



# ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS  
INNOVACIÓN PARA LA EXCELENCIA

**DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA**

**CARRERA DE INGENIERIA EN ELECTRÓNICA Y  
TELECOMUNICACIONES**

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL  
TÍTULO DE INGENIERO EN ELECTRÓNICA Y  
TELECOMUNICACIONES**

**TEMA: DISEÑO DE LA INTERFAZ DE CONTROL PARA UN  
CUADRICÓPTERO CON EL DISPOSITIVO KINECT**

**AUTOR: MENDOZA AYO DAVID ARMANDO**

**DIRECTOR: ING. ALULEMA, DARWIN**

**SANGOLQUÍ**

**2016**

*Certificado de tutoría*

## DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA

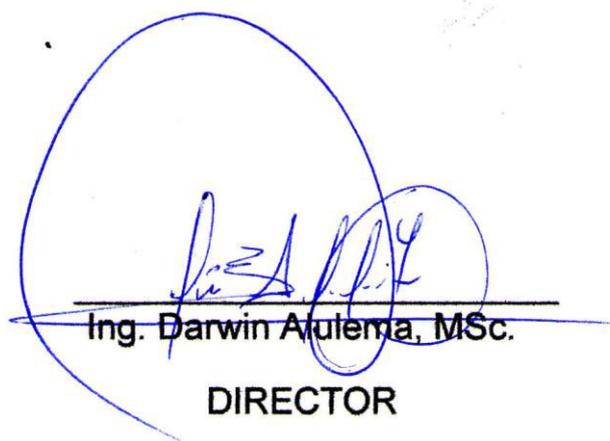
### CARRERA DE INGENIERIA EN ELECTRÓNICA Y TELECOMUNICACIONES

#### CERTIFICACIÓN

Certifico que el trabajo de titulación, “**DISEÑO DE LA INTERFAZ DE CONTROL PARA UN CUADRICÓPTERO CON EL DISPOSITIVO KINECT**” realizado por el señor **DAVID ARMANDO MENDOZA AYO**, 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 Fuerzas Armadas ESPE, por lo tanto me permito acreditarlo y autorizar al señor **DAVID ARMANDO MENDOZA AYO** para que lo sustente públicamente.

Sangolquí, 19 de diciembre de 2016.

Atentamente,



Ing. Darwin Aulema, MSc.  
DIRECTOR

*Autoría de Responsabilidad*

## **DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA**

### **CARRERA DE INGENIERIA EN ELECTRÓNICA Y TELECOMUNICACIONES**

#### **AUTORÍA DE RESPONSABILIDAD**

Yo, DAVID ARMANDO MENDOZA AYO con cédula de identidad N° 1715182810 declaro que este trabajo de titulación “DISEÑO DE LA INTERFAZ DE CONTROL PARA UN CUADRICÓPTERO CON EL DISPOSITIVO KINECT”, 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 declaro que este trabajo es de mi autoría, en virtud de ello me declaro responsable del contenido, veracidad y alcance de la investigación mencionada.

Sangolquí, 19 diciembre de 2016.



---

David Armando Mendoza Ayo

1715182810

*Autorización de publicación*

## **DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA**

### **CARRERA DE INGENIERIA EN ELECTRÓNICA Y TELECOMUNICACIONES**

#### **AUTORIZACIÓN**

Yo, DAVID ARMANDO MENDOZA AYO, autorizo a la Universidad de las Fuerzas Armadas ESPE publicar en la biblioteca Virtual de la institución la presente trabajo de titulación **“DISEÑO DE LA INTERFAZ DE CONTROL PARA UN CUADRICÓPTERO CON EL DISPOSITIVO KINECT”** cuyo contenido, ideas y criterios son de mi autoría y responsabilidad.

Sangolquí, 19 de diciembre de 2016.



---

David Armando Mendoza Ayo

1715182810

## DEDICATORIA

Esta tesis se la dedico a mi Dios quién supo guiarme por el camino del bien, supo darme fuerzas para seguir adelante y no desmayar en los problemas que se presentaban, enseñándome a encarar los infortunios sin perder nunca la dignidad ni desfallecer en el intento.

A mi familia quienes por ellos soy lo que soy. Para mis padres por su apoyo, consejos, comprensión, amor, ayuda en los momentos difíciles, y por ayudarme con los recursos necesarios para estudiar. Me han dado todo lo que soy como persona, mis valores, mis principios, mi carácter, mi empeño, mi perseverancia, mi coraje para conseguir mis objetivos.

*David*

## **AGRADECIMIENTO**

Gracias a Dios por permitirme tener y disfrutar a mi familia, gracias a mi familia por apoyarme en cada decisión y proyecto no ha sido sencillo el camino hasta ahora, pero gracias a sus aportes, a su amor, a su inmensa bondad y apoyo, lo complicado de esta meta se ha notado menos. Les agradezco, y hago presente mi gran afecto hacia ustedes, mi hermosa familia,

Gracias a mi universidad por haberme permitido formarme en ella, gracias a mi tutor por apoyarme y creer en la realización de este proyecto, gracias a todas las personas que fueron partícipes de este proceso, ya sea de manera directa o indirecta, gracias a todos ustedes, este es un momento muy especial que espero, perdure en el tiempo, no solo en la mente de las personas a quienes agradecí, sino también a quienes invirtieron su tiempo para darle una mirada a mi proyecto de titulación.

*David*

## ÍNDICE GENERAL

<b>CERTIFICACIÓN</b> .....	<b>ii</b>
<b>AUTORÍA DE RESPONSABILIDAD</b> .....	<b>iii</b>
<b>AUTORIZACIÓN</b> .....	<b>iv</b>
<b>DEDICATORIA</b> .....	<b>v</b>
<b>AGRADECIMIENTO</b> .....	<b>vi</b>
<b>ÍNDICE</b> .....	<b>vii</b>
<b>ÍNDICE DE TABLAS</b> .....	<b>x</b>
<b>ÍNDICE DE FIGURAS</b> .....	<b>xi</b>
<b>RESUMEN</b> .....	<b>xiv</b>
<b>ABSTRACT</b> .....	<b>xv</b>
<b>CAPÍTULO I</b> .....	<b>1</b>
<b>INTRODUCCIÓN</b> .....	<b>1</b>
1.1. ANTECEDENTES.....	1
1.2. OBJETIVOS.....	3
1.2.1 Objetivo General.....	3
1.2.2 Objetivos específicos.....	3
1.3. JUSTIFICACIÓN E IMPORTANCIA.....	4
1.4. ALCANCE DEL PROYECTO.....	6
<b>CAPÍTULO II</b> .....	<b>8</b>
<b>BASE TEÓRICA</b> .....	<b>8</b>
2.1 ESTUDIO DEL ESTADO DEL ARTE.....	8
2.1.1 GRTool.....	12
2.1.2 Sistemas de Visión para el desarrollo de la robótica situada aplicada a drones(cuadricópteros).....	13
2.1.3 Mecanismos de Navegación Autónomos de cuadricópteros en el contexto de robótica situada.....	13
2.1.4 Mecanismos de Navegación Híbridos de cuadricópteros en el contexto de la robótica situada con el empleo de sensor de velocidad.....	13
2.1.5 Ambiente de robótica situada aplicada a drones.....	14
2.2 MARCO CONCEPTUAL.....	14
2.2.1 Sistemas Teleoperados.....	14
2.2.2 Elementos de un sistema teleoperado.....	15
2.2.3 Aplicaciones de la teleoperación.....	16
2.2.4 Vehículo aéreo no tripulado.....	20
2.2.5 ¿Qué es un cuadricóptero?.....	23
2.3 CARÁCTERÍSTICAS TÉCNICAS DEL CUADRICÓPTERO.....	27
2.3.1 Phantom 3 Standard.....	27
2.3.2 Principales Características.....	27
2.3.3 Diagrama de la aeronave.....	28
2.3.5 Especificaciones Técnicas.....	29
2.3.6 Preparación previa al vuelo.....	31
2.3.7 Indicadores LED.....	34

2.3.8 Arranque y parada de los motores .....	34
2.3.10 Aplicación DJI GO.....	36
2.4 KINECT .....	42
2.4.1 Composición.....	43
2.4.2 Dimensiones.....	44
2.4.3 Funcionamiento.....	44
2.4.4 Especificaciones Técnicas .....	47
2.4.5 Controladores y Plataformas .....	47
2.4.6 Campo de Visión .....	51
2.4.7 Sistema de reconocimiento de gestos.....	53
<b>CAPÍTULO III.....</b>	<b>55</b>
<b>IMPLEMENTACIÓN DE LA INTERFAZ DE CONTROL .....</b>	<b>55</b>
3.1 ANÁLISIS PARA LA IMPLEMENTACIÓN DE LA INTERFAZ .....	55
3.1.1 Descripción del sistema.....	55
3.1.2 Requisitos del sistema .....	55
3.1.3 Diagrama de bloques del sistema .....	56
3.2 ANÁLISIS DE LOS TIPOS DE MOVIMIENTOS PARA CONTROLAR EL CUADRICÓPTERO.....	57
3.2.1 Análisis biomecánico.....	57
3.2.2 Modelo biomecánico .....	58
3.2.3 Segmentación del Cuerpo Humano .....	59
3.2.4 Captura de movimiento .....	61
3.3 DISEÑO DEL HARDWARE PARA EL PROCESAMIENTO DE INFORMACIÓN.....	66
3.3.1 Etapa IN DC FUENTE DE PODER .....	67
3.3.2 Etapa USB TTL SERIAL.....	68
3.3.3 Etapa BOTON DE EMERGENCIA .....	68
3.3.4 Etapa POTENCIOMETROS PLACA DE CONTROL.....	69
3.3.5 Etapa POTENCIOMETROS CONTROLADOR REMOTO.....	69
3.3.6 Etapa CONTROL POTENCIOMETROS DIGITALES .....	70
3.3.7 Etapa SEGUIDORES DE VOLTAJE .....	70
3.3.8 Etapa CONEXIÓN DE RELES .....	71
3.3.9 Etapa MICROCONTROLADOR PIC 18F452 .....	72
3.3.10 Diseño de la placa de control .....	73
3.3.11 Dimensiones de la caja para la placa de control .....	75
3.3.12 Implementación del circuito electrónico.....	75
3.4 DISEÑO DE LA ETAPA DE PROCESAMIENTO DE INFORMACIÓN.....	77
3.4.1 Requisitos de la Etapa de Procesamiento de Información .....	77
3.4.2 Diagrama de bloques de la Etapa de Procesamiento de Información .....	77
3.4.3 Requisitos de la Interfaz Gráfica de Usuario (GUI) .....	78
3.4.4 Diagrama de Flujo de la Interfaz Gráfica de Usuario (GUI) .....	81
3.4.5 Diagrama de Casos de uso de la Interfaz Gráfica de Usuario.....	82
3.4.6 Diseño de diagramas de flujo para la aplicación .....	86

3.5	DISEÑO DE LA INTERFAZ GRÁFICA DE USUARIO (GUI) .....	96
3.5.1	Diseño de la Ventana “Menú de Acceso” .....	96
3.5.2	Diseño de las Ventanas de “Ayuda” .....	97
3.5.3	Diseño de las Ventanas de “Información” .....	98
3.5.4	Diseño de la Ventana “Interfaz” .....	99
3.5.5	Implementación de la Interfaz Gráfica de Usuario (GUI) .....	100
3.5.6	Elementos del sistema para la interfaz de control .....	104
	<b>CAPÍTULO IV .....</b>	<b>106</b>
	<b>PRUEBAS DE FUNCIONAMIENTO .....</b>	<b>106</b>
4.1	PRUEBAS DE FUNCIONAMIENTO DEL DISPOSITIVO KINECT ...	106
4.2	PRUEBAS DE FUNCIONAMIENTO DE LA APLICACIÓN.....	107
2.1.6	Ventana Menú de Acceso .....	107
4.2.1	Prueba de la interfaz sin conexión del dispositivo Kinect .....	110
2.1.7	Prueba de la interfaz sin conexión de la placa de control .....	111
2.1.8	Prueba de la interfaz conectado el Kinect y la placa de control .....	112
4.3	PRUEBAS DE COMUNICACIÓN CON EL CUADRICÓPTERO .....	117
4.3.1	Escenario de Evaluación .....	118
4.3.2	Evaluación del sistema.....	119
4.4	RESULTADOS Y ANÁLISIS.....	120
	<b>CAPÍTULO V .....</b>	<b>123</b>
	<b>CONCLUSIONES Y RECOMENDACIONES.....</b>	<b>123</b>
5.1	CONCLUSIONES.....	123
5.2	RECOMENDACIONES .....	125
	<b>BIBLIOGRAFÍA .....</b>	<b>128</b>

## ÍNDICE DE TABLAS

<b>Tabla 1</b>	Usos civiles de los UAV.....	20
<b>Tabla 2</b>	Ventajas y desventajas de los UAV.....	22
<b>Tabla 3</b>	Especificaciones Técnicas del cuadricóptero.....	30
<b>Tabla 4</b>	Especificaciones de la aplicación DJI GO.....	36
<b>Tabla 5</b>	Botones e íconos de la vista de la cámara.....	38
<b>Tabla 6</b>	Características del dispositivo Kinect.....	47
<b>Tabla 7</b>	Características de los controladores OpenNI y Microsoft SDK.....	49
<b>Tabla 8</b>	Segmentos del modelo biomecánico.....	58
<b>Tabla 9</b>	Segmentos que participan en la movilidad.....	59
<b>Tabla 10</b>	Características de los modelos de Kinect.....	62
<b>Tabla 11</b>	Comparación de los SDK Microsoft vs Open Source.....	63
<b>Tabla 12</b>	Componentes etapa IN DC FUENTE DE PODER.....	67
<b>Tabla 13</b>	Componentes etapa POTENCIOMETROS PLACA DE CONTROL ELECTRÓNICA.....	69
<b>Tabla 14</b>	Componentes etapa POTENCIOMETROS CONTROLADOR REMOTO.....	70
<b>Tabla 15</b>	Componentes etapa CONEXIÓN DE RELÉS.....	72
<b>Tabla 16</b>	Componentes etapa MICROCONTROLADOR PIC 18F452.....	73
<b>Tabla 17</b>	Elementos de la Placa de Control.....	74
<b>Tabla 18</b>	Nomenclatura de los Tipos de Requisitos de la Aplicación.....	79
<b>Tabla 19</b>	Requisitos Funcionales de la Aplicación.....	80
<b>Tabla 20</b>	Requisitos Funcionales de la Opción “Interfaz”.....	80
<b>Tabla 21</b>	Requisitos No Funcionales de la Aplicación.....	80
<b>Tabla 22</b>	Modelo de Casos de Uso “Iniciar Aplicación”.....	83
<b>Tabla 23</b>	Modelo de Casos de Uso “Menú de Acceso”.....	83
<b>Tabla 24</b>	Modelo de Casos de Uso “Ayuda”.....	83
<b>Tabla 25</b>	Modelo de Casos de Uso “Información”.....	84
<b>Tabla 26</b>	Modelo de Casos de Uso “Interfaz”.....	84
<b>Tabla 27</b>	Modelo de Casos de Uso “Genera los movimientos para controlar el cuadricóptero”.....	85
<b>Tabla 28</b>	Modelo de Casos de Uso “Reconoce y procesa los movimientos realizados por el usuario”.....	85
<b>Tabla 29</b>	Modelo de Casos de Uso “Cerrar Aplicación”.....	86
<b>Tabla 30</b>	Funciones para reconocer el flujo de esqueletos.....	90
<b>Tabla 31</b>	Controladores de las ventanas de la aplicación.....	101
<b>Tabla 32</b>	Pruebas de altura y distancia.....	107
<b>Tabla 33</b>	Pruebas de los botones de la aplicación.....	110
<b>Tabla 34</b>	Pruebas de funcionamiento Interfaz con diferentes usuarios.....	118

