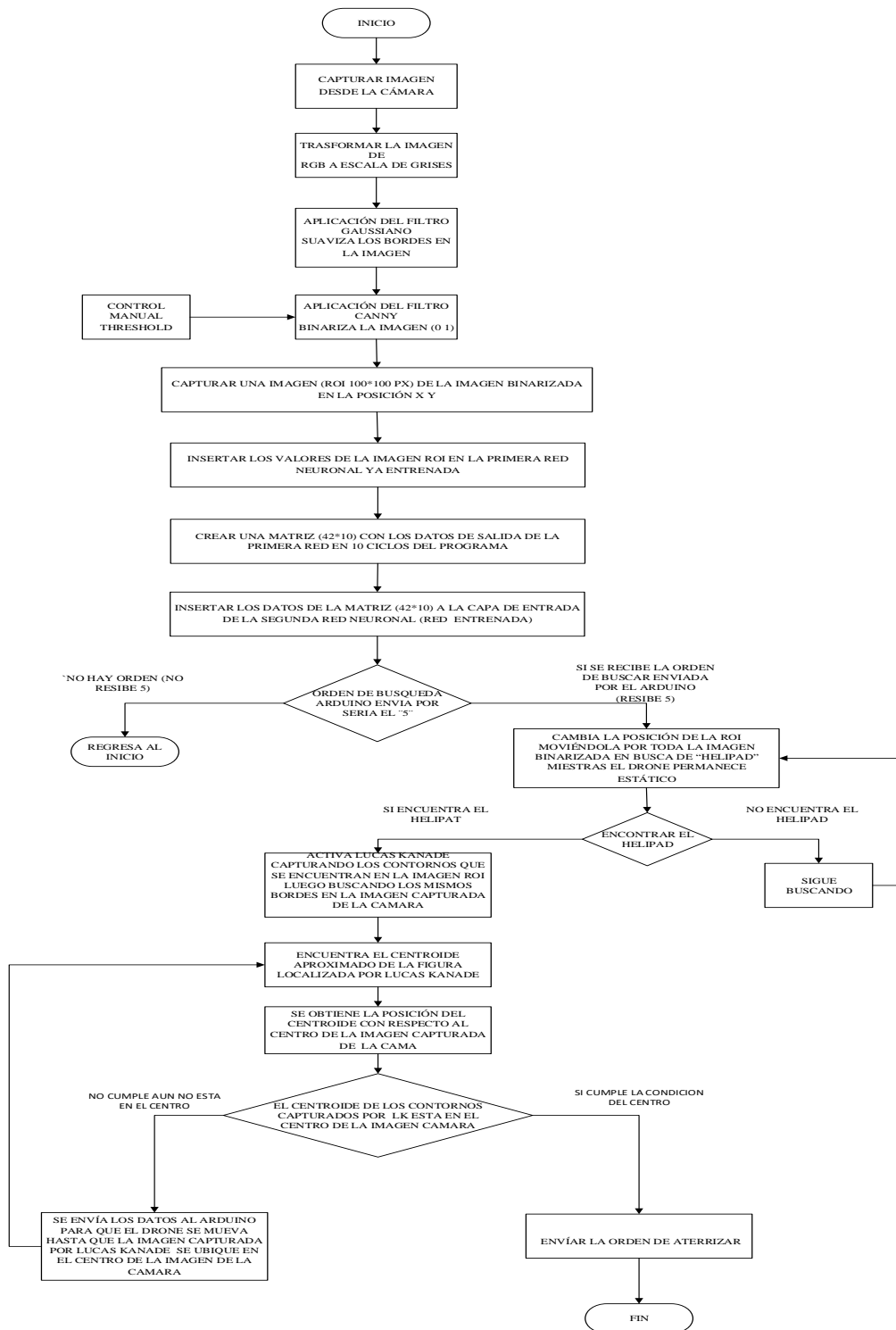


Figura 101. Flujo grama del programa buscador.

Con el dato de la posición y la posibilidad de seguir de forma constante el helipad, se mueve el dron desde la posición en la cual se encuentre, para lo cual se implementa librerías y sub programas que permitan enviar datos de forma serial, los datos a enviar corresponden a la posición del dron relativa al centro de la imagen, con el objetivo de posicionar al dron sobre el helipad para poder aterrizarlo, el programa queda de la siguiente forma figura 101.



Figura 102. Lucas Kanade.

CAPÍTULO IV

ANÁLISIS DE RESULTADOS

4.1 Introducción.

En el presente proyecto se ha desarrollado e implementado un método que permite a un cuadricóptero realizar un despegue, control de altura, ruta GPS, identificación, seguimiento de imágenes y aterrizaje autónomos.

4.2 Control ARDUINO

A continuación, se detallan las principales pruebas para el control mediante la tarjeta “Arduino MEGA 2560” de los procesos de generación de “SEÑALES PWM” encargadas de: “DESPEGUE”, “CONTROL ALTURA”, “RUTA”, “COMUNICACIÓN - PC” y “ATERRIZAJE”, que se han ejecutado mientras se desarrollaba el proyecto.

4.2.1 Señales PWM

Para lograr una exitosa emulación de las señales PWM, generadas por el aparato de recepción y transmisión “Turnigy 9X” para los canales 1,2,3 correspondiente a los canales “Alerón”, “Elevador y “Acelerador” respectivamente, se identificó los parámetros de estas señales mediante un osciloscopio que se muestra en el capítulo tres y se ha experimentado con algunas ideas como se muestra a continuación.

- a) Inicialmente se ha optado por la utilización de relés de bobina para el cambio entre el modo manual y el modo autónomo, de lo cual se determinó que al poseer una bobina cada relé genera una peligrosa interferencia magnética a la cual la tarjeta de control de vuelo “NAZA M Lite” no se encuentra acostumbrada y podría provocar un siniestro de la aeronave, también al poseer una velocidad de conmutación de 10 ms muy lento en comparación con los

relés de estado sólido cuyo tiempo de conmutación es de 20 μ S obtenidos de su datasheet respectivos y un número de accionamientos limitado, se ha decidido descartar esta posibilidad.

- b) Para realizar la emulación de la señal PWM de cada canal del receptor “Turnigy 9x”, en principio se utilizó el comando arduino “AnalogWrite (Pin, PWM)” donde se genera una señal PWM controlando un ciclo de trabajo entre 0 y 255, pero con una frecuencia y periodos constantes establecidos de fábrica por la tarjeta “Arduino MEGA 2560” como se muestran en la tabla 44, y debido a que la frecuencia de trabajo de las señales PWM a emular es de 51,28 Hz demostrado en el capítulo tres, se decidió la generación de cada señal mediante el comando arduino “DigitalWrite(PIN,VAL)” y “delayMicroseconds(VAL)” que permiten un control total sobre el ciclo de trabajo, el periodo y la frecuencia de las señales PWM.

Tabla 44.

Frecuencias establecidas de fábrica para la tarjeta arduino MEGA 2560.

PWM PIN Arduino	Frecuencia Hz
5,6	980 Hz
9	3906 Hz
3,9,10,11	31250 Hz
10	31 Hz
5,6,9,10	256 y 1024 Hz
3	1,8,32,64,128 Hz

Fuente: (Arduino, Arduino MEGA 2560, 2015)

- c) Cada vez que no se logró la correcta emulación de la señal PWM de cada canal la tarjeta NAZA entraba en modo emergencia como se muestran en la figura 103, es por ello la necesidad del uso de los parámetros anteriores.

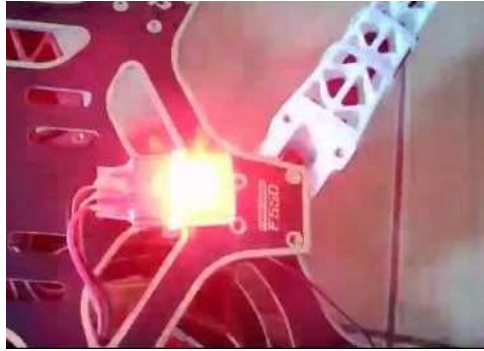


Figura 103. Modo Emergencia NAZA M Lite.

4.2.2 Despegue

Para conseguir un correcto despegue de la aeronave se necesitó del control del canal 3 o “Acelerador” y de la correcta medición de la altura a la cual se encontraba la aeronave mediante un sensor ultrasónico, para ello se ensayó algunas propuestas.

- a) Para comenzar se ubicó los pesos de cada uno de los componentes que se implementaron sobre el multirrotor, de modo que el centro de gravedad se encuentre en el centro de la aeronave como se muestra en la figura 104, para así lograr un despegue totalmente vertical sobre el eje Z.



