

TABLA DE CONTENIDOS

	página
Dedicatoria	I
Agradecimientos	II
Tabla de Contenidos	III
Índice de Figuras	VI
Índice de Tablas	VIII
Resumen	IX
Abstract	x
1. Introducción	1
1.1. Descripción del problema	2
1.1.1. Análisis del problema	3
1.2. Objetivo	4
1.3. Alcances	5
1.4. Resumen del capítulo	5
2. Contexto del Proyecto	6
2.1. Vehículo Aéreo no tripulado	6
2.1.1. Origen de los UAV	6
2.2. Arquitectura de un UAV	7
2.3. Componentes del UAV	11
2.3.1. BeagleBone Black rev B	11
2.3.2. PixHawk	13
2.3.3. Batería	15
2.3.4. Cámara web	16
2.3.5. GPS	17
2.3.6. Control RC	18
2.4. Software	19

2.4.1. Debian OS	19
2.4.2. C++	19
2.4.3. Python	19
2.4.4. ROS	19
2.4.5. MAVlink	24
2.4.6. MAVROS	25
2.4.7. MAVProxy	25
2.4.8. Reconocimiento de puntos de interés	25
2.5. Resumen del capítulo	39
3. Implementación	40
3.1. Diseño arquitectónico del UAV	40
3.2. Comunicación de Pixhawk con Beaglebone	41
3.2.1. Comunicación	41
3.2.2. Pruebas fallidas	50
3.3. Diseño experimental	50
3.3.1. Cálculo de puntos	50
3.3.2. Metodología de la prueba	54
4. Resultados	58
4.1. Resultados esperados	65
4.2. Cuantificación del error	65
5. Conclusión	66
5.1. Visión General del trabajo realizado	66
5.2. Cumplimiento de los objetivos	67
5.3. Trabajo futuro	68
Glosario	69
Bibliografía	70
Anexos	
A: Formato archivo del PixHawk	74

ÍNDICE DE FIGURAS

	página
1.1. Logo de la competencia del año 2009.	2
1.2. Outback Joe es el nombre del maniquí que se utiliza en la competencia de búsqueda y rescate.	3
2.1. Descomposición en niveles de competencia de <i>Subsumtion Architecture</i> . Los niveles aumentan de abajo hacia arriba, comenzando en cero.	10
2.2. <i>Subsumtion Architecture</i> permite agregar nuevos comportamientos o capas de forma sencilla. La inclusión de sensores y actuadores es opcional [16]	11
2.3. BeagleBone Black rev b	12
2.4. PixHawk	14
2.5. Batería LiPo 5200 mAh 4 celdas.	16
2.6. Playstation Eye.	17
2.7. GPS Ublox Neo 7.	18
2.8. Control RC de 9 canales.	18
2.9. Típica configuración de red en ROS	20
2.10. Estructura de mensajes en ROS	22
2.11. Controlador de vuelo MultiWii.	26
2.12. Flujo Óptico.	27
2.13. Imagen original de un área determinada	28
2.14. Imagen procesada con los objetos detectados.	29
2.15. Imagen que muestra los contornos detectados.	30
2.16. Puntos que forman el rectángulo de un contorno.	31
2.17. imagen obtenida luego de filtrar por área.	32
2.18. Rectángulo contenido en otro.	33
2.19. Rectángulos sin tener otros contenidos dentro.	34
2.20. Rectángulos con el centro marcado.	35
2.21. Puntos a visitar.	36
2.22. Cuadrantes del plano cartesiano.	37
2.23. Ángulo del primer punto.	38

3.1.	Diagrama de la comunicación entre los sensores y los niveles de competencia.	41
3.2.	Diagrama de la comunicación estándar de <i>MAVProxy</i> con un <i>QuadCopter</i>	42
3.3.	Diagrama de la comunicación modificada del <i>MAVProxy</i> abordando un <i>QuadCopter</i>	43
3.4.	Mensaje del servidor dentro de MAVProxy.	45
3.5.	Mensaje del servidor luego de recibir y procesar peticiones dentro de MAVProxy.	48
3.6.	Mensaje que muestra el estado de la petición de altura y <i>heading</i> al servidor MAVProxy.	49
3.7.	Puntos de interés mostrados en el plano cartesiano.	51
3.8.	Ajuste de ángulo de los puntos de interés P y Q.	52
3.9.	Quadricoptero sobre Cancha de Fútbol.	55
3.10.	Apertura de la cámara web, que es $\theta = 75^\circ$	56
3.11.	Cancha de Fútbol con puntos de prueba.	57
4.1.	Distribución de poleras en primer intento.	59
4.2.	Fotografía sobre la cancha de fútbol con puntos de prueba a ochenta metros de altura.	60
4.3.	Fotografía procesada con detección de puntos de interés.	60
4.4.	Fallida detección de puntos en la imagen a ochenta metros de altura.	61
4.5.	Fotografía sobre la cancha de fútbol con puntos de prueba a treinta metros de altura.	62
4.6.	Fotografía umbralizada sobre la cancha de fútbol con puntos de prueba a treinta metros de altura.	63
4.7.	Fotografía umbralizada con puntos de interés detectados.	64
4.8.	Resultado del cálculo de puntos GPS mediante el algoritmo.	65
A.1.	Formato de archivo con coordenadas GPS.	74

ÍNDICE DE TABLAS

página