## ÍNDICE DE FIGURAS

<b>Figura 1</b>	Robots Roomba con visión 3D.....	10
<b>Figura 2</b>	Manipulador móvil MANFRED-2 .....	11
<b>Figura 3</b>	Esquema conceptual de <i>GRTool</i> .....	12
<b>Figura 4</b>	Flujo de información en un sistema teleoperado.....	15
<b>Figura 5</b>	Vehículo explorador Curiosity .....	17
<b>Figura 6</b>	iRobot PackBot. ....	18
<b>Figura 7</b>	Sistema Quirúrgico Da Vinci .....	19
<b>Figura 8</b>	Tipos de aeronaves UAVS.....	21
<b>Figura 9</b>	Cuadricóptero Phantom 3 Standard.....	24
<b>Figura 10</b>	Descripción de los grados de libertad .....	25
<b>Figura 11</b>	Dimensiones del cuadricóptero Phantom 3 Standard .....	28
<b>Figura 12</b>	Elementos del cuadricóptero Phantom 3 Standard.....	28
<b>Figura 13</b>	Dimensiones del controlador remoto.....	29
<b>Figura 14</b>	Elementos del controlador remoto .....	29
<b>Figura 15</b>	Fijación de las hélices .....	32
<b>Figura 16</b>	Calibración Horizontal .....	33
<b>Figura 17</b>	Calibración Vertical .....	33
<b>Figura 18</b>	Indicadores LED .....	34
<b>Figura 19</b>	Arranque de los motores.....	35
<b>Figura 20</b>	Métodos para detener los motores .....	35
<b>Figura 21</b>	Vista de la cámara .....	38
<b>Figura 22</b>	Elementos del sensor Kinect.....	43
<b>Figura 23</b>	Dimensiones del sensor Kinect.....	44
<b>Figura 24</b>	Esquema Cámara Kinect 1.0 .....	45
<b>Figura 25</b>	Modos de distancia admitidos por Kinect.....	46
<b>Figura 26</b>	Arquitectura del SDK de Microsoft .....	50
<b>Figura 27</b>	Campo de Visión Horizontal y Vertical.....	51
<b>Figura 28</b>	Campo de visión vertical del sensor Kinect.....	52
<b>Figura 29</b>	Rango de Profundidad Modo Near .....	52
<b>Figura 30</b>	Rango de Profundidad Modo Default.....	53
<b>Figura 31</b>	Desplazamiento de la mano derecha.....	53
<b>Figura 32</b>	Diagrama de bloques del sistema .....	56
<b>Figura 33</b>	División de los planos anatómicos .....	57
<b>Figura 34</b>	Estructura de un modelo biomecánico .....	58
<b>Figura 35</b>	Segmentos del cuerpo humano .....	60
<b>Figura 36</b>	Segmentación actual del cuerpo humano .....	60
<b>Figura 37</b>	Diagrama de captura de movimientos.....	64
<b>Figura 38</b>	Articulaciones disponibles en el SDK de Microsoft .....	65
<b>Figura 39</b>	Plano de Referencia de las Coordenadas de las Articulaciones..	65
<b>Figura 40</b>	Diseño Electrónico de la conexión entre elementos de la placa ..	66
<b>Figura 41</b>	Diseño Electrónico de la etapa de alimentación .....	67
<b>Figura 42</b>	Diseño Electrónico de la etapa USB TTL SERIAL .....	68
<b>Figura 43</b>	Diseño Electrónico de la etapa BOTON DE EMERGENCIA.....	68
<b>Figura 44</b>	Diseño Electrónico de la etapa POTENCIOMETROS PLACA DE CONTROL .....	69
<b>Figura 45</b>	Diseño Electrónico de la etapa POTENCIOMETROS CONTROLADOR REMOTO .....	69

<b>Figura 46</b>	Diseño Electrónico de la etapa CONTROL POTENCIOMETROS DIGITALES .....	70
<b>Figura 47</b>	Diseño Electrónico de la etapa SEGUIDORES DE VOLTAJE .....	71
<b>Figura 48</b>	Diseño Electrónico de la etapa CONEXIÓN DE RELES.....	71
<b>Figura 49</b>	Diseño Electrónico de la etapa MICROCONTROLADOR PIC 18F452.....	72
<b>Figura 50</b>	Diseño de la placa de control .....	73
<b>Figura 51</b>	Vista 3D del diseño completo de la tarjeta de control .....	75
<b>Figura 52</b>	Dimensiones de la caja para la tarjeta de control .....	75
<b>Figura 53</b>	Vista la tarjeta de control implementada .....	76
<b>Figura 54</b>	Caja de control implementada .....	76
<b>Figura 55</b>	Diagrama de bloques de la etapa de procesamiento de información .....	77
<b>Figura 56</b>	Hardware para el procesamiento de información.....	78
<b>Figura 57</b>	Diagrama de Flujo de la Aplicación.....	81
<b>Figura 58</b>	Diagrama de Casos de Uso .....	82
<b>Figura 59</b>	Diagrama de flujo del método crear_kinect.....	87
<b>Figura 60</b>	Algoritmo utilizado para habilitar el flujo de datos .....	88
<b>Figura 61</b>	Nombres de las Articulaciones Reconocidas por el dispositivo Kinect.....	89
<b>Figura 62</b>	Sentencias para definir las coordenadas de las articulaciones ....	89
<b>Figura 63</b>	Diagrama de Flujo del Algoritmo Utilizado para el Control de los Datos de Esqueletos .....	90
<b>Figura 64</b>	Diagrama de flujo del método agregarLinea .....	91
<b>Figura 65</b>	Imagen esqueletizada .....	91
<b>Figura 66</b>	Diagrama de flujo para reconocer los movimientos de las manos	92
<b>Figura 67</b>	Movimientos mano derecha .....	93
<b>Figura 68</b>	Movimientos mano izquierda .....	94
<b>Figura 69</b>	Diagrama de flujo de la Clase drone_comm .....	94
<b>Figura 70</b>	Diagrama de flujo del reconocimiento de voz .....	95
<b>Figura 71</b>	Sentencias para reconocimiento de voz .....	96
<b>Figura 72</b>	Diseño de la ventana “Menú de Acceso” .....	97
<b>Figura 73</b>	Diseño de las ventanas de “Ayuda” .....	98
<b>Figura 74</b>	Diseño de las ventanas de “Información” .....	99
<b>Figura 75</b>	Diseño de la ventana “Interfaz” .....	100
<b>Figura 76</b>	Ventana “Menú de Acceso” .....	101
<b>Figura 77</b>	Ventanas de “Ayuda” .....	102
<b>Figura 78</b>	Ventanas de “Información” .....	103
<b>Figura 79</b>	Ventana “Interfaz” .....	104
<b>Figura 80</b>	Elementos del sistema.....	105
<b>Figura 81</b>	Pruebas de altura y distancia .....	106
<b>Figura 82</b>	Pruebas botón Ayuda.....	108
<b>Figura 83</b>	Pruebas botón Información .....	109
<b>Figura 84</b>	Pruebas del botón Interfaz .....	109
<b>Figura 85</b>	Pruebas de los botones de la ventana menú acceso.....	110
<b>Figura 86</b>	Prueba de detección del dispositivo Kinect.....	111
<b>Figura 87</b>	Prueba de excepción dispositivo Kinect.....	111
<b>Figura 88</b>	Prueba de detección de la placa de control .....	111

<b>Figura 89</b>	Prueba de excepción placa de control .....	112
<b>Figura 90</b>	Prueba Interfaz sin conexión placa de control .....	112
<b>Figura 91</b>	Prueba de reconocimiento del Kinect y la placa de control.....	113
<b>Figura 92</b>	Prueba de la interfaz conectado el Kinect y la placa de control .	113
<b>Figura 93</b>	Prueba de la interfaz con el usuario.....	114
<b>Figura 94</b>	Prueba de la interfaz comando de voz CENTROS .....	114
<b>Figura 95</b>	Prueba de la interfaz comando de voz PRENDER .....	115
<b>Figura 96</b>	Prueba de la interfaz en estado Encendido .....	115
<b>Figura 97</b>	Prueba de la interfaz en estado Funcionando.....	116
<b>Figura 98</b>	Prueba de la interfaz en estado Apagado .....	116
<b>Figura 99</b>	Escenario de la prueba .....	117
<b>Figura 100</b>	Dimensiones del Espacio controlado .....	118
<b>Figura 101</b>	Modelo de la encuesta .....	119
<b>Figura 102</b>	Resultados de la encuesta.....	120
<b>Figura 103</b>	Progreso de los usuarios en el control del cuadricóptero.....	121

## RESUMEN

El presente proyecto describe el diseño e implementación de una interfaz de control para un cuadricóptero mediante el uso del dispositivo Kinect a través de una aplicación desarrollada en Visual Studio 2013 con el lenguaje de programación C#- WPF. La interfaz de control funciona con los movimientos que el operador realiza con sus manos para que el cuadricóptero pueda elevarse, descender, desplazarse a la izquierda, desplazarse a la derecha, hacia adelante y hacia atrás para el desarrollo del proyecto se ha utilizado un dispositivo Kinect para Xbox 360, el software oficial (SDK de Microsoft) que permite controlar el sensor, el sistema operativo Windows 7, el cuadricóptero DJI Phantom 3 Standard y un microcontrolador AVR para el manejo de señales desde el computador hacia el cuadricóptero. En el análisis y evaluación se contemplan los requisitos funcionales y no funcionales que debe cumplir la aplicación, ya que el sistema requiere que el dispositivo Kinect se encuentre dentro de un ambiente controlado para la toma de señales, esto implica que no debe existir exposición al sol, ambientes de lluvia y luz reflectante hacia la cámara. La aplicación recepta los datos del dispositivo Kinect, procesa la información en una computadora la cual a través de la red inalámbrica propia del cuadricóptero envía las señales para poder controlar el movimiento del mismo. Tomado en cuenta las consideraciones de diseño se implementó la interfaz de control y los resultados de las pruebas realizadas demostraron que el proyecto cumple con el objetivo principal de manejar el cuadricóptero.

### **PALABRAS CLAVE:**

- **DISPOSITIVO KINECT**
- **C# - WPF**
- **CUADRICÓPTERO**
- **SDK**
- **INTERFAZ**

## **ABSTRACT**

The present project describes the design and implementation of a control interface for a quadricopter using the Kinect device through an application developed in Visual Studio 2013 with the programming language C#-WPF. The control interface works with the movements that the operator performs with his hands so that the copter can rise, descend, move left, move right, forward and backward, for the development of the project has been used a Kinect device for Xbox 360 with your respective adapter, official software (Microsoft SDK) allowing to control sensor, the operating system Windows 7, the quadricopter DJI Phantom 3 Standard and an AVR Microcontroller for the handling of signals from the computer to the quadricopter. Functional and non-functional requirements that must comply with the application, since the system requires that the Kinect device is within an environment controlled signals decision referred to in the analysis and evaluation, this implies that not there must be exposure to sun, rain & reflecting light towards the camera. The application receives data from the Kinect device, processes the information into a computer, which sends signals to control the movement through the wireless network of the quadricopter. Taken into account the considerations of design control interface was implemented and results of tests carried out showed that the project complies with the objective of driving the quadricopter.

### **KEY WORDS:**

- **KINECT DEVICE**
- **C# - WPF**
- **QUADRICOPTER**
- **SDK**
- **INTERFACE**

# CAPÍTULO I

## INTRODUCCIÓN

La tecnología a lo largo de su desarrollo ha tenido como objeto la gestión de los recursos, como mecanismos de ayuda y soporte a los diferentes modos de vida del ser humano. En este sentido, la tecnología, a lo largo de su evolución, se ha interiorizado como parte de las necesidades humanas, según la cual se promueven actividades que permiten que la vida del ser humano, no sólo sea más asequible, sino que ha acortado el tiempo y espacio para los procesos de transferencia de la información.

En este sentido, este capítulo expone los antecedentes tecnológicos desde los que parte el objeto de la investigación, realizando un breve recorrido de la evolución de los vehículos aéreos no tripulados, así como una breve exposición sobre los objetivos de investigación. De esta forma, se presenta al lector la justificación que motiva a esta investigación, así como sus posibles alcances teóricos a la vez que prácticos. Finalmente, se detalla el alcance global de este proyecto de investigación.

### 1.1. ANTECEDENTES

La palabra dron sirve “para referirse a una ‘aeronave no tripulada’”. En principio, este término se relacionaba a aparatos, básicamente, de uso militar y con aspecto similar al de un avión” (dron, adaptación al español de drone, 2014).

La investigación de Cortés sostiene que el ejemplo de mayor data de un dron nace a lo largo de la primera guerra mundial, mismo que fue empleado a lo largo de la segunda guerra mundial con motivo de adiestrar a quienes tenían a su cargo el empleo de bombas antiaéreas, así como para atacar o hacer reconocimiento de territorio enemigo (Cortes, s/a, web). En ambos casos, el objeto primordial de su uso fue para proteger a los soldados durante misiones de guerra, mediante la implementación de la modalidad de

objetos no tripulados. Sin embargo, a finales del siglo XX apenas se inició la operación de Drones. Esta operación, fundamentalmente, se efectuó mediante un control a distancia para controlar la autonomía del dron. De esta forma, este tipo de tecnología fue empleada en varias épocas beligerantes del siglo pasado, con especial atención en la guerra del Golfo (Ibídem:web). Entre las principales características desarrolladas, por ejemplo, se identifica el gran potencial que tienen, puesto que su implementación y uso de sus herramientas ha permitido la maximización de los objetivos de acción planteados durante las intervenciones, así como el acceso a escenarios de riesgo y de alto grado de complicación para seres humanos. Del mismo modo, los drones permiten una recolección y transferencia de datos respecto a un objeto definido. Durante los últimos años, este proceso resulta más efectivo, puesto que los drones actuales presentan radios de comunicación y transferencia de data con altos niveles de protección y seguridad.

De esta forma, y gracias al desarrollo tecnológico, la ciencia ha desarrollado varios tipos de vehículos diferentes al sistema de un avión tradicional. Para este tipo de vehículos se puede emplear expresiones más genéricas como Vehículos Aéreos No Tripulados, conocido como *Unmanned Aerial Vehicle* o UAV –siglas en inglés–. Actualmente, se emplean dos modelos definidos que pueden ser controlados a distancia. Por un lado, el vehículo aéreo de hélice fija que, básicamente gracias a la capacidad de poder planear, tiene un consumo bastante bajo y, por el otro, el tipo de vehículo aéreo que tiene más de una hélice que, en contraste con el anterior, puede mantenerse estático en el aire (Oñate de Mora, 2015, pp. 49-57).

Durante el siglo XXI han surgido considerables avances y nuevas tecnologías que han revolucionado la manera de comunicación y telecomunicación. Uno de estos avances es el cuadricóptero. Éste tiene una gran capacidad de maniobra, de transportar objetos de un lugar a otro, de acceder a lugares remotos. El cuadricóptero puede retransmitir fotos y videos de planos de difícil acceso, especialmente en el mar o lugares inhóspitos. Otros tipos de uso que se le da al cuadricóptero es como

salvavidas, capturador de ortofotos (para mapas), como emisor y receptor de señal, vigilancia y monitoreo, y, últimamente, se ha empleado en el entretenimiento (Ibídem, pp.49-51). Todo esto ha definido la importante utilidad de los cuadricópteros debido a sus costos menos elevados.

Los drones pueden ser manejados por un control remoto, controlados por personas o sistemas prácticamente autónomos. Es decir, podrían funcionar sin necesidad de personal, sólo necesitando de ordenadores, avanzados algoritmos y medios de detección y navegación para trasladarse de un punto a otro (Dans, 2014, web).