Figura 104. Centro de gravedad del multirrotor.

- b) El multirrotor con los sistemas implantados adquirió un peso de 2.9 kg, para lo cual se realizó un análisis de peso vs capacidad de motores, cada motor puede cargar un peso de 820 g. en la configuración de hélices 10x4.5, utilizando una eficiencia de 6 g/w, los datos fueron obtenidos del datasheet respectivo, entonces se dispone una configuración de cuatro motores que brindan una carga de 3.28 kg, suficiente para conseguir elevarse sin inconvenientes.
- c) En comienzo se intentó utilizar el sensor barométrico de presión atmosférica “BMP180”, pero se determinó que sus características no se aplican para proyectos en donde la detección mínima del sensor debe ser 2 a 3 m y este cambio debe ser manifestado y registrado por la tarjeta arduino, entonces se optó por el sensor ultrasónico “HSR04” con una detección mínima de 5 mm.
- d) De igual manera se intentó varios despegues en diferentes terrenos como cemento (figura 105.a), asfalto (figura 105.b), arenilla (figura 105.c), tierra (figura 105.d) y césped (figura 105.e), de los cuales se ensayó la respuesta de las ondas ultrasónicas enviadas por el sensor se muestra en la tabla 46, tabla 47, tabla 48, tabla 49 respectivamente.

Tabla 45.

Pruebas de respuesta del sensor sobre una superficie de cemento.

Prueba #	Control de altura 2m	% error	Apagado de motores menos 30 mm	% error
1	2,2 m	10 %	22 mm	26,66%
2	2,6 m	30 %	25 mm	16,66 %
3	2,4 m	20 %	35 mm	16,66%

CONTINÚA 

4	2,3 m	15 %	32 mm	6,66%
5	2,2 m	10 %	36 mm	20 %
6	2,1 m	5 %	24 mm	20 %
7	2,3 m	15 %	29 mm	3,33 %
8	2,4 m	20 %	30 mm	0 %
Total	18,5 m	125 %	233 mm	110%
Promedio	2,31 m	15,6%	29,12 mm	13,75 %
Efectividad	84,375 %		86,25 %	

Tabla 46.

Pruebas de respuesta del sensor sobre una superficie de asfalto

Prueba #	Control de altura 2m	% error	Apagado de motores menos 30 mm	% Error
1	2 m	0 %	25 mm	16,66 %
2	1,9 m	5 %	29 mm	3,33 %
3	2,2 m	10 %	32 mm	6,66 %
4	2,5 m	25 %	36 mm	20 %
5	1,85 m	7,5 %	25 mm	16,66 %
6	2,3 m	15 %	30 mm	0
7	2,4 m	20 %	27 mm	10
8	2 m	0 %	29 mm	3,33 %
Total	15,15 m	82,5 %	233 mm	76,66 %
Promedio	1,89 m	10,3%	29,12 mm	9,58 %
Efectividad	89,68 %		90,41 %	

Tabla 47.

Pruebas de respuesta del sensor sobre una superficie de arenilla.

Prueba #	Control de altura 2m	% error	Apagado de motores menos 30 mm	% error
1	2,9 m	45 %	40 mm	33,33 %
2	3,2 m	60 %	35 mm	16,66 %
3	3,8 m	90 %	26 mm	13,33 %
4	3 m	50 %	46 mm	53,33 %
5	2,7 m	35 %	29 mm	3,33 %
6	3,4 m	70 %	39 mm	30 %
7	3,6 m	80 %	34 mm	13,33 %
8	2,9 m	45 %	38 mm	26,66 %
Total	25,5m	475 %	241 mm	190 %
Promedio	3,18 m	59,3 %	30,125 mm	23,75 %
Efectividad	40, 625 %		76,25 %	

Tabla 48.

Pruebas de respuesta del sensor sobre una superficie de tierra.

Prueba #	Control de altura 2m	% error	Apagado de motores menos 30 mm	% error
1	3,2 m	60 %	20 mm	33,33 %
2	4,1 m	105 %	36 mm	20 %
3	5 m	150 %	41 mm	36,66 %
4	3,9 m	95 %	56 mm	86, 66 %
5	3,2 m	60 %	43 mm	43,33 %

CONTINÚA 

6	4,1 m	105 %	59 mm	96,66 %
7	2,6 m	30 %	65 mm	116,66 %
8	2,9 m	45 %	63 mm	110 %
Total	29 m	650 %	383 mm	543,33 %
Promedio	3,625 m	81,2 %	47,87 mm	67,91 %
Efectividad	18,75 %		32,08 %	

Tabla 49.

Pruebas de respuesta del sensor sobre una superficie de césped.

Prueba #	Control de altura 2m	% error	Apagado de motores menos 30 mm	% error
1	4 m	100 %	40 mm	33,33 %
2	5,6 m	180 %	52 mm	73,33 %
3	4,1 m	105 %	63 mm	110 %
4	3,9 m	95 %	71 mm	136,66 %
5	2,7 m	35 %	26 mm	13,33 %
6	3,8 m	90 %	54 mm	80 %
7	4,2 m	110 %	46 mm	53,33 %
8	2,1 m	5 %	81 mm	170 %
TOTAL	17,15 m	720 %	433 mm	670 %
PROMEDIO	2,14 m	90 %	54,125 mm	83,75 %
EFFECTIVIDAD	10 %		16,25 %	

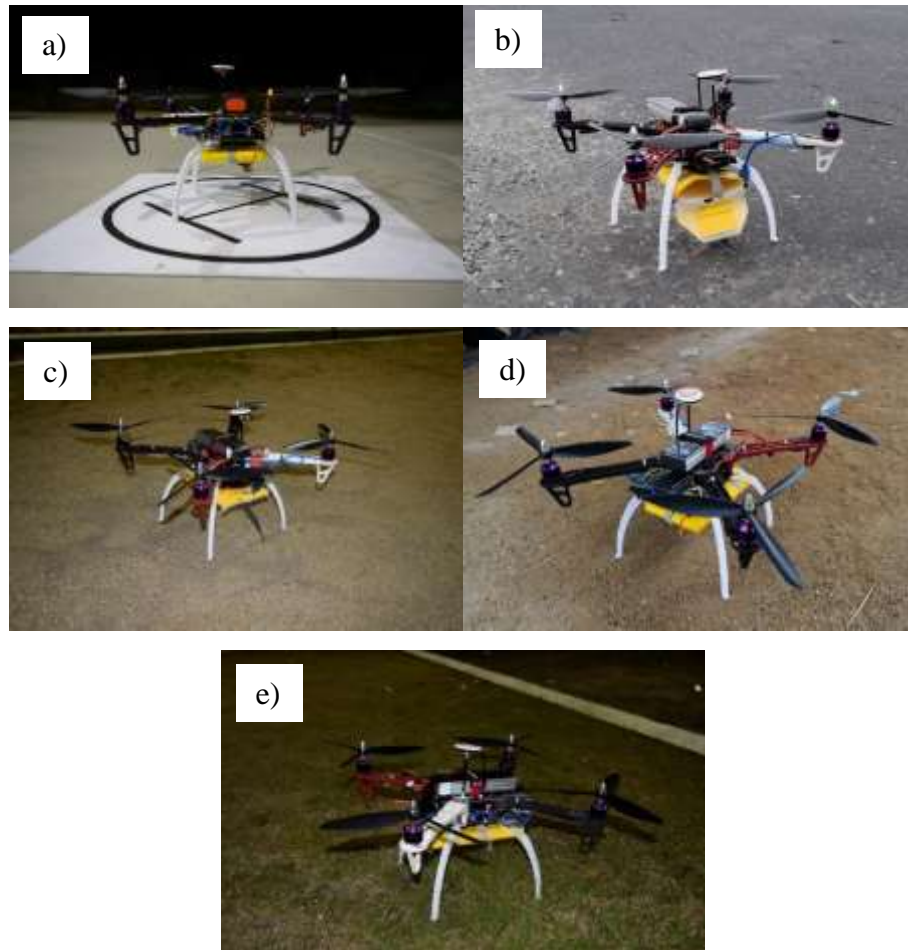


Figura 105. Superficie de pruebas.

Así, se determinó que el mejor terreno para un correcto despegue y control de altura es el asfalto con una efectividad de 89,68 % para despegue y 90,41 % en aterrizaje, seguido por el cemento con una efectividad de 84,37 % para despegue y 86,25 % en aterrizaje, ya que al poseer una superficie puramente solida se obtiene una mejor lectura de la distancia entre el multirrotor y el piso, así también la arenilla con una efectividad de 40,625 % para despegue y 76,25 % en aterrizaje, y la tierra con una efectividad de 18,75 % para despegue y 32,08% en aterrizaje, pero con el inconveniente de que el tren de aterrizaje se incrusta en las mismas provocando un despegue forzado

y una lectura con menor eficiencia de datos, ya que posee una superficie semisólida, y finalmente el césped con una efectividad de 10 % para despegue y 16,25 % en aterrizaje,, con una superficie nada sólida, como se muestra en la tabla 50.

Tabla 50.

Análisis de superficies de vuelo.

Superficie	% Efectividad	% Efectividad	Observaciones
	Despegue	Aterrizaje	
Asfalto	89,68 %	90,41 %	
Cemento	84,37 %	86,25 %	
Arenilla	40, 625 %	76,25 %	El tren de aterrizaje se incrustaba en la arena
Tierra	18,75 %	32,08%	El tren de aterrizaje se incrustaba en la tierra
Césped	10 %	16,25 %	La lectura del sensor era muy mala.

4.2.3 Control de Altura

Una vez superado el poseer un sensor que brinde una lectura confiable de la distancia entre el piso y el multirotor, se ensayó varios métodos para realizar el control de altura y ubicar el centro de gravedad en la posición más favorable, que permita realizar el correcto control de la altura a la cual se desarrollará la ruta hacia la zona del helipad y su posterior identificación mediante la RNA programada en el computador abordo, se experimentó algunas ideas.

- a) Inicialmente se efectuó un controlador ON/OFF para realizar la tarea del control de altura, pero al no poseer una rápida reacción no se logró un correcto control de la misma

- b) Luego se implementó un sistema de control mediante la librería PID propia de las tarjetas arduino, pero no se consiguió un control eficiente de la altura, debido a que los cambios en la lectura del sensor eran muy lentos y la reacción ante un descenso o ascenso de altura no era corregido con seguridad.
- c) Entonces, ya que la tarjeta controladora de vuelo posee un PID interno programado para mantener la altura si la señal GPS es buena, se determinó mantener la velocidad de aceleración del multirrotor cuando el sensor marque la altura deseada, logrando así el control de altura.

4.2.4 Ruta

Ya conseguido el correcto control de la altura de la aeronave, se ensayó algunas ideas de como el multirrotor podía dirigirse de manera autónoma hacia la zona helipad, donde se realizaría su posterior aterrizaje.

- a) Inicialmente se implementó una ruta mediante el sensor GPS “GY-GPS6MV2” y la programación descrita en el Anexo D, para la cual el multirrotor necesitaba de 2 minutos para lograr la posición establecida en la programación, ya que, debido al peso extra de los elementos necesarios para el proyecto, el tiempo de vuelo se redujo a 4 minutos, razón por la cual no se usó esta forma de posicionamiento hacia la zona de aterrizaje.
- b) Entonces, se programa una ruta más simple en la que el multirrotor se acerca a la zona donde se encuentra el helipad, para lo cual dispone de 15 segundos desde el despegue hasta su correcto posicionamiento en la referencia establecida.
- c) Hay que tomar en cuenta que el viento es un factor crítico y que afecta más en ciudades de la sierra como Latacunga y Ambato, las pruebas realizadas se muestran en la tabla 51.

Tabla 51.**Pruebas de ruta GPS realizadas en las ciudades de Ambato y Shell-Puyo.**

Prueba #	Ambato		Shell	
	Desviación ruta	% error	Desviación ruta	% error
	1 m		1 m	
1	2 m	100 %	1 m	0 %
2	3 m	200 %	2 m	100 %
3	2 m	100 %	1 m	0 %
4	1 m	0 %	3 m	10 %
5	0,9 m	10 %	2 m	30 %
6	1,9 m	90 %	1 m	0 %
7	1,7 m	70 %	4 m	10 %
8	1,6 m	60 %	2 m	40 %
Total	14,1 m	630 %	16 m	190 %
Promedio	1,76 m	78,75 %	2 m	23,75 %
Efectividad	21,25 %		76,25 %	

Donde se determinó que la ciudad de Shell-Puyo con el 76,25 % de efectividad en sus rutas es la más idónea para el proceso logrando que el multirotor alcance la zona

helipad como se muestra en la Figura 106 mientras que la ciudad de Ambato con tan solo un 21,25 % de efectividad en sus rutas no consiguió cumplir los requerimientos, debido al viento de la sierra, haciendo que la aeronave no logre llegar a la zona de aterrizaje como se muestra en la figura 107.



Figura 106. Ruta del multirrotor afectada por el factor del viento.

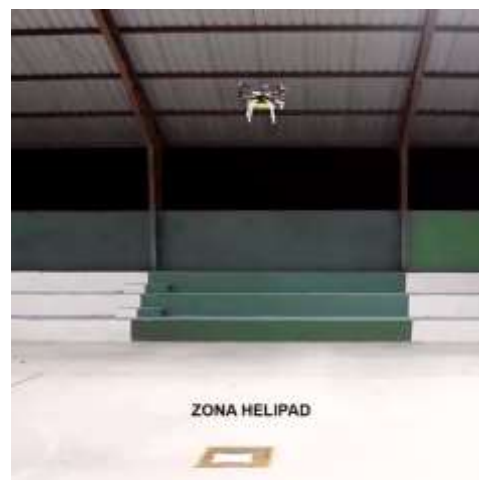


Figura 107. Ruta del multirrotor completada.

4.2.5 Comunicación - PC

Para implementar el direccionamiento controlado mediante la RNA programada en el computador, fue necesario la correcta generación y control de señales PWM de

cada canal y la necesidad de establecer códigos de posicionamiento específicos que se muestra en el capítulo 3, por lo cual se probó algunas propuestas.

- a) Normalmente la mayor velocidad de transmisión es la más efectiva al momento de trabajar con comunicación serial, pero debido a que el módulo GPS-GY6MV2 solo alcanza velocidades de hasta 9600 bps, por lo cual se limita la velocidad de comunicación del sistema a este valor.
- b) También, se implantó un proceso de discretización de datos mediante códigos ASCII, para lo cual fue necesario el desarrollo de un vector de caracteres que almacene toda la trama serial enviada desde la RNA y un contador que mueva la posición del vector y lea la misma para obtener el código de posicionamiento correcto.
- c) Mediante dichos códigos la tarjeta arduino puede controlar la posición del multirrotor para así ubicarlo sobre el helipad, gracias a la identificación de imágenes de la RNA.

4.3 Análisis del desarrollo de las redes neuronales artificiales.

Para mejorar el reconocimiento inteligente del helipad se decidió el uso de dos redes neuronales artificiales, conectadas en cascada, así como se describe a continuación.

4.3.1 Análisis del desarrollo de la primera red neuronal artificial.

Para esta red neuronal artificial se consideró una capa de entrada de diez mil neuronas las mismas que se alimentaron de la ROI, para la capa de neuronas escondidas se consideró cien neuronas lo que significa una compresión de la data lo suficientemente alta para aligerar los procesos de cálculo sin tener pérdidas significativas de los valores de entrada y permitiendo su tratamiento eficiente.

$$\frac{\text{número de neuronas escondidas}}{\text{número de neuronas de entrada}} \times 100\% = \text{primera compresión de la data}$$

$$\frac{100}{10000} \times 100\% = 1\%$$

Luego se consideró distintos números de neuronas de salida en la primera red neuronal artificial se probó con 22 neuronas de salida lo que permitía captar la posición del helipad con una variación pequeña de 11 pixeles por eje (11 en “x” y 11 en “y”) lo que posteriormente vuelve lento el proceso y reduce aún más la compresión de la data, por lo cual se decidió utilizar 42 neuronas de salida consiguiendo una compresión de datos entre la capa media y la de salida del 42% y una compresión total del 0.0042 datos además de reducir el desplazamiento al momento de mapear a la mitad.

$$\frac{\text{número de neuronas de salida}}{\text{número de neuronas escondidas}} \times 100\% = \text{segunda compresión de la data}$$

$$\frac{42}{100} \times 100\% = 42\%$$

compresión total de la data

$$0.01 * 0.42 = \text{compresión total de la data}$$

$$\text{compresión total de la data} = 0.0042$$

Esta red neuronal artificial fue entrenada usando el método de “Backpropagation”, en el proceso de entrenamiento fue notorio que se producían mínimos locales, al entrenar la red neuronal artificial con todos los parámetros que se deseaba enseñarle (posición respecto al centro de la ROI, ángulo, y tamaño de la imagen) desde el principio, por lo que se procedió a entrenar la red neuronal artificial aumentando de forma paulatina las variaciones de las imágenes.