Actualmente, dentro del Plan del Buen Vivir del Ecuador se establece la implementación de políticas para democratizar la "...prestación de servicios públicos de telecomunicaciones y de tecnologías de información y comunicación o TICs" (Mora, 2015, p. 39). En adición, se puede incluir el espectro radial para medios auditivos como audiovisuales y de comunicación en general, profundizando su uso y acceso universal con la meta de incrementar la media de ciudadanos utilizando TIC al 82% (Plan Nacional del buen Vivir, 2013). En este sentido, el proyecto de investigación tiene como propósito justificar que generación de nuevos procesos y herramientas tecnológicas para gestionar la comunicación, que a más de ser elementos didácticos educativos, se pueden convertir en herramientas intelectuales que generen acciones oportunas y pertinentes dentro de la academia, el sector privado y el público.

## **1.2. OBJETIVOS**

### **1.2.1 Objetivo General**

- Implementar una interfaz basada en el dispositivo Kinect para controlar el movimiento de un cuadricóptero

### **1.2.2 Objetivos específicos**

- Analizar el estado del arte respecto al dispositivo Kinect.

- Investigar el funcionamiento y operación del cuadricóptero para el desarrollo de la aplicación.
- Diseñar la interfaz de control que permita el manejo del cuadricóptero.
- Comunicar el dispositivo Kinect con el cuadricóptero a través de una placa que utilice un microcontrolador AVR para que se envíen los datos a la red inalámbrica.
- Realizar pruebas de funcionamiento de la interfaz de control con el cuadricóptero.

### **1.3. JUSTIFICACIÓN E IMPORTANCIA**

La empresa Ingenius Worlds, dentro de su misión, busca satisfacer los requerimientos y superar las expectativas de sus clientes brindando servicio con soluciones tecnológicas e innovadoras (2016,web). La idea principal que tiene la empresa es cubrir las necesidades concretas de sus clientes en varias dimensiones, mediante el empleo de metodologías aplicativas de inteligencia artificial al comercializar un sistema de monitoreo. Este sistema se basa en el empleo del cuadricóptero, con el objetivo de captar imágenes o grabar videos en espacios abiertos.

De acuerdo con Intelligencia Dynamics, "...existe un amplio mercado para estas aplicaciones [...como el...] recuento de árboles en una finca afecta a subvenciones, la realización de fotografías y vídeos publicitarios de carácter comercial para inmuebles, campos de golf, hoteles, otros" (2012, web). Mediante el uso de un cuadricóptero en este campo se consiguen las siguientes ventajas:

- "Reducción general de precio, en especial frente a alternativas pilotadas.

- Mejora de las prestaciones: mayor estabilidad (que permite mejores fotografías) sin renunciar al dinamismo y posibilidad de vuelo a bajas velocidades.
- Automatización del proceso cuando es necesario fotografiar un área más grande.
- Obtención de mapas 3D de un terreno.
- Reducción del tiempo de trabajo.
- Reducción de los efectivos humanos necesarios” (Ibídem, web).

En este sentido, radica la importancia de desarrollar nuevos recursos con diferentes implementos, con motivo de promover prototipos cada vez más avanzados a la vez que económicos. Así, este proyecto busca proporcionar un nuevo procedimiento para captar fotografías, videos y cartografía aérea. Está dirigido a instituciones públicas o privadas que necesiten un sistema de navegación de un cuadricóptero guiado por el movimiento de las manos para diferentes campos. Según lo resumen Inteligencia Dynamics, este tipo de vehículo aéreo no tripulado se puede utilizar en:

- “Medio Ambiente: Parametrización del índice de contaminación lumínica para elaborar mapas de polución lumínica y monitorizar la eficiencia de medidas ecoenergéticas; control y seguimiento de accidentes industriales con vertidos tóxicos en medios terrestres; control de áreas de depósito y almacenaje de residuos industriales y de su tratamiento.
- Agricultura: Control y monitorización del estado de los cultivos mediante imágenes multispectrales, control de la eficiencia de regadíos. Conteo y supervisión de producción agrícola subvencionada (por ejemplo, número de árboles).
- Geología: Realización de mapas geológicos sedimentológicos, mineralógicos y geofísicos, control y monitorización de explotaciones mineras y su impacto ambiental: movimientos de tierras, producción de áridos, residuos metálicos, balsas de decantación, etc.
- Construcción e inspecciones: Inspección de obras desde el aire. Estimación de impacto visual de grandes obras.
- Control y análisis de multitudes: Manifestaciones, conciertos, etc.” (Ibídem, web).

En este sentido, uno de los problemas que repercute directamente en el cultivo y producción agrícola, la caña de azúcar en el Ecuador por ejemplo, es la cantidad y variedad de enfermedades causadas por presencia de

plagas y maleza en los cultivos, mismas que deben ser gestionadas de tal forma que se promueva mayores niveles de efectividad dentro del período de producción. Así, el análisis agronómico actual carece de uso de nuevas tecnologías para la recolección de datos. Por tradición, la gestión se basa en estadísticas de cosechas anteriores, básicamente. De acuerdo con el trabajo de Carvajal y Ramírez es imprescindible, para evitar cualquier baja de producción, implementar dentro de los procesos de agrocalidad actividades de estudio y evaluación de la producción, con motivo de mejorar el acceso a datos e información de alta calidad que permita tomar decisiones pertinentes y oportunas (2003, p. 7).

Desde esta perspectiva, la necesidad de aplicación de este tipo de vehículos aéreos no tripulados permitiría simplificar el trabajo humano, así como resolver actividades de alto riesgo y complejidad para los usuarios. Una aplicación a desarrollarse a medida para una tarea concreta, por ejemplo, es utilizar el cuadricóptero en agricultura de precisión, con motivo de efectuar un análisis agronómico de seguimiento de cultivos. Esto sería de utilidad ya que, a través de la información obtenida de las imágenes y los videos, se lograría delimitar zonas a investigar así como en la medición de la variabilidad de lotes, o darle seguimiento a cultivos con el fin de evitar plagas y enfermedades, para realizar sectorización de malezas, entre otros usos.

#### **1.4. ALCANCE DEL PROYECTO**

En vista de la necesidad de desarrollar herramientas tecnológicas que permitan simplificar el trabajo humano, este proyecto pretende desarrollar un cuadricóptero controlado mediante el uso de recursos de fácil acceso y bajo coste. Para ello, se gestiona el diseño de una interfaz de control que funciona con los movimientos que el operador realiza con sus manos. Estos son detectados con el dispositivo Kinect. El proyecto presenta un sistema con un proceso de transmisión de video en tiempo real, el cual se visualiza en dispositivos móviles tales como tablets y celulares que tengan instalada la plataforma Android e IOS. La señal de video es enviada desde el

cuadricóptero al dispositivo móvil, a través de la aplicación propia de la aeronave mediante una red Wi-Fi. El sistema requiere que el dispositivo Kinect se encuentre dentro de un ambiente controlado para la toma de señales. Esto implica que no debe existir exposición al sol, ambientes de lluvia y luz reflectante hacia la cámara.

Se utiliza el cuadricóptero DJI Phantom 3 Standard, mismo que posee una cámara de video integrada. Éste soporta un peso máximo de 1280g y cuenta con un sistema inteligente de vuelo que informa continuamente el estado de la batería, la geolocalización y la altitud a la que se encuentra. La autonomía de vuelo es proporcionada por una batería de 4480 mAh, la que provee un tiempo de vuelo máximo de hasta 25 minutos (DJI, 2015, p.6). El cuadricóptero está catalogado dentro de la tercera generación de vehículos voladores, el cual es provisto por la empresa Ingenius Worlds,

La interfaz que controla y detecta los movimientos del sensor Kinect es programada en Microsoft Visual Studio C#, a través de las librerías que son de uso libre para el público y de los métodos propios del kit de desarrollo de software. La aplicación recepta los datos del dispositivo Kinect, procesa la información en una computadora la cual a través de la red inalámbrica propia del cuadricóptero envía las señales para poder controlar el movimiento del mismo.

Por tanto, el proyecto presentará, en su fase final, una interfaz de control, la cual se comunica con el cuadricóptero mediante una red inalámbrica, mismo que se opera en una frecuencia de 2,4 GHz (Ibídem, p. 7). Para este fin, se plantea el uso de un microcontrolador AVR para el manejo de señales desde el computador hacia el cuadricóptero. El cuadricóptero es controlado a través de gestos predefinidos con las manos. Los desplazamientos son de elevación, descendimiento, giro a la izquierda y derecha, frontal y posterior. Todos los movimientos son detectados por el dispositivo Kinect. Así, el proyecto se enfoca principalmente en el manejo del cuadricóptero.

## **CAPÍTULO II**

### **BASE TEÓRICA**

Con motivo de dar sustento teórico y las condiciones técnicas suficientes para dar soporte a la actual investigación, en este capítulo se presenta el marco conceptual relevante a partir desde el que se profundiza el análisis y estudio de la tecnología a emplear. En primer lugar, se inicia el capítulo con una breve narrativa del estado de arte actual, sobre el cuál versa la tecnología a emplearse. Seguido, se presenta un análisis extenso respecto a los sistemas teleoperados, a partir de los que se describen sus principales características, tipologías y aplicaciones. Luego, se realiza una valoración técnica sobre los diferentes tipos de vehículos aéreos no tripulados, específicamente, sobre cuadricópteros y sus funciones, modos de uso, entre otros. Finalmente, para concluir el capítulo se presenta el dispositivo Kinect, análisis desde el cual se pretende identificar el vínculo de uso y control respecto al cuadricóptero.

#### **2.1 ESTUDIO DEL ESTADO DEL ARTE**

Actualmente, la adopción de interfaces naturales entre el humano y una máquina es cada vez más importante, particularmente en el sector de la educación donde las herramientas y aplicaciones interactivas pueden facilitar la presentación y comprensión de conceptos complejos, estimular el trabajo colaborativo y mejorar las prácticas de enseñanza (Soares Christhope et al, 2003, p.1).

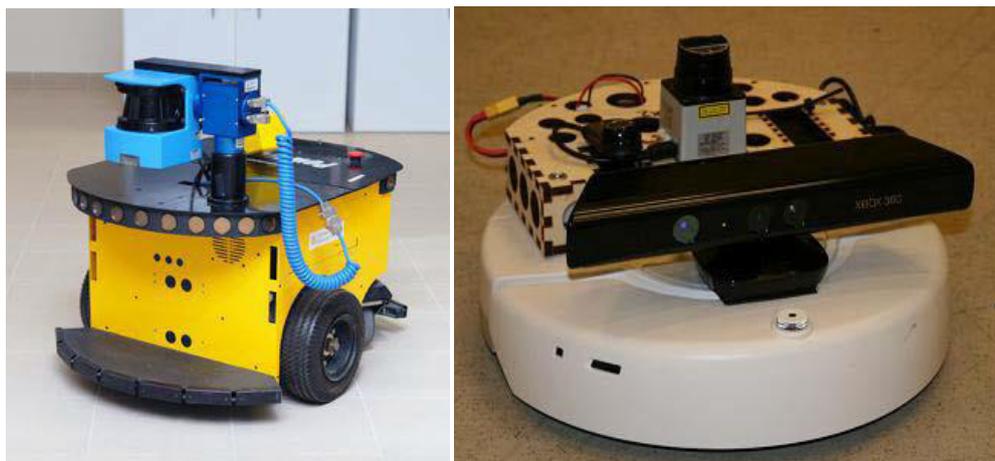
Gracias a la producción de bajo coste de dispositivos tecnológicos de interacción, el espectro de desarrollo de aplicaciones gesticulares se ha incrementado en los últimos años. Esto a decir verdad ya que promueve mayor facilidad en su acceso y uso. Tal es el caso de Kinect, desarrollado por la Microsoft. Este dispositivo sirve como un medio receptor de imágenes gesticulares en movimiento, efectuado por los usuarios. En este sentido, el

Kinect se ha convertido en una herramienta que ayuda a la interacción del usuario con la aplicación de forma efectiva, aunque "...el reconocimiento de gestos y la ejecución de las correspondientes acciones en la aplicación siguen siendo problemas aún no resueltos" (Ibañez & Fanaro, 2013, p. 321).

Desde esta perspectiva, es preciso mencionar que la evolución rápida que tiene la robótica hace necesario una automatización más estable, ya que debe adaptarse a los continuos avances de la ciencia y la sociedad, con motivo de brindar ayuda en diversos campos, tales como desastres naturales o conflictos de guerra, en donde el apoyo de los robots evita poner en riesgo la vida humana. En el contexto de ayuda a la sociedad aparece la necesidad de trabajar en la visión artificial. Hoy en día, el uso del láser 3D es un gran apoyo para el desarrollo de este campo, aunque se utilizan diferentes tipos de cámaras, tales como:

- La cámara del dispositivo Kinect que posee tecnología CMOS y sensores infrarrojos.
- La cámara con tecnología CCD (dispositivo de cargas acopladas), que posee "un sistema matricial de píxeles en los que se recoge la información de la luz y se transforma en una señal analógica, cada señal analógica del dispositivo es digitalizada por un único conversor" (Toribios Blázquez, 2012, p. 2).

En la Figura 1 se presenta dos robots cuyas tecnologías recolectan los datos en forma de nubes de puntos. El robot de la izquierda trabaja con un láser 3D, mientras que el robot de la derecha trabaja con el dispositivo Kinect.



**Figura 1 Robots Roomba con visión 3D**

Fuente: (Toribios Blázquez, 2012)

La complejidad de ambas tecnologías es similar, aunque el rango de trabajo del láser 3D (30 metros) supera en gran medida al del Kinect (3.5 metros). El precio de los dispositivos los diferencia. El Kinect es un 90% menor que de la tecnología láser. Así, y de acuerdo a las necesidades de la sociedad, el Kinect es el dispositivo de mejor opción para el desarrollo de aplicaciones. (Toribios Blázquez, 2012, p. 2).

Microsoft desarrolló el Kinect en un principio para su consola Xbox, con la finalidad que el jugador se convierta en el centro de mando dentro del videojuego. No obstante, como la tecnología se extiende cada día se lo empezó a utilizar dentro de la visión artificial en 3D. En la actualidad, por ejemplo, el dispositivo Kinect se emplea para el desarrollo de robots tipo *roomba*. Se lo usa para determinar zonas transitables, reconocimiento de personas y objetos. En la Figura 2 se muestra el robot MANFRED-2, que es capaz de abrir puertas, evitar obstáculos y recoger y manipular objetos.