Durante el entrenamiento de la primera red neuronal artificial se procedió a mostrar imágenes con variaciones, pero siempre limpias, ver figura 108 y se continuó el

entrenamiento hasta conseguir que el error entre la posición real y posición aparente que identifica la RNA sea del 97%.

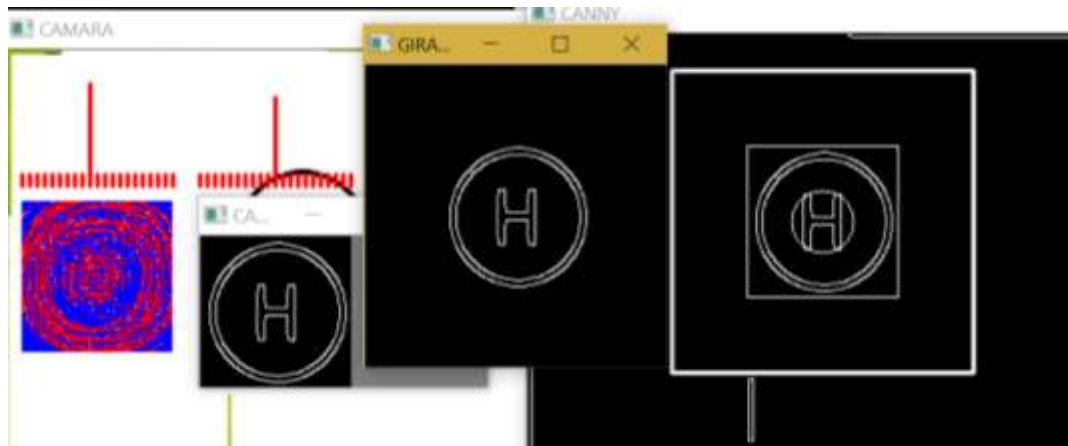


Figura 108. Resultados del entrenamiento de la primera red neuronal artificial.

En la figura 114 también se puede observar la respuesta de una de las neuronas escondidas en color azul y rojo, mientras que las barras de color rojo son las neuronas de salida, siendo la de mayor altura la posición correcta de la imagen tanto en “X” e “Y”.

Para el entrenamiento de la RNA se utilizó 15 ejemplos, cada ejemplo se rotó 36 veces una vez por cada 10 grados, y se lo trasladó 21 pixeles en “x” de uno en uno y 21 pixeles en “y” de uno en uno por lo tanto se obtienen 238140 ejemplos que alimentaron a la RNA.

total de ejemplos

= ejemplos reales * angulos rotados x desplazamientos en y
* desplazamientos en x

Total de ejemplos = $15 \times 36 \times 21 \times 21$

Total de ejemplos = 238 140

Se ha ensayado la RNA con 10 ejemplos, variando la posición del objetivo helipad a razón de 1 pixel en “x” e “y”, obteniendo así los resultados que se exhiben en la tabla

Tabla 52.

Posiciones del helipad al centro de la ROI vs posición detectada por la RNA.

Posición real en X	Posición real en Y	Posición identificada por la red en X	Posición identificada por la red en y	Error en X En porcentaje	Error en Y En porcentaje
0	0	0	0	0	0
5	5	5	5	0	0
10	10	10	10	0	0
15	15	13	16	15%	5%
20	20	18	17	10%	15%
-5	-5	-5	-5	0	0
-10	-10	-11	-10	5%	0
-15	-15	-15	-16	0	5%
-20	20	18	19	10%	5%
5	-5	5	-5	0	0
-5	5	-5	5	0	0
Promedio de errores				2.7%	2%

Promedio total de error 2.4%

Donde se concluyó que la RNA posee una efectividad de 97.6 % en la identificación de la posición del helipad, en X 97.3 % en la identificación de la posición del objetivo en Y 98% en la identificación del objetivo estos valores se consiguieron usando ejemplos que no contenían ruido y una mínima distorsión en su forma.

4.3.2 Análisis del desarrollo de la segunda red neuronal artificial.

La segunda RNA estaba orientada a analizar los patrones de salida de la primera red neuronal artificial, permitiendo de esta manera examinar si es o no es un helipad,

esto lo hace alimentándose de la salida de la primera RNA en diez ciclos, lo que quiere decir que contiene 420 entradas, las mismas que se van cargando continuamente, una capa interna de 100 neuronas para la compresión de la data y una salida de 2 neuronas que comprimen más aun la data y permite diferenciar el helipad, esta red neuronal artificial tiene un menor rango de compresión que la primera RNA.

$$\frac{\text{número de neuronas escondidas}}{\text{número de neuronas de entrada}} \times 100\% = \text{primera compresión de la data}$$

$$\frac{100}{420} \times 100\% = 23.8\%$$

$$\frac{\text{número de neuronas de salida}}{\text{número de neuronas escondidas}} \times 100\% = \text{segunda compresión de la data}$$

$$\frac{3}{100} \times 100\% = 3\%$$

$$0.03 * 0.238 = \text{compresión total de la data}$$

$$\text{compresión total de la data} = 0.0071$$

De la misma manera que en la red neuronal artificial número uno, en esta también se entrenó usando Backpropagation, pero para el caso se alimentó de la capa de salida de la primera RNA, la misma que se la entrenó usando las posiciones del helipad proporcionadas por la salida de la primera red neuronal artificial y con contra ejemplos en los que no se encontró las posiciones del helipad. Se la entrenó hasta que la neurona de salida número dos, que es la que indica si las posiciones ingresadas a la RNA son de un helipad, y tenga una efectividad del 95%.

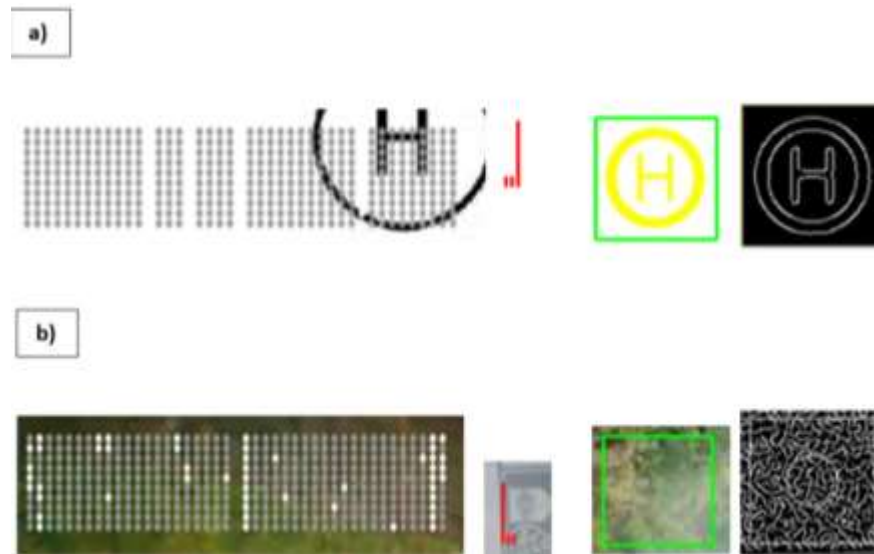


Figura 109. Respuesta de la segunda RNA A) cuando existe un helipad y B) cuando no existe.

En la figura 109 A En tonalidades de blanco se puede ver la matriz que alimenta a la segunda RNA, en listones de color rojo se visualiza cuando detecta ruido o detecta el helipad en el caso de la figura 109 A. se visualiza que si se ha detectado un helipad puesto que el tercer listón está elevando y en la figura 109 B cuando es ruido, se observa que la primera línea de color rojo se dispara.

La RNA puede y está entrenada para identificar helipuertos de diferentes tamaños, pero tiene una mayor precisión en el momento que las perturbaciones tiende a ser nulas dentro de la ROI y el helipad cubre entre un 70% y un 90% de la zona de interés como se observa en la figura 115 A

En la tabla 53 se observa la respuesta que tiene la salida de la segunda RNA de forma inmediata, al presentarle ejemplos sin perturbaciones. Este análisis permite saber el mejor tamaño del helipad a dibujarse para la búsqueda desde el aire, para lo cual se tomaron cinco muestras y se analizó el valor de excitación de la neurona que nos muestra la presencia del helipad.

Tabla 53.**Análisis de respuesta de la salida de la segunda RNA.**

Tamaño del helipad en comparación a la ROI	Respuesta de la segunda red neuronal					
	Número de pruebas 5					Porcentaje de eficiencia
	1	2	3	4	5	
Más del 90%	84.5	86.4	84.2	83.1	88.6	85%
Entre el 70 y 90 %	99.1	95.7	98.4	98.8	97.8	98%
Entre el 50 y 70%	75.2	76.6	81.4	83.4	80.9	79.5%
Entre el 30 y el 50 %	75.4	68.9	70.4	82.6	74.8	74%
Menor al 30%	49.9	37.8	54.1	39.7	45.5	45%

4.4 Análisis del desarrollo de los elementos de visión artificial.

En el desarrollo, el primer paso es la captura de una imagen en tiempo real la misma que se trabaja con un tamaño de 640 x 480 píxeles, esta imagen fue la que luego alimentó a la ROI, es de este tamaño para aumentar la velocidad de búsqueda del helipad y aún conservar una visión amplia de la zona.

Así también debido a la presencia del ruido y a pesar de que está implementado el filtro de Gauss para eliminar en lo posible las perturbaciones presentes en la imagen, también se optó por colocar un control de threshold ver tabla 54 que permite controlar de mejor manera el umbral de los bordes detectados por Canny, tanto en alto como en bajo, estos valores se los coloca según la cantidad de luz y brillo del ambiente de forma manual. Ver Figura 110.

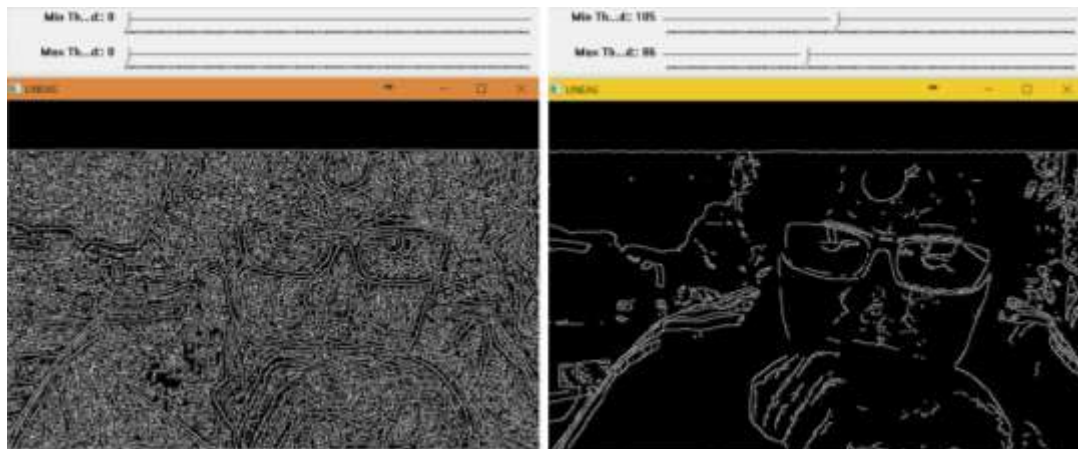


Figura 110. Variación del threshold.

Tabla 54.

Control del threshold.

Luminosidad con respecto a la luz del día	Threshold máximo	Threshold mínimo
Medio día	86	105
Mañana /tarde	45	75
Noche (iluminación artificial)	10	25

Para posicionar el cuadricóptero sobre el helipad se decidió usar el predictor Lucas Kanade debido a que responde más rápido y mejor al movimiento de la imagen que la red neuronal artificial cuando ya ha detectado bordes específicos, además permite tener una mayor velocidad de respuesta y menor confusión que al usarlas.

4.5 Análisis de la RNA y la cámara en el aire.

Al realizar las pruebas sobre una superficie de gravilla y tierra se pudo notar que, la interferencia presente en la imagen es muy alto, para lo cual se modificó manualmente el valor del threshold para conseguir una mejor resolución, así como se

puede observar en la figura 111 verificando de esta manera que el uso del corrector threshold consigue un mejoramiento en la captación de la imagen en las pruebas.

Considerando que la superficie gravilla y tierra es una de las más complicadas debido a la presencia de piedras las mismas que, generan una gran cantidad de bordes, interfiriendo con la detección del helipad y dificultando el encontrar un valor adecuado para el threshold.

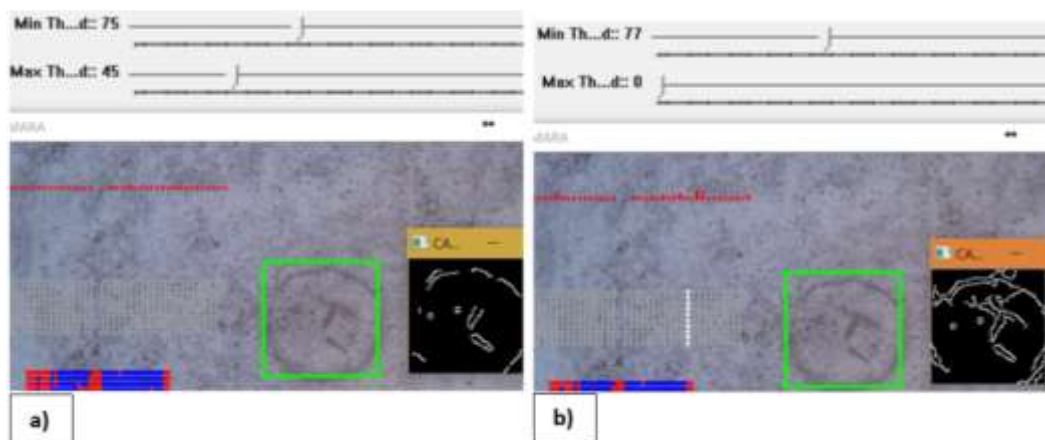


Figura 111. Variación del threshold desde el aire A) valor normalmente usado en las pruebas B) valor adecuado al terreno.

Después de realizar las pruebas en diferentes terrenos como se muestra en la figura 112 y en la tabla 55 se concluye que; los terrenos más difíciles para la detección del helipad son los que tiene superficies rugosas puesto que estas generan mayor cantidad de bordes que interfieren con la detección del helipad.

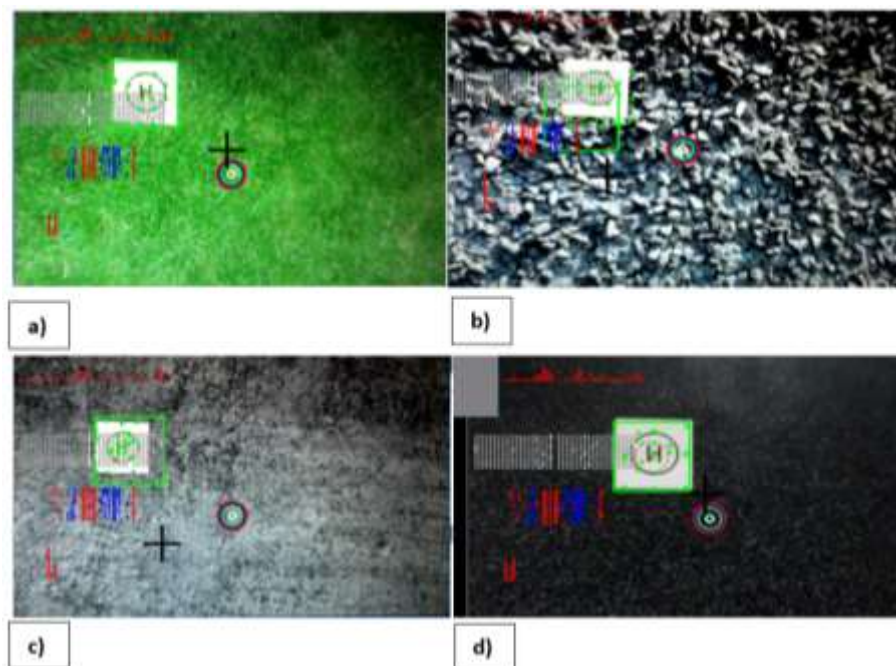


Figura 112. Tipos de terreno A) césped B) gravilla C) Tierra D) asfalto.

Tabla 55.

Pruebas de TRACKING en diversos terrenos.

Material	Pruebas realizadas	Pruebas efectuadas correctamente	Porcentaje de aciertos
Cemento	10	10	100%
Tierra	10	10	100%
Césped	10	10	100%
Asfalto	10	8	80%
Gravilla	10	7	70%

CONTINÚA 

4.6 Aterrizaje

Una vez conseguido el posicionamiento de la aeronave sobre el helipad, el control de descenso se realiza mediante la manipulación del acelerador de la aeronave a una velocidad constante establecida de 45% de aceleración como se explicó en el capítulo 3, hasta que el sensor ultrasónico marque una distancia menor a 30 mm desde el piso para realizar el posterior apagado de motores.

- a) De igual manera en las pruebas realizadas en la tabla 56, el aterrizaje autónomo se vio afectado por el factor del viento

Tabla 56.