**Figura 2 Manipulador móvil MANFRED-2**

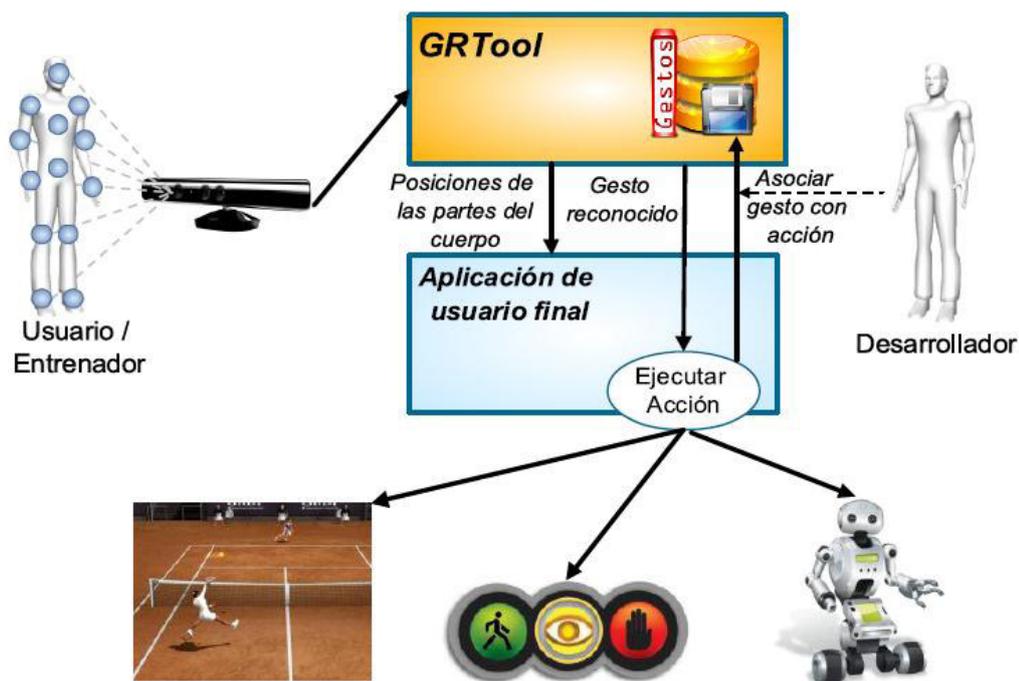
Fuente: (Toribios Blázquez, 2012)

Este tipo de dispositivos robotizados se han desarrollado gracias a un profundo análisis de gestos y movimientos humanos, tanto generales como particulares de varios sujetos. En este sentido, dicho estudio se ha convertido en el factor de apalancamiento para promover aplicaciones de realidad aumentada. Este tipo de aplicaciones, por tanto, reflejan movimientos replicados de forma simétrica y coordinados usuario-máquina, lo que permite una mayor calidad y cantidad de interacciones reales (Cuenca, Delgado, & Chuya, 2013, p. 8). Cabe mencionar que los Dispositivos de Interfaz Natural, NUI por sus siglas en inglés, cada vez es de mayor recursividad en las actividades humanadas, tales como los controles remotos 3D que capturan movimientos en espacios abiertos, entre otros. Entre estos, se identifica el Software Development Kit o SDK, con interfaces que amplían la cantidad de datos que reconoce el sistema. Las principales imágenes que reconoce el SDK son las RGD, de profundidad y las ubicadas

en espacios abiertos de hasta un máximo de 20 partes por cada cuerpo (Ibañez & Fanaro, 2013, p. 321)

### 2.1.1 GRTool

Ésta es un dispositivo reconocedor de movimientos y gestos. El reconocimiento se lo realiza mediante el recurso del Machine Learning. Su principal función es generar aprendizajes de nuevos movimientos y gestos sobre referencia a la data de movimientos registrados en la información general del dispositivo. Así, toda vez captura los movimientos los compara con sus datos, y de ser nuevos, entonces, los reconoce, clasifica y guarda. (Ibañez & Fanaro, 2013, p. 323). En la figura 3 se describe el proceso según el cual el GRTool interactúa con el usuario.



**Figura 3 Esquema conceptual de GRTool**

Fuente: (Ibañez & Fanaro, 2013)

Como se observa en la figura precedente, el GRTool reconoce los movimientos que recibe del Kinect. Una vez se reciben dichos movimientos se los reconoce según la base de datos por cada evento, mismo que se relaciona con uno de los registrados en aplicación. De esta forma, el GRTool

mantiene actualizado al avatar sobre la posición de los miembros del cuerpo que representa el sujeto frente al Kinect (Ibídem, p.324).

Según Lorusso (2014, p. 818), la línea de investigación para el desarrollo e innovación del dispositivo Kinect en conjunto con un cuadricóptero se concentran en:

### **2.1.2 Sistemas de Visión para el desarrollo de la robótica situada aplicada a drones (cuadricópteros)**

En este campo se identifica la implementación de sistemas de visión en tiempo real, utilizando algoritmos de reconocimientos de imágenes para la detección de elementos situados en el ambiente. De esta manera, se tiene total conocimiento de las posiciones de los objetos, pudiendo tomar decisiones sobre ellos.

### **2.1.3 Mecanismos de Navegación Autónomos de cuadricópteros en el contexto de robótica situada**

El trabajo se concentra en el cálculo de los recorridos, que realizará el cuadricóptero, y la utilización de algoritmos, tal es el caso del PID (Proportional, Integral, Derivative). Este último permite la corrección de los errores propios del traslado del vehículo, a fin de lograr un vuelo autónomo, fluido y estable dentro del ambiente.

### **2.1.4 Mecanismos de Navegación Híbridos de cuadricópteros en el contexto de la robótica situada con el empleo de sensor de velocidad**

La línea de trabajo se concentra en el desarrollo de navegación híbrida basada en la utilización del sistema de visión externa junto con los sensores provistos por el drone (sensor de velocidad), logrando así tener mayor control sobre las variables que afectan al vuelo.

### **2.1.5 Ambiente de robótica situada aplicada a drones**

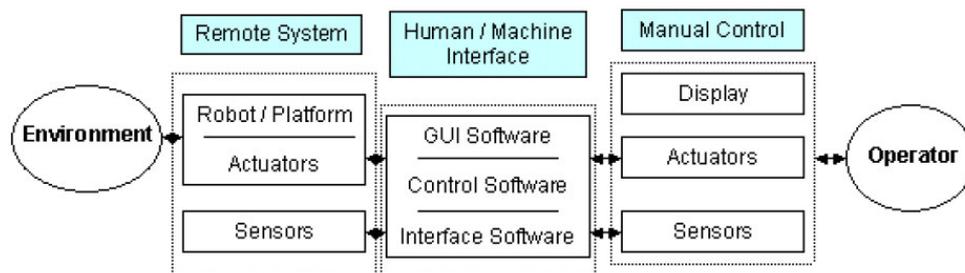
La línea de trabajo se concentra en el desarrollo de una herramienta que identifique el ambiente, la posición del drone y la posición de los checkpoint, permitiendo a un tercero incluir sus mecanismos de navegación. De esta forma se puede generar un ambiente de competencia de robótica situada aplicada a drones.

El AeroQuad es un proyecto de hardware y software de código abierto, dedicado a la construcción de helicópteros de control remoto de cuatro rotores. También es conocido como cuadricóptero (AeroQuad, 2016, web).

## **2.2 MARCO CONCEPTUAL**

### **2.2.1 Sistemas Teleoperados**

Según estudios de Melero, se sostiene que la teleoperación hace referencia a la operación de una máquina a distancia (2014, p.3), pese a que el término teleoperación se aplica con mayor frecuencia, en el ámbito de la investigación, a entornos académicos y técnicos. Por ello, este concepto se asocia con mayor frecuencia a la robótica y a los robots móviles, aunque es posible su aplicación en una amplia gama de circunstancias en las que un dispositivo o máquina es operada por una persona a distancia. La Figura 4 muestra el flujo de información en un sistema teleoperado.



**Figura 4 Flujo de información en un sistema teleoperado**

Fuente: (Melero, 2014)

## 2.2.2 Elementos de un sistema teleoperado

El estudio de Melero de (2014, pp. 6-7) muestra que un sistema teleoperado se compone de los siguientes elementos:

**Operador.-** Ser humano que realiza una serie de acciones en un entorno remoto, puede ser mediante el control de un joystick o el envío de información a través de pulsadores en una estación de teleoperación.

**Teleoperador.-** Máquina que permite al operador moverse, sentir y manipular objetos mecánicamente a distancia. Esta máquina realiza la acción de comunicar al robot con el humano a través de un medio de comunicación físico o inalámbrico.

**Interfaz.-** Conjunto de dispositivos hardware que dan al usuario información del entorno donde se encuentra el robot a manipular, así como la capacidad de realizar acciones en el mismo. Se puede incluir pantallas, teclados, joysticks, entre otros medios.

**Telerobot.-** Es el elemento final de la teleoperación. Éste es el que acepta instrucciones a distancia. Por lo general tiene sensores y actuadores para la manipulación, a más de un medio por el cual se realiza la comunicación.

**Canal de comunicación.-** Es el medio por el cual circula la información que envía el operador y su retroalimentación. Este medio puede ser visible, a través de cableado, o inalámbrico, por una red. Los mensajes se codifican a través de un protocolo de comunicación que varía según el medio y el tipo de mensaje.

**Sensores.-** Éste dispositivo es capaz de detectar magnitudes físicas y transformarlas en variables eléctricas. Las variables de instrumentación pueden ser por su temperatura, intensidad lumínica, distancia, aceleración, inclinación, desplazamiento, presión, fuerza, torsión, humedad, movimiento, pH, entre otros. Estas variables proporcionan la información necesaria del entorno remoto donde se encuentra el robot o son usadas por el usuario para mandar información a este.

### **2.2.3 Aplicaciones de la teleoperación**

Con el desarrollo de la tecnología las aplicaciones de los sistemas teleoperados se han vuelto más necesarios en la ejecución de tareas en entornos remotos. A continuación se presentan los diferentes campos de aplicación.

#### **Aplicaciones en el espacio**

La aplicación de la teleoperación en las sondas espaciales es cada vez más frecuente, ya que se utilizan telerobots que son manipulados remotamente para recolectar información fiable de imágenes y otros datos sensoriales de la superficie en la que se encuentran. La Figura 5 muestra al robot explorador robot Curiosity el cual es un vehículo usado por la NASA para misiones de investigación.



**Figura 5 Vehículo explorador Curiosity**  
Fuente: (Melero, 2014)

Según trabajo de investigación publicado, la aplicación de técnicas de teleoperación en el espacio para la manipulación remota o el envío de información de una superficie terrestre posee las siguientes ventajas (Melero, 2014, p. 11)

- Seguridad. Al tratarse de operaciones de alto riesgo, que en algunos casos podrían provocar la muerte del astronauta, ya que se trabaja en entornos peligrosos.
- Costo. El coste del equipo necesario y los medios necesarios para mandar a un humano son mucho mayores a los de un sistema de teleoperación.
- Tiempo. Las misiones en muchos casos requieren años de investigación en el sitio remoto, para ello es necesario que la tripulación no sea humana. La autonomía para la movilidad del telerobot es obtenida mediante paneles solares.

## Aplicaciones en la industria nuclear

Para precautelar la salud del operario surge la necesidad de utilizar robots controlados remotamente para tareas particulares, como manipular sustancias radiactivas, limpiar residuos tóxicos, operaciones de mantenimiento, descontaminación de instalaciones, entre otros. La Figura 6 muestra el robot teleoperado iRobot PackBot usado para tareas en entornos radiactivos.



**Figura 6 iRobot PackBot.**

Fuente: (Melero, 2014)

## Aplicaciones Submarinas

Para estas aplicaciones se utiliza el telerobot denominado Remote Operated Vehicle, ROV por sus siglas en inglés. Estos son usados para trabajos a grandes profundidades donde el ser humano corre peligro (2014, web). Según el estudio de Melero, “entre [las principales] funciones [del ROV] se pueden destacar las siguientes: reparación y mantenimiento de plataformas petrolíferas en alta mar y estudios geológicos” (2014, p. 12).

## Aplicaciones médicas

Una de las principales aplicaciones de la teleoperación está dentro del sector médico, ya que presenta la posibilidad de control remoto y supervisión de pacientes mediante telepresencia, sin que el médico tenga que desplazarse hasta el lugar donde se encuentra el paciente. Este tipo de aplicación da mayor precisión y control al cirujano dentro de operaciones quirúrgicas. En la Figura 7 se muestra el sistema quirúrgico Da Vinci.



**Figura 7 Sistema Quirúrgico Da Vinci**  
Fuente: (Melero, 2014)

## Aplicaciones militares

Dentro de este campo se utilizan vehículos terrestres denominados *Unmanned Ground Vehicles* o UGVs. Mediante sensores estos vehículos pueden tomar decisiones de manera automática o pasar la información a un operador humano para llevar a cabo tareas de operaciones de vigilancia terrestre, presencia urbana de la calle, entre otras. Otros vehículos que utilizan sistemas teleoperados dentro de este campo y con aplicaciones como vigilancia, adquisición de objetivos militares, detección de enemigos, reconocimiento, entre otros. A este tipo de vehículos se les denomina *Unmanned Air Vehicles* o UAV.

Gracias al avance tecnológico la manipulación de un operador está pasando a un segundo plano ya que este tipo de vehículos se vuelven cada vez más inteligentes y autónomos. (Melero, 2014, p. 13)

#### 2.2.4 Vehículo aéreo no tripulado

Conocido como Unmanned Air Vehicle o UAV, por sus siglas en inglés, este vehículo es "...un tipo de aeronave controlada, ya sea remotamente o localmente, mediante un microprocesador, pero siempre sin presencia humana a bordo" (Mayorga, 2009, p. 1).

Actualmente, los grupos de investigación y empresas han mostrado un gran interés en desarrollar aplicaciones civiles para los UAVS. En la tabla 1 se puede observar algunas aplicaciones civiles de los UAVS.

**Tabla 1**  
**Usos civiles de los UAV**

Industria Civil	Uso
Agricultura	Usados en actividades de campo abierto como fumigación y fertilización
Vigilancia cultivos	Control y supervisión de plantaciones
Control medioambiental e investigación de clima	Análisis a espacio abierto
Exploración mineral	Búsqueda en zonas de riesgo o acceso complicado
Vigilancia guardacostas	Manejo de la seguridad marítima
Telecomunicaciones	Varios medios de telecomunicaciones, desde audiovisuales hasta transferencia de data a distancia

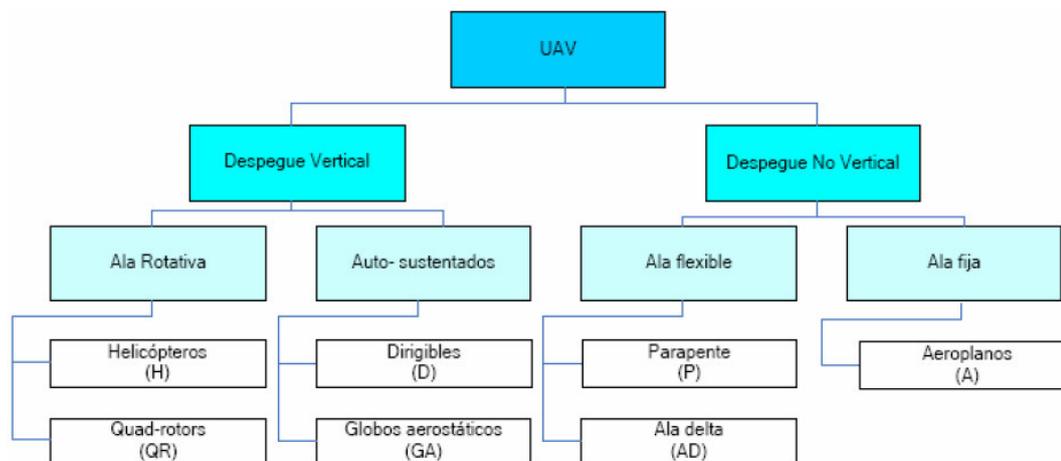
CONTINÚA 

Industria Civil	Uso
Radiodifusión de noticias	Teletransmisión de información de acceso complicado
Detección de artillería sin detonar	Búsqueda de bombas ocultas u objetos militares similares
Control de tráfico aéreo	Obtención de información que mejore los sistemas de toma de decisiones
Control de tráfico terrestre	Revisión constante del tráfico y accidentes

Fuente: (Mayorga, 2009)

### Clasificación de los UAVS

Se los puede clasificar en dos grandes grupos de acuerdo a su despegue. La figura 8 muestra los tipos de aeronaves UAVS



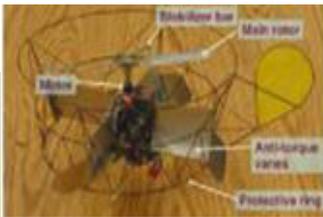
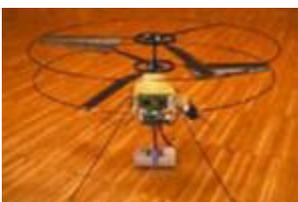
**Figura 8 Tipos de aeronaves UAVS**

Fuente: (Mayorga, 2009)

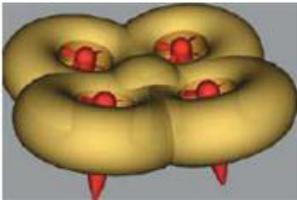
En la tabla 2 se muestran las ventajas y desventajas de acuerdo a su clasificación

Tabla 2

**Ventajas y desventajas de los UAV**

<b>Configuración</b>	<b>Fotografía</b>	<b>Ventajas</b>	<b>Desventajas</b>
Ala fija		<ul style="list-style-type: none"> <li>*Mecanismo de baja complejidad</li> <li>*Actividad no ruidosa</li> </ul>	<ul style="list-style-type: none"> <li>*Es móvil</li> </ul>
Rotor único		<ul style="list-style-type: none"> <li>* Fácil manejo</li> </ul>	<ul style="list-style-type: none"> <li>*Mecanismo de alta complejidad</li> <li>*Rotor de gran volumen</li> <li>*Profundidad en su cola</li> </ul>
Rotor Axial		<ul style="list-style-type: none"> <li>*Condensado</li> <li>*Mecanismo de baja complejidad</li> </ul>	<ul style="list-style-type: none"> <li>*Control complejo</li> <li>*Mala maniobrabilidad</li> </ul>
Rotor Coaxial		<ul style="list-style-type: none"> <li>*Compacto</li> <li>*Mecánica simple</li> </ul>	<ul style="list-style-type: none"> <li>*Control complejo</li> <li>*Mala maniobrabilidad</li> </ul>
Rotor Tandem		<ul style="list-style-type: none"> <li>*Fácil manejo</li> <li>*Sin interferencia en su aerodinámica</li> </ul>	<ul style="list-style-type: none"> <li>*Aerodinámica de alta complejidad</li> </ul>
Cuadricóptero		<ul style="list-style-type: none"> <li>* Fácil manejo</li> <li>*Mecanismo de baja complejidad</li> <li>*Aumento en la carga útil</li> </ul>	<ul style="list-style-type: none"> <li>*Gran tamaño</li> </ul>

CONTINÚA →

Configuración	Fotografía	Ventajas	Desventajas
Dirigible		*Bajo consumo de energía *Auto sustentable	*Gran tamaño
Híbrido		* Fácil manejo * Nivel alto de supervivencia	*Gran tamaño
Tipo pájaro		* Fácil manejo * Bajo consumo de energía	*Gran tamaño *Complejidad en su diseño
Tipo insecto		* Fácil manejo *Compacto	*Mecanismo de alta complejidad * Manejo de alta complejidad
Tipo pez		*Movilidad multimodo *Aerodinámica de alta calidad	*Manejo de alta complejidad *Maniobrabilidad deficiente

Fuente: (Mayorga, 2009)

Luego de este análisis de comparación y contraste, de UAVS se concluye que el mejor vehículo de despegue vertical es el cuadricóptero.

### 2.2.5 ¿Qué es un cuadricóptero?

El cuadricóptero es "...una aeronave de estructura generalmente en forma de cruz equipada con cuatro motores, uno en cada extremo de la cruz, que brindan la tracción necesaria para su vuelo" (Sánchez & Olea, 2014, p. 19).



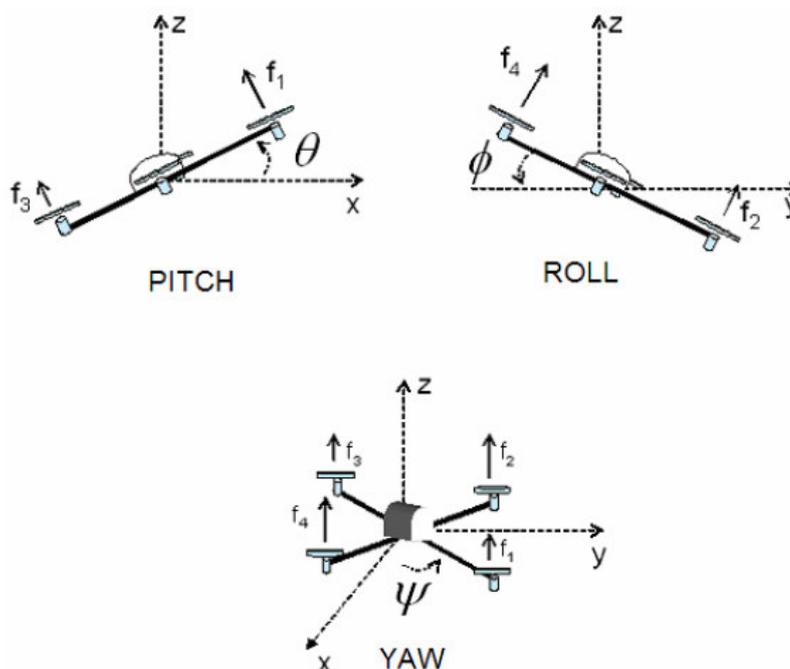
**Figura 9 Cuadricóptero Phamtom 3 Standard**

Fuente: (dji store)

Los cuadricópteros utilizan la energía de una batería, respecto a sus sistemas de control así como de los sensores electrónicos, para mantener su estabilidad. En contraste frente a los helicópteros tradicionales, su movilización depende de la velocidad angular de uno de sus rotores. De acuerdo con el estudio de Sánchez & Olea de (2014):

Los movimientos a través de los ejes se realizan con la manipulación de la potencia enviada a los motores, esto se refleja en la velocidad con que las hélices giran además de la contraposición de giro que dos de ellas realizan para la compensación de los movimientos de inercia. (p. 19)

En la figura 10 se presenta los tres grados de libertad que poseen estos vehículos voladores.



**Figura 10 Descripción de los grados de libertad**

Fuente: (Mayorga, 2009)

En el estudio realizado por Mayorga (2009, p. 8) se da a conocer que para maniobrar libremente al cuadricóptero se necesita la combinación de tres movimientos:

### **Movimiento de guiñada YAW**

De tipo vertical lineal en su propio eje, es un movimiento que se alcanza cuando se incrementa o reduce la potencia de forma equidistante en los rotores 1-3 o, a su vez, en los rotores 2-4. Cuando se incrementa la potencia de uno de estos pares de rotores, que debe ser equilibrada frente al opuesto, produce el efecto contrario en el par paralelo, lo que genera este tipo de movimiento. La fuerza de los rotores debe ser constante para que el cuadricóptero se mantenga estático. Su ángulo es de tipo  $\psi$ .

### **Movimiento de inclinación PITCH**

Este tipo de movimiento permite que el vehículo se desplace enfrente y hacia tras. Su proceso se activa cuando el primer rotor es acelerado en sentido contrario de la dirección destino. Para ello, se debe mantener a los rotores pares (2 y 4) en potencia media y al paralelo (3) en potencia baja. Su ángulo es de giro tipo  $\theta$ .

### **Movimiento de bamboleo ROLL**

Se conoce como movimiento roll cuando el proceso de rotores induce a un movimiento de izquierda a derecha o viceversa del motor. El proceso es exactamente igual que el del movimiento tipo PITCH de forma opuesta. Su ángulo es de giro  $\phi$ .

### **Ventajas y Desventajas del Sistema Cuadricóptero**

Sánchez & Olea (2014, p. 19) encontraron que el cuadricóptero tiene una ventaja frente a otros UAVS, que radica gracias a la configuración de la estructura de los motores que, en conjunto con los sensores de estabilidad, le brindan al vehículo la capacidad de levantamiento o sustentación en el aire. Otra ventaja es que con el pasar del tiempo, el avance tecnológico ha permitido aumentar la relación carga-peso, ya que mientras los componentes funcionales del vehículo se reducen en tamaño y peso, la potencia y poder han aumentado la posibilidad de cargar mayores volúmenes y pesos.

Una de las principales desventajas es el bajo nivel de autonomía, ya que utiliza motores eléctricos, más no motores de combustión, por lo que deben estar constantemente alimentados por baterías. Dependiendo de la calidad y carga de la batería, la autonomía máxima puede llegar a ser de 20 a 40 minutos.

Otra desventaja es que mientras más pequeñas sean sus dimensiones su sistema de control y estabilidad no será tan preciso el cuadricóptero, ya que su desplazamiento depende no sólo del efecto aerodinámico sino de las dinámicas de los rotores.

## 2.3 CARÁCTERÍSTICAS TÉCNICAS DEL CUADRICÓPTERO

### 2.3.1 Phantom 3 Standard

Es un cuadricóptero que, gracias a sus potentes motores, se puede controlarlo de manera fácil y segura. Es utilizado para grabar videos y capturar fotografías de gran calidad gracias a la tecnología de su cámara. Posee un controlador remoto personalizado, una batería de vuelo inteligente y una aplicación para dispositivos móviles denominada DJI GO.

### 2.3.2 Principales Características

**Cámara y gimbal:** De acuerdo con el Manual del usuario del PHANTOM 3 Standard (2015, p. 6) “la cámara integrada captura vídeo a 2.7K y fotos de 12 megapíxeles. El gimbal de 3 ejes mantiene la cámara estable y firme en el vuelo, lo que se traduce en filmaciones sin sacudidas ni temblores”.

**Batería de vuelo inteligente:** Está incorporada por un sistema avanzado de gestión de energía que ofrece un tiempo de vuelo aproximado de 25 minutos (DJI, 2015, p. 6).

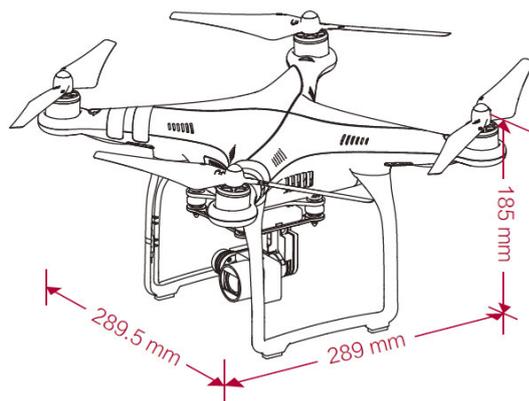
**Transmisión de vídeo WiFi:** Durante el vuelo, la cámara transmite el video HD en tiempo real a la aplicación DJI GO a través de WiFi. El controlador remoto está integrado con un amplificador de WiFi (DJI, 2015, p. 6).

**Controlador de vuelo:** Proporciona estabilidad, seguridad y control al cuadricóptero. Para optimizar el manejo de la aeronave se han

diseñado distintos modos de vuelo de acuerdo a las condiciones y fines del usuario. Los datos cruciales para el vuelo se recopilan, procesan y comunican a la aeronave en tiempo real (DJI, 2015, p. 6).

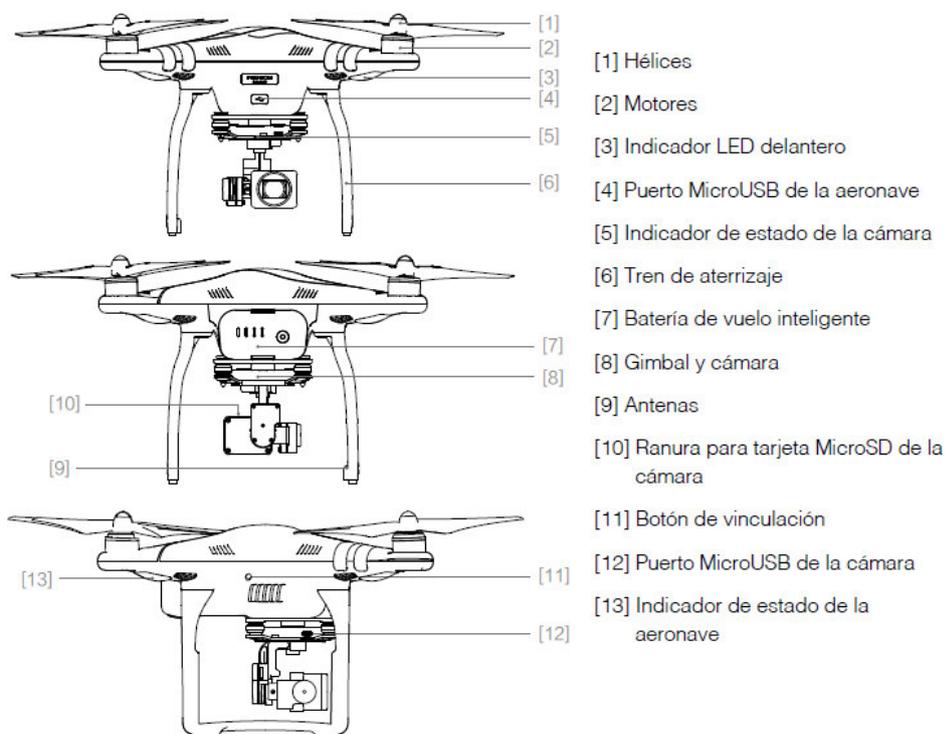
### 2.3.3 Diagrama de la aeronave

#### Dimensiones.



**Figura 11 Dimensiones del cuadricóptero Phantom 3 Standard**  
Fuente: (PHANTOM 3 Guía de inicio rápido, 2015)

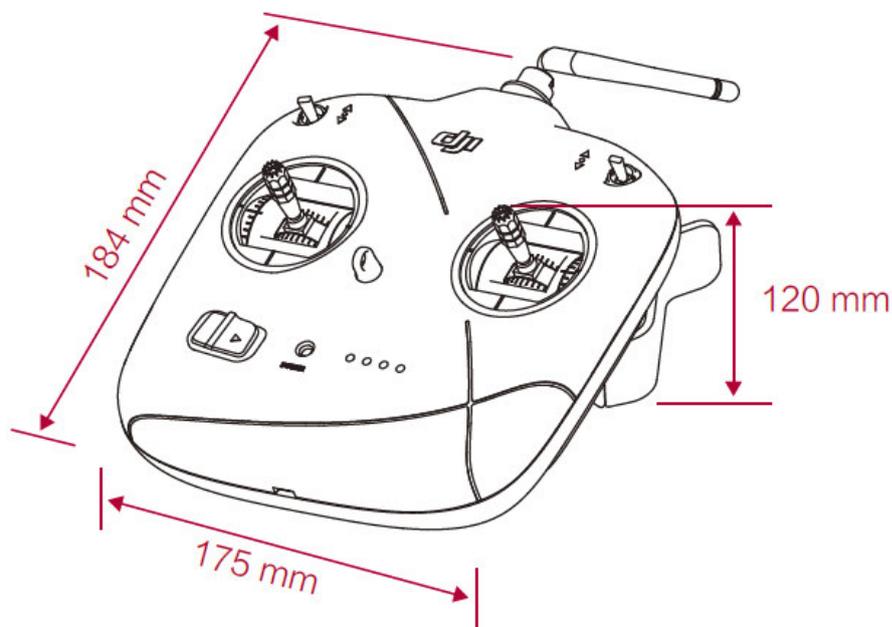
#### Elementos.