Desviación de aterrizaje en pruebas de vuelo.

Prueba #	Desviación	% error	Desviación	% error
	Aterrizaje		Aterrizaje	
	Ambato		Shell	
	1 m		1m	
1	2 m	100 %	0,5 m	50 %
2	2,9 m	190 %	1 m	0 %
3	1,5 m	50 %	1,2 m	20 %
4	1,2 m	20 %	0,7 m	30 %
5	2 m	100 %	1,3 m	30 %
6	1 m	0 %	0,9 m	10 %
7	2,1 m	110 %	1,4 m	40 %

CONTINÚA 

8	0,7 m	30 %	1,1 m	10 %
Total	13,4 m	600 %	8 m	190 %
Promedio	1,67 m	75 %	1 m	23,75 %
Efectividad		25 %		76,25 %

Donde, se estableció un error permisible de 0 a 1m, considerado moderado para el sistema, así también se determinó que en la ciudad del Puyo – Shell con un 76,25 % de efectividad en las pruebas, se consigue un desarrollo más eficiente del aterrizaje, como se exhibe en la figura 114, mientras que en la ciudad de Ambato con un 25 % de eficiencia no se logró conseguir un mejor desempeño como se muestra en la Figura 113.



Figura 113. Aterrizaje afectado por el viento.



Figura 114. Aterrizaje cerca del objetivo helipad.

- b) Se ha cronometrado la autonomía de vuelo original del drone y es de 10 minutos, pero por el peso adicional de los sistemas, en la ciudad de Puyo se ha cronometrado una autonomía de 4 minutos, dando una eficiencia de vuelo del 40% y en la ciudad de Ambato unos 2 minutos, debido a las fuertes corrientes y la menor densidad del aire, dando como resultado una eficiencia de vuelo del 20%.
- c) De esta manera ya que el proyecto presenta un error promedio de 1 m representando un 76.25 % de efectividad en las pruebas de aterrizaje sobre el objetivo helipad, que comparado con un sistema de posicionamiento GPS como el “Arduipilot”, que dispone de un error de 2 metros (obtenidos del datasheet) de la posición de aterrizaje establecida, el proyecto presenta un 50% más de efectividad con respecto a lo presente el mercado, demostrando satisfactoriamente que el proyecto tiene validez.

CAPITULO V

CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones

Luego de haber realizados la investigación sobre el conjunto de técnicas en visión artificial y redes neuronales artificiales se puede afirmar que se logró usar de manera óptima y coordinada dos redes neuronales artificiales, así como también un sistema de binarización de imagen y un predictor que permiten identificar y posicionar al dron sobre el helipad para el aterrizaje autónomo.

Luego de la investigación se logró establecer las siguientes afirmaciones:

- a) Se realizó de forma correcta la transmisión de datos desde la imagen capturada por la cámara a las redes neuronales con una buena ejecución.
- b) Se determinó que el mejor terreno para un correcto despegue y control de altura es el asfalto con una efectividad de 89,68 % para despegue y 90,41% en aterrizaje.
- c) Se logró filtrar mucho las perturbaciones presentes en la imagen gracias a los algoritmos de CANNY.
- d) Se determinó que la ciudad de Shell-Puyo con el 76,25 % de efectividad en sus rutas es la más idónea para el proceso logrando que el multirrotor alcance la zona helipad.
- e) Se implementó de forma eficiente el predictor Lucas Kanade para el seguimiento de bordes del helipad identificado anteriormente por las redes neuronales artificiales.

- f)** Se logró crear una base de datos para entrenar las RNAs que posterior mente permitieron reconocer con éxito el punto de aterrizaje superando problemas de perturbaciones ambientales y variaciones en las muestras.
- g)** Se investigó acerca de los diferentes elementos que afectan un vuelo estable, así como los sistemas necesarios de comunicación y los diversos elementos que se pueden incorporar al cuadricóptero sin afectar enormemente su estabilidad y habilidad en el aire.
- h)** Se implementó elementos de bajo peso y alta calidad que permitan cumplir la misión con éxito.
- i)** Se determinó que en la ciudad del Puyo – Shell con un 76,25 % de efectividad en las pruebas, se consigue un desarrollo más eficiente del aterrizaje.
- j)** Se consiguió un acople exitoso entre el drone, el computador abordo, un arduino y un sistema de relés que además permite el control manual del UAV además de garantizar el control en caso de emergencias o fallos en el programa.
- k)** Se corrigió parámetros de iluminación que mejoran la detención del punto de aterrizaje además de parámetros de comunicación entre los distintos dispositivos y también el piloto.
- l)** Luego de realizar distintas pruebas tanto en tierra con ejemplos y simulaciones, como en el aire se consiguieron resultados favorables, aún a pesar de los distintos parámetros ambientales como la vibración la luminosidad el tiempo de vuelo entre otros.
- m)** Se implementó un sistema de alimentación independiente tanto para la parte de control a través de una batería de tres celdas y 2200 mAh y una fuente de

alimentación para la parte de potencia (motores, esc, tarjeta dji) usando una batería de tres celdas y 5200 mAh.

- n)** Se generó con éxito las señales PWM similares a las emitidas desde el receptor “Turnigy 9X”, que permitieron el control de los canales “Alerón, Elevador y Acelerador”, para los procesos de despegue, control de altura, ruta, posicionamiento neuronal y aterrizaje, evitando que la controladora de vuelo “NAZA M Lite” entre en modo de emergencia.
- o)** Se realizó el despegue autónomo de la aeronave con un 60 % de la aceleración del mismo, gracias al control de la tarjeta “NAZA M Lite”, que proporcionó la estabilidad y el control de altura para el desarrollo de los diferentes procesos en el aire.
- p)** Se implementó con éxito una ruta GPS, donde se le indican las coordenadas a las que debe ir el multirrotor para acercarse a la zona de aterrizaje donde se encuentra el objetivo de la identificación de imágenes de la red neuronal artificial.
- q)** Se estableció una adecuada comunicación serial entre la tarjeta “Arduino MEGA 2560” y la RNA programada en el computador “LIM8-RW10HP”, donde mediante códigos de posicionamiento se ubica el objetivo helipad en su campo de visión y envía los códigos hacia dónde debe ir el multirrotor para alcanzar su posición.
- r)** Se desarrolló un exitoso aterrizaje autónomo con un valor del 42 % de aceleración, cuando la red neuronal artificial ha identificado el objetivo helipad y posicionado la aeronave sobre una aproximación menor a 35 píxeles del mismo.

- s) Se instauró un sistema de emergencia, donde en caso de un fallo en algún proceso autónomo del multirrotor, se pueda tomar el control manual de la aeronave y evitar así un potencial siniestro.
- t) Se logró un correcto posicionamiento de la cámara para adquirir una completa visión de la zona de aterrizaje y una mayor facilidad en la identificación del objetivo helipad.
- u) Se implementó un procedimiento secuencial que evita que se activen fuera de orden los procesos de despegue, control de altura, ruta, comunicación y posicionamiento neuronal y aterrizaje, previniendo así una falla fatal.
- v) Se realizó un correcto procedimiento de cambio entre en modo manual y autónomo, gracias al uso de relés de estado sólido, evitando la activación del modo emergencia de la tarjeta “NAZA M Lite”.
- w) Se ejecutó un análisis de pesos para ubicar el centro de gravedad de manera que se logre un despegue controlado, donde el peso de la aeronave llegó a 2,9 kg, pero como el total de peso de carga de los cuatro motores es de 3.2 kg, la aeronave se elevó sin problemas.
- x) Se consiguió un posicionamiento adecuado del sensor ultrasónico que permita un correcto análisis de la señal, logrando así una fidelidad en la lectura de los datos de altura de la aeronave.
- y) Se implementó un sistema anti vibración para la cámara, evitando un 80% la vibración generada por el multirrotor y facilitando la tarea de identificación de imágenes.

- z) Se realizó la configuración necesaria para que el indicador led de la tarjeta “NAZA M Lite”, muestre al momento de que la batería principal se encuentre en niveles críticos.

5.2 Recomendaciones

Una vez terminadas las conclusiones se realiza las siguientes recomendaciones para un mejor funcionamiento del cuadricóptero y mejoras en futuras investigaciones dentro de este campo que podrían ser las siguientes.

- a) Se podría mejorar el sistema de filtración de contraste y luminosidad para una mejor identificación del helipad.
- b) Se debería evitar al momento de entrenar la RNA entrar en mínimos locales que no permitan la reducción del error para lo cual se debe avanzar incrementando la dificultad ligeramente.
- c) Se debería lograr un mejor equilibrio si entre la capacidad de la batería y el peso del multirotor se obtiene una sinergia.
- d) Sería bueno verificar el correcto funcionamiento y acople de todos los sistemas antes de realizar las pruebas o la ruta.
- e) Tratar de mantenerse atento de las cargas en las baterías para evitar el daño de las mismas.
- f) Debería asegurarse que la imagen no presente exceso de perturbaciones al momento de realizar el escaneo de la zona.
- g) Se debería optimizar el peso de los elementos para aumentar el tiempo de vuelo o a su vez implementar en un dron de mayor tamaño (hexacópteros, octacópteros, etc.).

- h)** Verificar las polaridades antes de realizar la conexión de los distintos dispositivos.
- i)** Intentar realizar las pruebas lejos de zonas pobladas, personas o niños.
- j)** Tratar de utilizar relés de rápido accionamiento y que no generen campos magnéticos que puedan perturbar el desempeño de la tarjeta controladora de vuelo y sus señales GPS.
- k)** Tratar de ubicar la tarjeta controladora de vuelo “NAZA M Lite”, justo en el centro de la aeronave ya que dispone de un giroscopio para el direccionamiento del multirrotor.
- l)** Manejar una batería con un “C” de descarga mínimo de 25C o mayor ya que debido al peso adicional implementado, los motores requieren de mayor amperaje para alcanzar las alturas deseadas.
- m)** Implementar un método anti vibración para la cámara ya que con el girar de los motores se produce una gran cantidad de vibraciones que dificultan la identificación de imágenes desde el aire.
- n)** Hacer lo posible para que la tarjeta controladora de vuelo “NAZA M Lite” no entre en modo emergencia, ya que al hacerlo toma el control de los sistemas y realiza un aterrizaje forzado que la mayoría de veces termina en un siniestro de la aeronave.
- o)** Siempre que se pueda, verificar que las hélices se encuentren en buen estado ya que si hubo antes algún accidente pueden encontrarse deterioradas y provocar futuras colisiones del multirrotor.

- p)** Disponer de un buzzer o indicador led que permita apreciar el nivel de voltaje de la batería, ya que. si se reduce demasiado puede terminar en un desastre de la aeronave.
- q)** Tratar de ubicar los sensores GPS en una posición elevada por encima de 10 cm de los sistemas electrónicos, porque, pueden perder la señal satelital y generar errores.
- r)** Tratar de disponer de un tren de aterrizaje robusto para que las pruebas de aterrizaje con el peso adicional de los sistemas que son algo bruscos no puedan romperlo.
- s)** Implementar un sistema de corrección de errores en la programación para evitar accidentes no deseados.
- t)** Intentar colocar el receptor de señales PWM lejos de los sistemas electrónicos ya que podría generar interferencias no deseadas.
- u)** Establecer una velocidad adecuada para el direccionamiento de la aeronave para que no pueda girar en vuelo ocasionando un inevitable siniestro.
- v)** Utilizar un piso que sea adecuado porque así los sensores ultrasónicos realizan mejor las mediciones cuando las superficies son más sólidas.

Bibliografía

- ADH-Tech. (02 de Agosto de 2014). *66-Channel GPS Module*. Recuperado el 22 de Octubre de 2016, de https://cdn.sparkfun.com/assets/learn_tutorials/4/6/8/GP3906-TLP_DataSheet_Rev_A01.pdf
- AliExpress. (14 de enero de 2013). *TCB 11.1 v 5000 mah 25c 3 s RC lipo*. Recuperado el 29 de abril de 2016, de <https://es.aliexpress.com/item/MOS-11-1v-5000mah-25c-lipo-battery-for-rc-airplane-free-shipping/32736498853.html>
- Angulo Valencia, L. V. (2014). *Generación de Cartografía Básica al Detalle Mediante una Metodología de UAVs*. Latacunga: Universidad de las Fuerzas Armadas - ESPE.
- Arduino. (20 de diciembre de 2013). *Arduino NANO*. Recuperado el 19 de octubre de 2016, de <https://www.arduino.cc/en/Main/ArduinoBoardNano>
- Arduino. (07 de febrero de 2015). *Arduino MEGA 2560*. Recuperado el 19 de Octubre de 2016, de <https://www.arduino.cc/en/Main/arduinoBoardMega>
- Arduino. (05 de Julio de 2015). *Arduino UNO*. Recuperado el 19 de Octubre de 2016, de <http://www.arduino.org/products/boards/arduino-uno>
- Banggood. (02 de diciembre de 2015). *Poder de zop 11.1v 5000mah 3s1p 30c lipo*. Recuperado el 29 de abril de 2016, de http://www.banggood.com/es/Wholesale-ZOP-Power-11_1V-5000mAh-3S1P-30C-Lipo-Battery-For-RC-Model-p-57136.html
- BOSCH. (12 de marzo de 2013). *BMP180 Digital pressure sensor*. Recuperado el 26 de octubre de 2016, de <https://cdn-shop.adafruit.com/datasheets/BST-BMP180-DS000-09.pdf>

- Calderón Francisco, U. G. (2008). *“Conteo Automático de Vehículos”*. Bogotá: Pontificia Universidad Javeriana.
- Casanova, A. (15 de febrero de 2012). *www.ub.edu/*. Recuperado el 15 de agosto de 2016, de ub: <http://www.ub.edu/pa1/node/101>
- DeMello, D. (28 de Abril de 2003). *library cornell*. Recuperado el 06 de octubre de 2016, de www.library.cornell.edu: https://www.library.cornell.edu/preservation/tutorial-spanish/tutorial_Spanish.pdf
- Dji. (13 de noviembre de 2014). *Dji Inspire 1*. Recuperado el 15 de octubre de 2016, de <https://www.dji.com/es/inspire-1/info>
- Dji. (19 de junio de 2014). *Dji Phantom 3*. Recuperado el 15 de octubre de 2016, de <https://www.dji.com/es/phantom-3-pro/info>
- Dji. (30 de mayo de 2014). *NAZA-M LITE*. Recuperado el 14 de abril de 2016, de <http://www.dji.com/es/naza-m-lite>
- Dji. (12 de octubre de 2014). *NAZA-M LITE User Manual*. Recuperado el 24 de abril de 2016, de http://dl.djicdn.com/downloads/naza-m%20lite/en/NAZA-M%20LITE_User_Manual_v2.00_en.pdf
- educacion, G. d. (5 de Febrero de 2012). *visionartificial.fpcat.cat*. Obtenido de [visionartificial.fpcat: http://visionartificial.fpcat.cat/wp-content/uploads/UD_1_didac_Conceptos_previos.pdf](http://visionartificial.fpcat.cat/wp-content/uploads/UD_1_didac_Conceptos_previos.pdf)
- Elecfreaks. (05 de marzo de 2015). *Ultrasonic Ranging Module HC - SR04*. Recuperado el 26 de octubre de 2016, de <http://www.micropik.com/PDF/HCSR04.pdf>

España, M. d. (04 de Julio de 2008). *ite.educacion.es*. Recuperado el 12 de julio de 2016, de [ite.educacion.es: http://www.ite.educacion.es/formacion/materiales/107/cd/imagen/imagen0105.html](http://www.ite.educacion.es/formacion/materiales/107/cd/imagen/imagen0105.html)

ETI, C. I. (agosto de 12 de 2013). *www.etitudela.com*. Recuperado el 19 de octubre de 2016, de [etitudela: http://www.etitudela.com/celula/downloads/visionartificial.pdf](http://www.etitudela.com/celula/downloads/visionartificial.pdf)

Flórez, R., & Fernández, M. (2008). *Las Redes Neuronales Artificiales*. España: NetBiblio.

Gómez, J., Fernández, M., López, T., & Díaz, A. (2009). *Aprendizaje con Redes Neuronales Artificiales*.

Gonzalo Pajares, J. M. (2001). *Visión por Computador. Imágenes digitales y aplicaciones. Ra-MA*.

GoPro. (03 de febrero de 2014). *HERO 3*. Recuperado el 20 de junio de 2016, de <https://es.gopro.com/update/hero3>

Guilcamaigua Maisincho, J., & Vaca Navas, W. (2013). *Diseño e implementación de una máquina semiautomática de termoformado con panel operador para la elaboración de empaques tipo blíster de un producto de la empresa Metal Arte Guilcamaigua*. Latacunga: Tesis de grado.