**Figura 12 Elementos del cuadricóptero Phantom 3 Standard**  
Fuente: (PHANTOM 3 Standard Manual del usuario, 2015)

### 2.3.4 Diagrama del controlador remoto

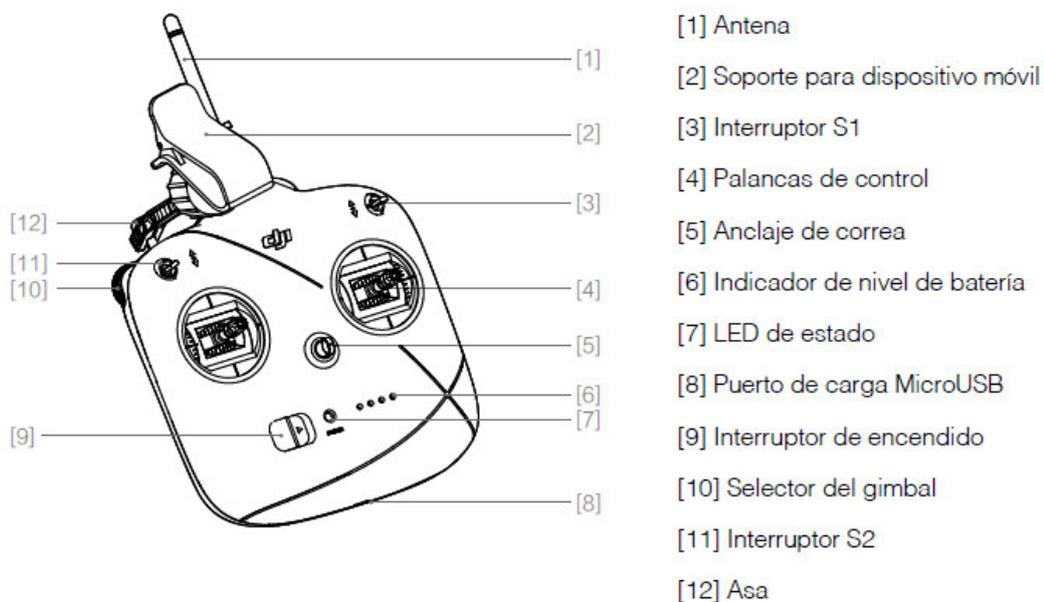
#### Dimensiones.



**Figura 13 Dimensiones del controlador remoto**

Fuente: (PHANTOM 3 Guía de inicio rápido, 2015)

#### Elementos.



**Figura 14 Elementos del controlador remoto**

Fuente: (PHANTOM 3 Standard Manual del usuario, 2015)

### 2.3.5 Especificaciones Técnicas

**Tabla 3**  
**Especificaciones Técnicas del cuadricóptero**

<b>Especificaciones Técnicas</b>	
<b>Aeronave</b>	
Peso (batería incluida)	1216 g
Velocidad de ascenso máx.	5 m/s
Velocidad de descenso máx.	3 m/s
Velocidad máx.	16 m/s (modo ATTI, sin viento)
Altitud de vuelo máx.	6000 m
Tiempo de vuelo máx.	Aproximadamente 25 minutos
Intervalo de temperaturas de funcionamiento	0 °C - 40 °C
GPS	Sistema de navegación por satélite
<b>Gimbal</b>	
Rango controlable	Inclinación: - 90° a +30°
Intervalo de vibración angular	±0,02°
<b>Cámara</b>	
Sensor	1/2,3" Píxeles efectivos:12 M
Lente	FOV (campo de visión) 94° 20 mm (equivalente a formato de 35 mm) f/2,8
Intervalo de ISO	100-3200 (vídeo) 100-1600 (fotografía)
Velocidad de obturador electrónico	8 s – 1/8000 s
Tamaño máx. imagen	4000 x 3000
Modos de fotografía fija	Disparo único Disparo en ráfagas: 3/5/7 fotogramas Horquilla de exposición automática (AEB): 3/5 fotogramas horquillados con sesgo de 0,7 EV Disparo a intervalos
Modos de grabación de vídeo	UHD: 2.7Kp30 (2704 x1520) FHD: 1920x1080p 24/25/30/48 HD: 1280x720p 24/25/30/48/50/60
Tasa de bits máx. para almacenamiento de vídeo	40 Mbps
Formatos de archivo admitidos	FAT32/exFAT Fotografía: JPEG, DNG Vídeo: MP4/MOV (MPEG-4 AVC/H.264)

CONTINÚA 

## Especificaciones Técnicas

### Cámara

Tipos de tarjetas SD admitidas	MicroSD, Capacidad máx.: 64 GB Clase 6 o superior
Intervalo de temperaturas de funcionamiento	0 °C - 40 °C

### Enlace de vídeo Wi-Fi

Frecuencia de funcionamiento	2,400 GHz – 2,483 GHz
Distancia de transmisión máx	FCC: 1000 m; CE: 500 m (al aire libre y sin obstáculos, altitud de la aeronave a 120 m [400 pies])
Potencia de transmisión (EIRP)	FCC: 27 dBm; CE: 20 dBm

### Control remoto

Frecuencia de funcionamiento	5,725 GHz - 5,825 GHz, 922,7 MHz - 927,7 MHz (Japón)
Distancia de transmisión máx.	FCC: 1000 m; CE: 500 m (al aire libre y sin obstáculos, altitud de la aeronave a 120 m [400 pies])
Intervalo de temperaturas de funcionamiento	0 °C - 40 °C
Batería	2600 mAh LiPo 18650
Potencia de transmisión (EIRP)	FCC: 19 dBm; CE: 14 dBm
Voltaje de funcionamiento	600 mA a 3,7 V
Puerto de carga	MicroUSB

### Cargador

Voltaje	17,4 V
Potencia nominal	57 W

### Batería de vuelo inteligente (PH3-4480 mAh-15,2 V)

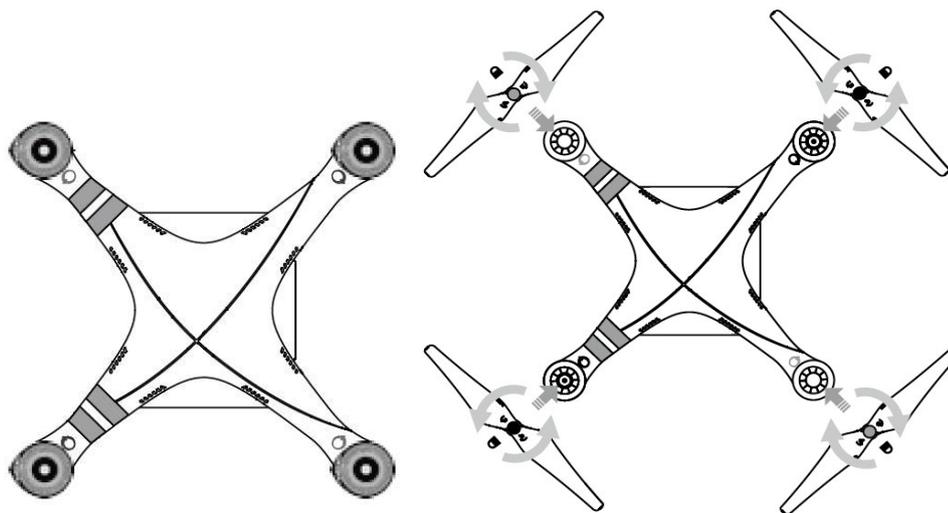
Capacidad	4480 mAh
Voltaje	15,2 V
Tipo de batería	LiPo 4S
Energía	68 Wh
Peso neto	365 g
Temperatura de funcionamiento	-10 °C - 40 °C
Potencia de carga máxima	100 W

Fuente: (dji, 2016)

## 2.3.6 Preparación previa al vuelo

### Fijación de las hélices

El proceso para montar las hélices en el cuadricóptero es muy sencillo. Primero, se monta las tuercas negras de la hélice en los motores con los puntos negros y, para fijarlas, las giramos hacia la izquierda. Seguido, se monta las tuercas plateadas de la hélice en los motores sin los puntos negros y, para fijarlas, las giramos hacia la derecha.



**Figura 15 Fijación de las hélices**

Fuente: (PHANTOM 3 Standard Manual del usuario, 2015)

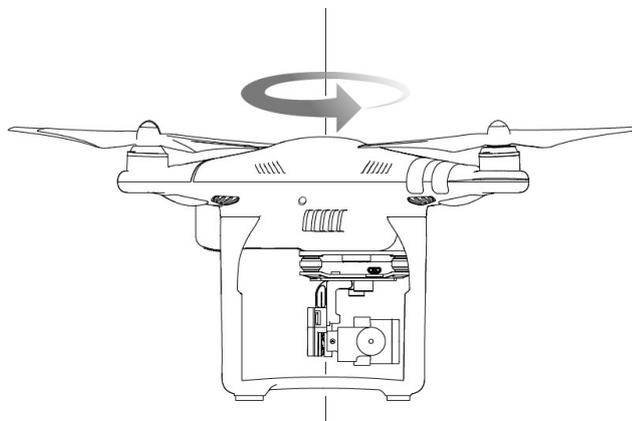
### **Calibración de la brújula**

Para un rendimiento óptimo como preparación previa a cada vuelo es necesario calibrar la brújula de acuerdo a cada nueva ubicación, un problema que puede afectar al vuelo y producir datos anómalos son las interferencias electromagnéticas ya que la brújula es muy sensible.

### **Procedimientos de calibración**

De acuerdo con el Manual del usuario del PHANTOM 3 Standard (2015, p. 41), de preferencia, el proceso de calibración debe desarrollarse en un espacio despejado. Primero, se debe abrir la aplicación DJI GO > Camera View > Aircraft Status Bar (parte superior de la pantalla). Al lado de 'Compass' cliquee Calibrate. El indicador de estado de la aeronave cambiará de amarillo intermitente a amarillo fijo cuando la aeronave esté lista para la calibración de la brújula. Seguido, mantenga la aeronave derecha y gírela

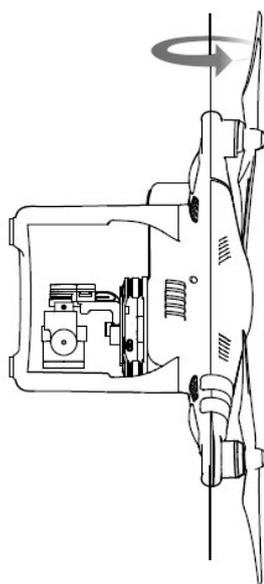
360 grados alrededor del eje central. El indicador de estado de la aeronave cambiará de amarillo fijo a verde fijo cuando se haya completado el proceso.



**Figura 16 Calibración Horizontal**

**Fuente:** (PHANTOM 3 Standard Manual del usuario, 2015)

Luego, mantenga la aeronave con la cámara apuntando hacia abajo y gírela 360 grados a lo largo de su eje central. El foco indicador cambiará de verde fijo a amarillo fijo cuando se haya completado el proceso. La calibración se ha completado con éxito. Cuando este indicador cambia a rojo fijo, la calibración no se ha realizado correctamente. (PHANTOM 3 Standard Manual del usuario, 2015: pp.41, 42).

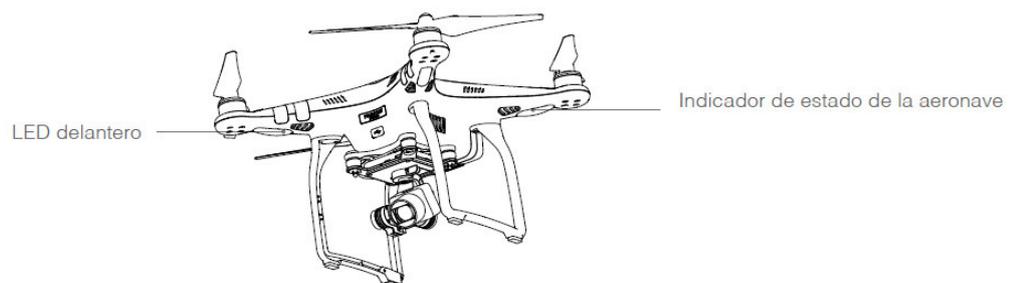


**Figura 17 Calibración Vertical**

**Fuente:** (PHANTOM 3 Standard Manual del usuario, 2015)

### 2.3.7 Indicadores LED

De acuerdo con el Manual del usuario del PHANTOM 3 Standard, el vehículo tiene dos indicadores LED frontales, que están situados bajo los dos motores delanteros y dos indicadores de estado de la aeronave ubicados bajo los dos motores traseros. Los LED frontales se iluminan en rojo fijo cuando se enciende la aeronave y le ayudan a orientar la aeronave cuando está en el aire. Se pueden apagar los LED frontales en la aplicación DJI GO para lograr un mejor resultado en el rodaje de vídeo. Por tanto, se concluye que todos los indicadores mencionados deben ser controlados siempre para evitar cualquier descontrol durante el desplazamiento. (2015, p. 10).



**Figura 18 Indicadores LED**

Fuente: (PHANTOM 3 Standard Manual del usuario, 2015)

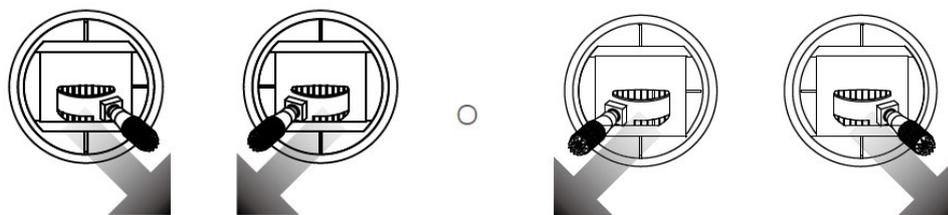
### 2.3.8 Arranque y parada de los motores

El comando de combinación de palancas CSC (Combination Stick Command) se debe ejecutar en un movimiento continuo, según se explica en los siguientes apartados.

#### Arranque de los motores

Para arrancar los motores es necesario empujar las dos palancas hacia las esquinas inferiores interiores o exteriores, según se da ejemplo en la siguiente figura. Una vez que los motores hayan empezado a girar, se debe

soltar las dos palancas a la vez (PHANTOM 3 Standard Manual del usuario, 2015, pág. 43).

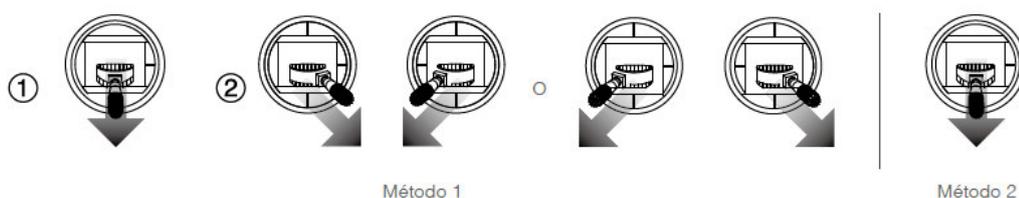


**Figura 19 Arranque de los motores**

Fuente: (PHANTOM 3 Standard Manual del usuario, 2015)

### Parada de motores

De acuerdo con el Manual del usuario del PHANTOM 3 Standard, existen dos métodos para detener los motores. El primero consiste en que, cuando la aeronave haya aterrizado, se mueva la palanca del acelerador hacia abajo y ejecute el comando CSC para detener los motores. Suelte las dos palancas una vez que se detengan los motores. El segundo consiste en que, cuando la aeronave haya aterrizado, se empuje el acelerador hacia abajo y manténgalo. Los motores se pararán después de 3 segundos. (2015, p. 43).



**Figura 20 Métodos para detener los motores**

Fuente: (PHANTOM 3 Standard Manual del usuario, 2015)

### 2.3.9 Transmisión de vídeo WiFi

La transmisión de vídeo WiFi de 2,4 GHz se conecta de manera inalámbrica a la aplicación DJI GO y al controlador remoto. Es importante mencionar que existe un amplificador WiFi, que se utiliza para aumentar el

alcance de transmisión entre la cámara del cuadricóptero y el controlador remoto. El alcance de comunicación efectivo es de 1000m (0,62 millas) con línea de vista.

### **Conexión a la Transmisión de vídeo WiFi:**

Los pasos para conectar al dispositivo móvil con el cuadricóptero son los siguientes:

1. Encender el controlador remoto;
2. Encender el cuadricóptero;
3. En la lista de redes WiFi del dispositivo móvil se debe seleccionar "PHANTOM3\_04abda" e introducir la contraseña predeterminada "12341234"; y
4. Acceder a la página "Camera View" en la aplicación DJI GO. Al visualizar la señal de vídeo de la cámara de la aeronave se habrá establecido la conexión con la transmisión de vídeo WiFi correctamente.

#### **2.3.10 Aplicación DJI GO**

Es un programa de vital importancia al utilizar el cuadricóptero, porque permite compartir imágenes, explorar y gestionar la cuenta de DJI.