HobbyKing. (08 de septiembre de 2014). *Afro ESC 20A Motor Speed Controller*. Recuperado el 12 de abril de 2016, de https://hobbyking.com/en_us/afro-esc-20amp-multi-rotor-motor-speed-controller-simonk-firmware.html

HobbyKing. (15 de junio de 2014). *Hobby King 20A ESC 3A UBEC*. Recuperado el 10 de abril de 2016, de https://hobbyking.com/en_us/hobby-king-20a-esc-3a-ubec.html

- HobbyKing. (11 de julio de 2014). *MT2213-935KV MultiStar Motor*. Recuperado el 25 de abril de 2016, de https://hobbyking.com/en_us/mt2213-935kv-multistar-motor-and-propeller-combo-10x4-5-cw-ccw.html
- HobbyKing. (09 de mayo de 2014). *Multistar High Capacity 3S 5200mAh Multi-Rotor Lipo Pack*. Recuperado el 30 de abril de 2016, de https://hobbyking.com/en_us/multistar-high-capacity-3s-5200mah-multi-rotor-lipo-pack.html
- HobbyKing. (14 de junio de 2014). *Propeller 10x4.5 CW/CCW*. Recuperado el 21 de abril de 2016, de https://hobbyking.com/en_us/mt2213-935kv-multistar-motor-and-propeller-combo-10x4-5-cw-ccw.html
- HobbyKing. (06 de agosto de 2014). *Turnigy 9X*. Recuperado el 17 de octubre de 2016, de https://hobbyking.com/en_us/turnigy-9x-9ch-transmitter-w-module-8ch-receiver-mode-2-v2-firmware.html
- HobbyKing. (14 de octubre de 2014). *Turnigy D2836/9 950KV Brushless Outrunner Motor*. Recuperado el 22 de abril de 2016, de https://hobbyking.com/en_us/turnigy-d2836-9-950kv-brushless-outrunner-motor.html
- HobbyKing. (10 de noviembre de 2015). *Quantum Venture Drone*. Recuperado el 15 de octubre de 2016, de https://hobbyking.com/en_us/quantum-venture-fpv-deluxe-quad-copter-set-w-dji-fatshark-afro-parts-pnf.html
- HobbyPro. (17 de septiembre de 2014). *Flight Control APM 2.8*. Recuperado el 21 de abril de 2016, de <http://www.hobbypro-rc.com/multi-rotors-parts/flight-control-apm-2-8-for-multicopter-ardupilot-mega.html>

- Intel. (22 de octubre de 2015). *Intel® Compute Stick STCK1A32WFC*. Recuperado el 10 de julio de 2016, de <https://ark.intel.com/es/products/86612/Intel-Compute-Stick-STCK1A32WFC>
- J. G. (04 de Octubre de 2016). *www.consein.com*. Recuperado el 02 de septiembre de 2016, de *consein*: <http://www.consein.com/2016/10/04/el-mundo-de-los-drones-i-parte/>
- Kim-Sung Jie, M. L. (2005). “Computer Vision Based Real-time Information acquisition for. *IEEE*, 6.
- Lechner, N. (2005). *www.tectonica.es*. Obtenido de *tectonica*: http://www.tectonica.es/arquitectura/iluminacion/natural/iluminacion_natural.html
- López, A. (2012). *Diseño, construcción y control de un hexacóptero en plataforma de prueba*. Cartagena.
- Luna, V. M. (2012). Introducción al procesamiento digital de imágenes. *Universidad tecnológica de Mixteca*, 5-6.
- M. C. (12 de Septiembre de 2011). *etitudela.com*. Recuperado el 10 de junio de 2016, de *etitudela.com*: <http://www.etitudela.com/celula/downloads/visionartificial.pdf>
- Manualslib. (15 de enero de 2013). *Fly Sky FS-TH9X Instruction Manual*. Recuperado el 17 de octubre de 2016, de <https://www.manualslib.com/manual/908657/Fly-Sky-Fs-Th9x.html>
- Matich, J. (2001). *Redes Neuronales: Conceptos Básicos y Aplicaciones*. Rosario: Universidad Tecnológica Nacional.

- MaxBotix. (29 de septiembre de 2015). *HRLV-MaxSonar MB1003*. Recuperado el 25 de octubre de 2016, de http://www.maxbotix.com/documents/HRLV-MaxSonar-EZ_Datasheet.pdf
- Melamed, M. (02 de Noviembre de 2011). *paseoclick.wordpress.com*. Recuperado el 05 de junio de 2016, de [paseoclick.wordpress: https://paseoclick.wordpress.com/2011/11/08/tipos-de-camaras-digitales/](https://paseoclick.wordpress.com/2011/11/08/tipos-de-camaras-digitales/)
- Mora, D., & A. P. (Julio de 2010). *Detección de objetos en una escena utilizando flujo optico*. Bogota, Colombia: Universidad Javeriana.
- Nonami, K., Kendoul, F., Suzuki, S., Wang, W., & Nakazawa, D. (2010). *Autonomous Flying Robots*. Tokyo: Spinger.
- NoRunaway. (18 de febrero de 2014). *A New Flight Related Terms*. Recuperado el 16 de octubre de 2016, de <http://norunway.com/wp/flight-related-terms/>
- OMRON. (16 de octubre de 2013). *Solid State Relay G3MB*. Recuperado el 20 de octubre de 2016, de <https://www.openhacks.com/uploadsproductos/g3mb-ssr-datasheet.pdf>
- Pacini, P. (2009). *Manual de Instrucciones Turnigy 9X 2.4 GHz*. España: Surhobbies.
- Pacini, P. (2009). *Turnigy 9X Instruction Manual*. SURHOBBIES.
- Pajares Martin-Sanz, G. (21 de Diciembre de 2010). *sabia.tic.udc.es*. Recuperado el 22 de julio de 2016, de [sabia.tic.udc.es: http://sabia.tic.udc.es/gc/Contenidos%20adicionales/trabajos/3D/VisionArtificial/index.html#def](http://sabia.tic.udc.es/gc/Contenidos%20adicionales/trabajos/3D/VisionArtificial/index.html#def)
- Pino, R., Gómez, A., & de Abajo, N. (2001). *Introducción a la inteligencia artificial*. Oviedo: Universidad de Oviedo.

- Raspberry. (05 de septiembre de 2015). *RASPBERRY PI 3 MODEL B*. Recuperado el 16 de julio de 2016, de <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>
- Rebaza, J. V. (2010). Detección de bordes mediante el algoritmo de Canny . *Escuela Académico Profesional de Informática Universidad Nacional de Trujillo* , 2-4.
- Rebaza, J. V. (2011). Detección de bordes mediante el algoritmo de Canny . *Universidad Nacional de Trujillo* .
- Riba, C. (2004). *Diseño concurrente*. Quito.
- Romero Paredes, L. E. (2014). *Diseño y construcción de un módulo de control PID para la estabilización de un cuadricoptero*. Quito: Escuela Politécnica Nacional.
- SJCAM. (24 de marzo de 2015). *SJ4000*. Recuperado el 20 de junio de 2016, de <http://sjcamhd.com/shop-sj4000-series/>
- Songle. (11 de noviembre de 2012). *Songle Relay*. Recuperado el 20 de octubre de 2016, de [http://www.mycomkits.com/reference/Songle_SRD\(T73\)_Relay.pdf](http://www.mycomkits.com/reference/Songle_SRD(T73)_Relay.pdf)
- U-Blox. (14 de marzo de 2014). *NEO-6*. Recuperado el 21 de octubre de 2016, de https://www.u-blox.com/sites/default/files/products/documents/NEO-6_DataSheet_%28GPS.G6-HW-09005%29.pdf?utm_source=en%2Fimages%2Fdownloads%2FProduct_Docs%2FNEO-6_DataSheet_%28GPS.G6-HW-09005%29.pdf
- Urzi, B. A. (2007). *EL CÁLCULO DE LA*. Murcia: Universidad de Murcia.
- Wang, F. (2014). *Manual Naza M Lite*. Beijing: Dji.
- Wright, T. (2001). *Manual de fotografía*. Madrid: MaterPrint S.L.

Zeidan, F. (1990). *Aereodinámica Práctica Avanzada*. Bogota: Universidad de Los Andes.

ANEXOS



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

**DEPARTAMENTO DE CIENCIAS DE LA ENERGÍA Y
MECÁNICA
CARRERA DE INGENIERÍA MECATRÓNICA**

CERTIFICACIÓN

Se certifica que el presente trabajo fue desarrollado por los señores **LUIS MARCELO ROBALINO IBARRA** y **ALEX FABRICIO SEGOVIA FREIRE**.

En la ciudad de Latacunga a los 24 días del mes de febrero de 2017.

Ing. Dario Mendoza

DIRECTOR DEL PROYECTO

Aprobado por:

Ing. Vicente Itallo

DIRECTOR DE CARRERA

Dr. Rodrigo Vaca

SECRETARIO ACADÉMICO