**Tabla 4**  
**Especificaciones de la aplicación DJI GO**

<b>APLICACIÓN / RETRANSMISIÓN EN DIRECTO</b>	
Aplicación Móvil	DJI GO
PIRE	100mW
Densidad Espectral de Potencia	2.4GHz ISM

CONTINÚA 

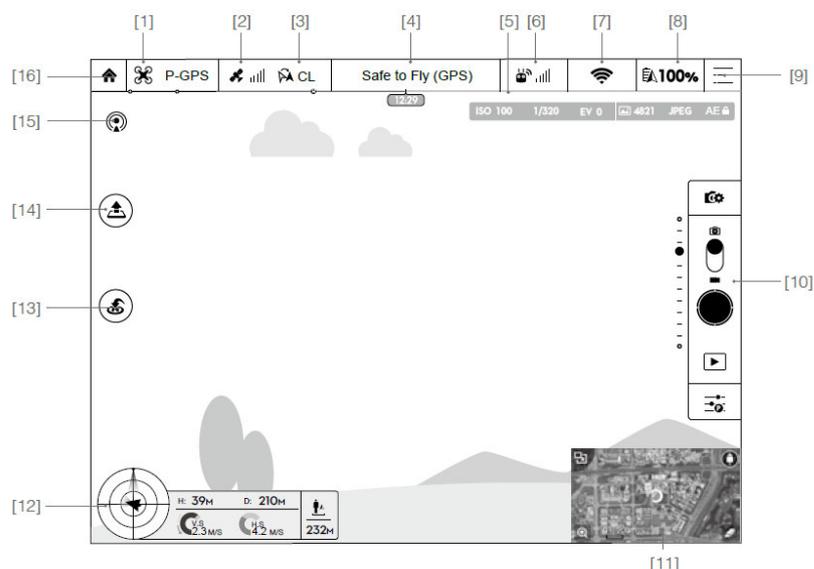
<b>APLICACIÓN / RETRANSMISIÓN EN DIRECTO</b>	
Frecuencia de Funcionamiento de la Retransmisión en Directo	720P @ 30fps (dependiendo de las condiciones y los dispositivos móviles)
Latencia	220ms (dependiendo de las condiciones y los dispositivos móviles)
Sistema Operativo Requerido	iOS 8.0 ó posterior Android 4.1.2 ó posterior
Dispositivos Recomendados	<p>Compatible con iPhone 5s, iPhone 6, iPhone 6 Plus, iPhone 6s, iPhone 6s Plus, iPod touch 6, iPad Pro, iPad Air, iPad Air Wi-Fi + Cellular, iPad mini 2, iPad mini 2 Wi-Fi + Cellular, iPad Air 2, iPad Air 2 Wi-Fi + Cellular, iPad mini 3, iPad mini 3 Wi-Fi + Cellular, iPad mini 4, y iPad mini 4 Wi-Fi + Cellular.</p> <p>Esta aplicación está optimizada para iPhone 5s, iPhone 6, iPhone 6 Plus, iPhone 6s and iPhone 6s Plus.</p> <p>Compatible con Samsung tabs 705c, Samsung S6, Samsung S5, Samsung NOTE4, Samsung NOTE3, Google Nexus 9, Google Nexus 7 II, Ascend Mate7, Huawei P8 Max, Nubia Z7 mini, Sony Xperia Z3, MI 3, MI PAD, Smartisan T1. Huawei Mate 8.</p> <p>*Disponible asistencia para otros dispositivos Android. Testeo y desarrollo en proceso.</p>

**Fuente:** (dji, 2016)

La página matriz se denomina *Equipment*, donde se puede entrar en la Vista de la cámara y ver los registros de vuelos.

### **Vista de la cámara**

El centro de la pantalla presenta el vídeo HD de la cámara en tiempo real, el panel principal está rodeado de iconos y botones que posibilitan el control del cuadricóptero y la cámara. (DJI, 2015, p. 32).

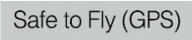


**Figura 21 Vista de la cámara**  
 Fuente: (PHANTOM 3 Standard Manual del usuario, 2015)

**Tabla 5**  
**Botones e íconos de la vista de la cámara**

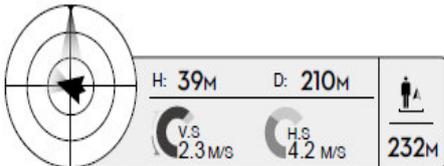
Nº	Botón	Icono	Especificación
[1]	Modo de vuelo		Ingresa en la configuración principal del controlador, donde puede configurar los parámetros que afectan al comportamiento de vuelo.
[2]	Intensidad de señal GPS		Muestra la intensidad de la señal GPS actual. Las barras verdes indican una intensidad GPS adecuada.
[3]	Configuración del IOC (Control de Orientación Inteligente)		Muestra los ajustes de IOC cuando la aeronave se encuentra en modo F. Al clicar este botón se apertura el menú IOC, permitiendo seleccionar el modo IOC deseado.

CONTINÚA →

Nº	Botón	Icono	Especificación
[4]	Barra de estado del sistema		Muestra el estado actual de todo el sistema, así como la intensidad de la señal GPS.
[5]	Indicador de nivel de batería		Describe el nivel de la batería de la aeronave de acuerdo con su estado inmediato. Las zonas en color representan los diferentes umbrales de carga. Cuando el nivel de la batería caiga por debajo de un umbral determinado, el sistema pedirá al usuario que tome las medidas adecuadas.
[6]	Señal del controlador remoto		Indica la intensidad de la señal del controlador remoto.
[7]	Señal de transmisión de vídeo		Indica la intensidad de la señal de la transmisión de vídeo WiFi entre la aeronave y el controlador remoto. Se toca para establecer el SSID y la contraseña de conexión de la conexión WiFi.
[8]	Nivel de batería		Muestra el nivel actual de carga de la batería. Se toca para ver el menú de la batería de la aeronave, donde puede ver información sobre la batería y definir los umbrales de alerta de la batería.

Nº	Botón	Icono	Especificación
[9]	Configuración general		Sirve para ver la configuración de la cámara, un minimapa y la memoria caché de vídeo
[10]	Barra de la cámara		Operaciones de la cámara Para configurar distintos valores de la cámara, que incluyen el espacio de color para grabación, la resolución de los vídeos, el tamaño de las imágenes, etc.
			<i>Obturador (Foto)</i> Para realizar una única foto. Manténgalo pulsado para cambiar entre los modos de disparo.
			<i>Obturador (Vídeo)</i> Toque este botón para iniciar o detener la grabación de vídeo.
			<i>Reproducción</i> Toque este botón para reproducir fotos y vídeos de la tarjeta MicroSD.
	[11] Mapa		Configuración de la cámara Tóquelo para establecer los valores de ISO, velocidad del obturador y exposición automática.
			Muestra la posición actual de la aeronave. Toque en el mapa para cambiar de la vista de cámara a la vista del mapa.

CONTINÚA 

Nº	Botón	Icono	Especificación
[12]	Telemetría de vuelo		El objeto tipo radar indica la actitud de la aeronave. Toque para abrir el mapa. (1) La flecha roja indica el rumbo de la aeronave. (2) La relación entre las zonas azul y gris indica la inclinación de la aeronave. (3) El nivel horizontal del límite azul-gris indica el ángulo de alabeo de la aeronave.
[13]	Regreso al punto de origen (RTH)		Inicia el despegue o el aterrizaje automático. Inicia el procedimiento RTH.
[14]	Despegue/aterrizaje automáticos		Tóquelo para que la aeronave regrese al último punto de origen registrado.
[15]	Transmisión en directo		Indica que una transmisión de vídeo se está retransmitiendo en directo por YouTube. Asegúrese de que el dispositivo móvil esté conectado a Internet.
[16]	Atrás		Regresa al menú principal.

Fuente: (PHANTOM 3 Standard Manual del usuario, 2015)

### 2.3.11 Condiciones de vuelo seguro

De acuerdo con el Manual del usuario del PHANTOM 3 Standard se debe cumplir con las siguientes normas para un vuelo seguro.

- No utilice la aeronave en condiciones climáticas adversas como lluvia, nieve, niebla ni con una velocidad del viento superior a 10 m/s.

- Vuele solo en espacios abiertos. Los edificios altos y las grandes estructuras metálicas pueden afectar a la precisión de la brújula de a bordo y a la señal GPS.
- Evite volar cerca de obstáculos, multitudes, líneas de alta tensión, árboles y masas de agua.
- Evite volar en zonas con altos niveles de electromagnetismo, incluidas estaciones base de telefonía móvil y torres de transmisión de radio.
- El rendimiento de la aeronave y de la batería depende de factores medioambientales, como la densidad del aire y la temperatura. Tenga mucho cuidado cuando vuele por encima de 6000 m (19 685 pies) sobre el nivel del mar, ya que las prestaciones de la aeronave y la batería pueden verse reducidas (2015, p. 37).

Para resolver cualquier pregunta sobre el funcionamiento y recomendaciones del PHANTOM 3 se recomienda revisar el manual general de funcionamiento de la aeronave.

## 2.4 KINECT

La empresa israelí PrimeSense ha empleado técnicas basadas en proyección infrarroja para desarrollar cámaras de profundidad. El hardware de Kinect fue desarrollado en conjunto por PrimeSense, mediante su cámara de profundidad como software, y Microsoft, con la aplicación de sus algoritmos. Seguido, PrimeSense trabajaría con ASUS en una cámara que sería de uso para televisiones, computadores, entre otros (Ramos, 2013, pág. 21).

De acuerdo con Samaniego, durante el evento *Electronic Entertainment Expo* de 2009, Microsoft presentó el Kinect con el nombre de "*Project Natal*", luego de veinte años de estudio en esta tecnología. El sensor Kinect, por tanto, es un accesorio de la consola XBOX 360, que interactúa a distancia con el XBOX 360 (Martínez Novo, 2015, p. 19). Su interfaz permite reconocer movimientos y gestos, sonidos y voz, y demás, evitando el empleo de un mando manual a distancia (2012, p. 8).

### 2.4.1 Composición

De acuerdo al estudio realizado por Mathe, Samban y Gómez, el sensor Kinect es un equipo que permite el reconocimiento de movimientos y gestos a distancia de colocación horizontal (2012, p.1). El Kinect presenta una "...cámara RGB, un sensor de profundidad y un micrófono multi-array bidireccional que, en conjunto, capturan imágenes y movimientos de los cuerpos en 3D, además de ofrecer reconocimiento facial y aceptar comandos de voz" (Ibídem, p.1).



**Figura 22 Elementos del sensor Kinect**

Fuente: (Toribios Blázquez, 2012)

En el estudio realizado por Martínez Novo se menciona que:

- El sensor de profundidad está adaptado a la resolución de la cámara RGB, por lo que las imágenes de profundidad que ofrece son también una matriz de 640x480.
- Ofrece un ángulo de visión horizontal de 57º y un ángulo vertical de 43º.
- Su rango de detección de profundidad oscila entre los 0.45 y los 4.5 metros.
- Su frecuencia de trabajo es configurable, pero por defecto son 30Hz tanto para las imágenes RGB como para las de detección de profundidad (Martínez Novo, 2015, p. 19).

Otra característica importante que tiene este dispositivo es su emisor de infrarrojos y su arreglo de micrófonos, según el siguiente:

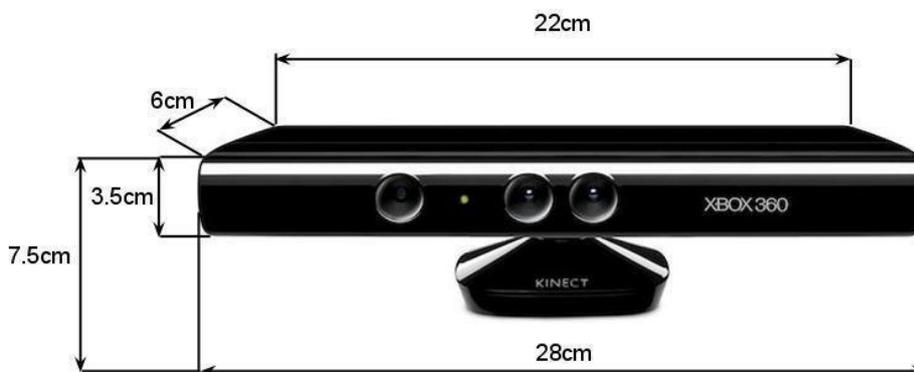
- Emisor IR y sensor de profundidad IR: el emisor IR emite rayos infrarrojos y el sensor de profundidad IR lee los rayos

que se reflejan hacia el sensor. Los rayos reflejados son transformados en información de la distancia entre el sensor y el objeto. Esto hace posible la adquisición de la imagen de profundidad.

- Arreglo de micrófonos: contiene cuatro micrófonos que capturan la información de audio. Gracias a la disposición de los micrófonos es posible grabar y determinar la dirección de la fuente de audio (Cuenca, Delgado, & Chuya, 2013, p. 13).

## 2.4.2 Dimensiones

Las dimensiones del dispositivo Kinect se detallan en la Figura 23



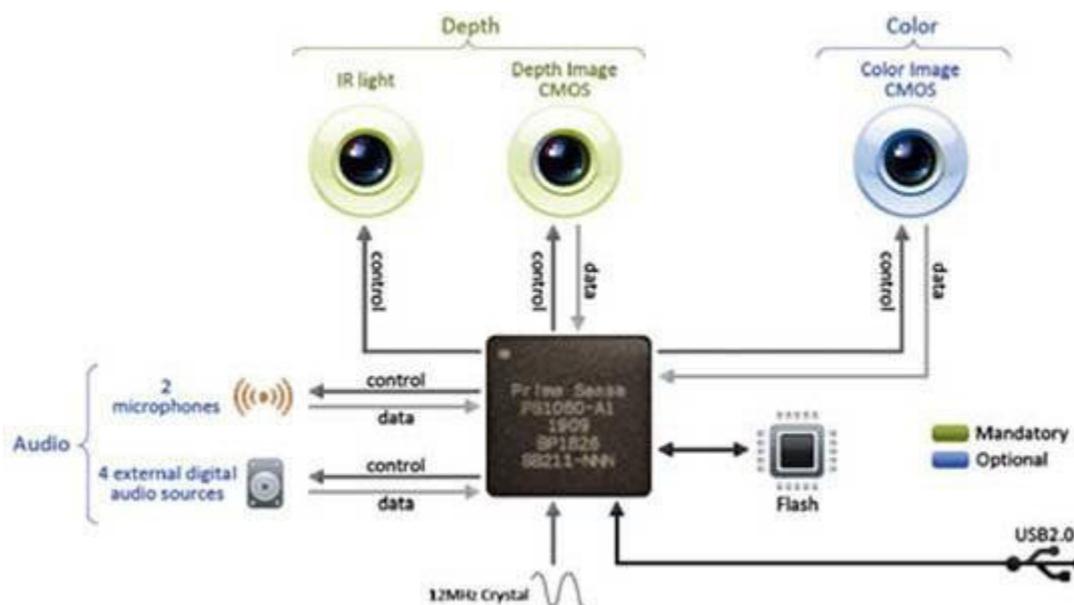
**Figura 23 Dimensiones del sensor Kinect**

Fuente: (Mathe, Samban, & Gómez, 2012)

## 2.4.3 Funcionamiento

### Cámara

El trabajo realizado por Toribios Blázquez se muestra que la cámara RGB detecta los tres colores primarios (Rojo, Verde, Azul) y tras un código indica el porcentaje de cada color. De esta forma se determina el color del píxel. La cámara es capaz de capturar 30 imágenes por segundo y su calidad es de 640x480 píxel en 32-bit (2012, p. 11).



**Figura 24 Esquema Cámara Kinect 1.0**

Fuente: (Martínez Novo, 2015)

Las funciones principales que tiene la cámara son:

- Recrear un mapa en 3D de la imagen que ha capturado;
- Identificar movimientos y gestos de los usuarios;
- Reconocer objetos estáticos y sus distancia entre sí y el usuario (Mathe, Samban, & Gómez, 2012, p. 2).

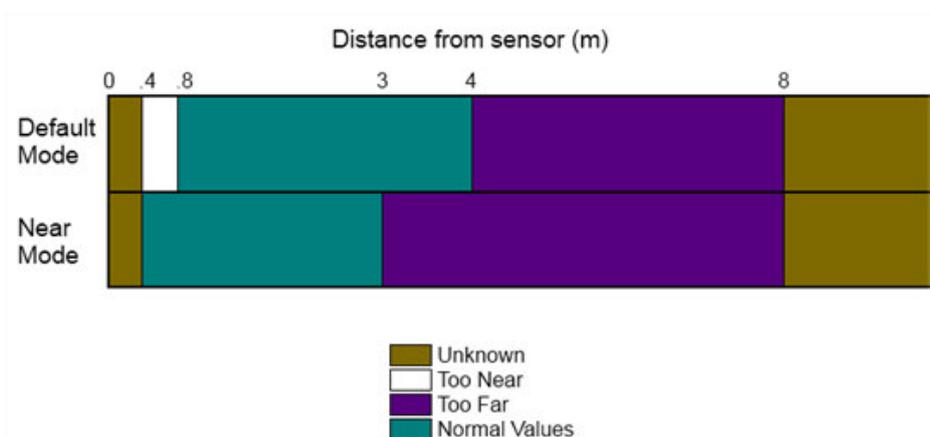
“El sensor Kinect puede llegar a distinguir la profundidad de cada objeto con una resolución de 1 centímetro y las estimaciones de la altura y anchura con una exactitud de aproximadamente 3 milímetros” (Ibídem: p.2).

### Emisor infrarrojo

El Kinect proyecta un conjunto de puntos de luz infrarroja dentro del entorno en que el dispositivo se encuentra. Este proceso se efectúa mediante la cámara radiación infrarroja. Estos puntos son invisibles para el usuario, más son reconocidos por el sensor de dicha cámara (Ramos, 2013, p. 22).

Las imágenes de video que adquiere el dispositivo Kinect son obtenidas a una frecuencia de 30Hz, mediante un sensor CMOS. El canal que emplea es un monocromo 16bits. Su resolución es de tipo VGA 640x480 píxeles, QVGA de 320x240 píxeles. Su sensibilidad máxima es de 65,536 (Mathe, Samban, & Gómez, 2012, p. 1).

Las distancias del entorno frente al dispositivo se miden por la red de puntos infrarrojos proyectados respecto al espacio en rededor. La cámara recibe la colocación en el espacio de los puntos infrarrojos, lo que permite identificar la profundidad de cada uno de estos puntos. La profundidad que reconoce puede efectuarse entre los 40 centímetros hasta un máximo de 4 metros. Actualmente, se identifican el modo Default y Near para el establecimiento de distancias. (Ibídem: p. 2).



**Figura 25 Modos de distancia admitidos por Kinect**

Fuente: (Mathe, Samban, & Gómez, 2012)

Las características principales son:

- Un ángulo de vista 58 grados de forma horizontal y 45° grados de forma vertical. El pivote establece la elevación, sea ésta para arriba o su contrario. Esto incrementa el ángulo hasta un máximo de 27 grados.
- El micrófono presenta un array de 4 cápsulas. Éste gestiona cada canal, con un procesamiento de audio 16bits y frecuencia de hasta 16 kHz de radio. (Mathe, Samban, & Gómez, 2012, p. 2).

Frente a los tipos de cámara tradicional, el Kinect puede detectar la luz infrarroja a través de su emisor de infrarrojos. Se aclara que la luz infrarroja no puede ser vista a simple vista debido a que su onda es de mayor longitud que la luz que es observable.

#### 2.4.4 Especificaciones Técnicas

**Tabla 6**  
**Características del dispositivo Kinect**

Kinect XBOX-360	
1	Sensores de profundidad Cámara: 640*840 CMOS, 16 bits, 30 fps. Infrarrojo: resolución 0,01 m eje Z; 0,003m ejes X e Y.
2	Cámara RGB VGA 640 * 480, 32bits, 30 fps.
3	Micrófonos Rango: $\pm 50^\circ$ Cancelación de ruido: 20 dB. Resolución: 24 bits PCM (Pulse Code Modulation). Frecuencia: 16 kHz
4	Motor Ángulo de inclinación: $\pm 27^\circ$
5	Acelerómetro 2G/4G/8G para un rango de 2G con una precisión de $\pm 1$ .(G Fuerza gravitacional)
6	Rango de profundidad Min: 0,8m a 1,2m. Max: 3,5m a 4m.
7	Ángulo de visión Horizontal: $57^\circ$ Vertical: $43^\circ$

**Fuente:** (Cuenca, Delgado, & Chuya, 2013)

#### 2.4.5 Controladores y Plataformas

Para el desarrollo de aplicaciones en el dispositivo Kinect se tiene disponible dos entornos de desarrollo de código abierto y un SDK (software developmnet kit). Éste último es exclusivo de Microsoft. En el otro caso, toda vez se comercializó el Kinect se generó la necesidad de desarrollar controladores de código abierto, conocidos en inglés como *open source*. De esta forma, se promueve su uso y aplicación en varios sistemas y plataformas.

### **a) OpenKinect:**

Estas librerías se encuentran bajo licencia de uso libre y están soportadas por la comunidad de software libre. Los controladores desarrollados permiten acceder a las cámaras libremente, obteniendo imágenes RGB y de profundidad. También se puede acceder al control del motor, acelerómetro y el led (Cuenca, Delgado, & Chuya, 2013, p. 15).

### **b) OpenNI/NITE**

La OpenNI, al ser una institución sin fines comerciales, ha desarrollado una tecnología genérica que permite reproducir el ejercicio de interacción natural, conocida por sus siglas en inglés como NI o Natural Interaction. Esta aplicación genérica funciona mediante un código abierto, al que se vincula el Kinect. Este proceso está basado en el Application Programming Interface o sus siglas APIs (Samaniego, 2012, p. 9).

Desarrollado por PrimeSense, este entorno permite acceder a los datos de profundidad, a las imágenes RGB de la cámara, así como al audio del arreglo de micrófonos. NITE es el entorno de trabajo que permite acceder directamente a la esqueletización del usuario para el reconocimiento de posturas y gestos. Éste presenta una licencia comercial (Cuenca, Delgado, & Chuya, 2013, p. 15).

### **c) Microsoft Kinect SDK**

Este entorno de desarrollo es de uso bajo la licencia de Microsoft. Permite acceder al motor, al arreglo de micrófonos para grabar audio, a la cámara RGB y a los datos de profundidad de las imágenes. Además, éste da seguimiento hasta dos usuarios y reconoce a seis en un área aproximada de 10m<sup>2</sup>. A diferencia de las anteriores, el desarrollo de aplicaciones en este

entorno está sujeto a las licencias de uso libre de Microsoft y sus políticas comerciales (Ibídem: p.16).

En la tabla 7, se muestra la comparación de los dos controladores que ofrecen más prestaciones como son el OpenNI y Microsoft SDK

**Tabla 7**  
**Características de los controladores OpenNI y Microsoft SDK**

	<b>SDK MICROSOFT</b>	<b>SDK OPENNI/NITE</b>
<b>Ventajas</b>	Soporte para audio.	Uso libre y comercial.
	Incluye las manos, pies y clavícula.	Adquiere datos de las manos para seguimiento y reconocimiento de gestos.
	Seguimiento de cuerpo completo.	Seguimiento de cuerpo entero.
	No necesita postura de calibración.	Puede calibrar la profundidad y el color de la imagen.
	Soporte para el motor de inclinación.	Cálculo de la rotación de las articulaciones
	Mejor tratamiento de articulaciones no visibles.	Plataforma múltiple: Windows XP, Vista, 7, Linux y MacOSX.
	Soporta múltiples sensores.	Soporte incorporado para grabación y reproducción.
	Instalación simple.	
	Gran cantidad de información disponible.	
	Sistema de reconocimiento de gestos.	
	La versión 1.7 permite correr el Kinect SDK sobre máquinas virtuales.	

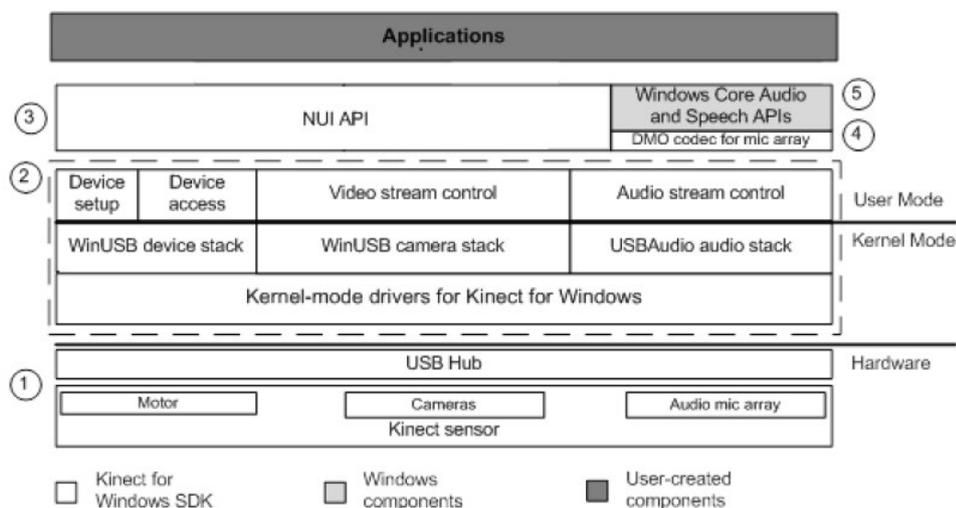
CONTINÚA →

	<b>SDK MICROSOFT</b>	<b>SDK OPENNI/NITE</b>
Desventajas	Licencia única para uso no comercial.	Instalación complicada.
	Mayor consumo de recursos computacionales.	Sin soporte para audio y motor de inclinación.
	Solamente para Windows 7 & 8 (x86 y x64).	Necesita de una postura de calibración.
	Sin soporte incorporado para grabación y reproducción.	Articulaciones no visibles no son estimadas.
	Sin soporte para transmitir los datos sin procesar del sensor infrarrojo.	Compatible con varios sensores, aunque la configuración y enumeración es complicada.

**Fuente:** (Cuenca, Delgado, & Chuya, 2013)

### Arquitectura del SDK de Microsoft

La figura 26 presenta la arquitectura completa del SDK de Microsoft.



**Figura 26 Arquitectura del SDK de Microsoft**

**Fuente:** (Rodríguez, 2013)

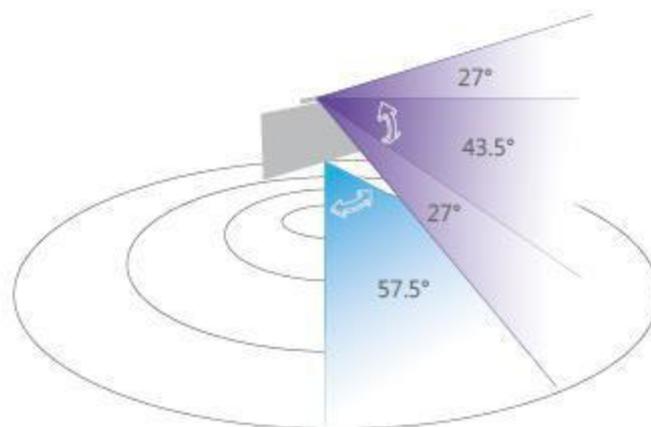
En el estudio realizado por Rodríguez en (2013, p. 14) se detallan los campos de la arquitectura mostrada en la Figura 26:

- **Hardware de Kinect.** Incluye todos los elementos que componen el dispositivo y para su conexión con un computador personal.
- **Drivers de Kinect.** El conjunto de controladores desarrollados por Microsoft, que permiten el uso del dispositivo.
- **NUI API.** Es una base de datos que permite reconocer, captar y procesar datos audiovisuales del entorno.
- **DirectX Media Object (DMO).** Es la central de recepción de audio.
- **Windows 7 API estándar.** Es el conjunto audiovisual API de Microsoft.

## 2.4.6 Campo de Visión

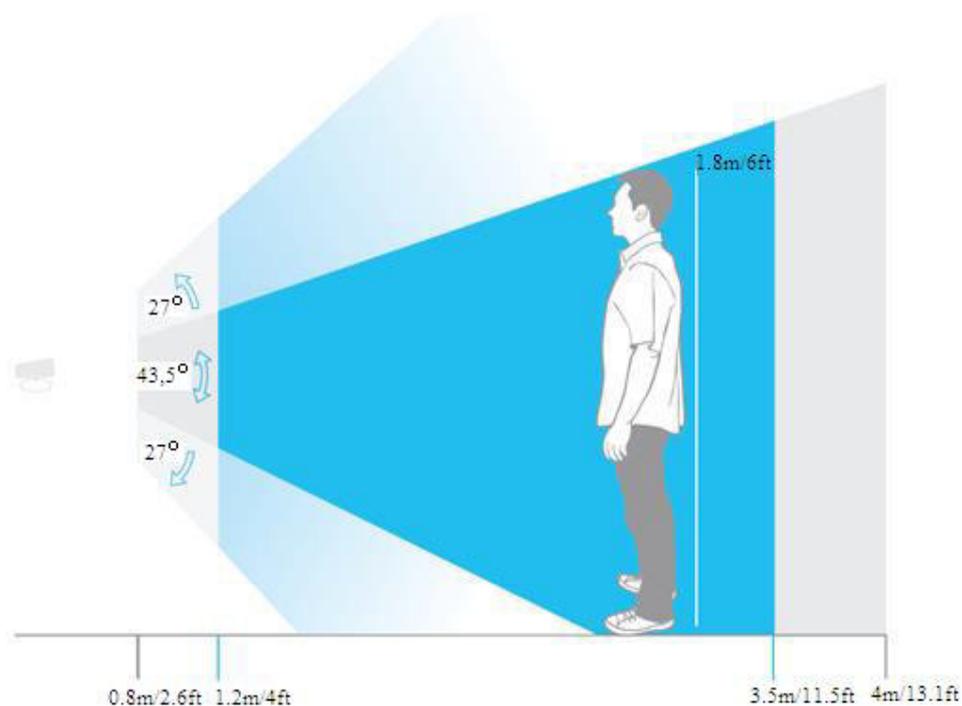
### 1. Visión Vertical y Horizontal

El dispositivo Kinect tiene un rango de visualización entre 0,8 metros (2,6 pies) y 4,0 metros (13,1 pies) de distancia cuando una persona se encuentra parada frente a él.



**Figura 27 Campo de Visión Horizontal y Vertical**

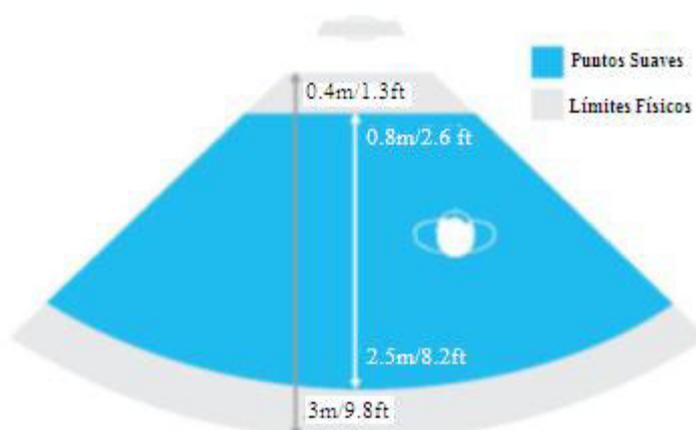
Fuente: (Rodríguez, 2013)



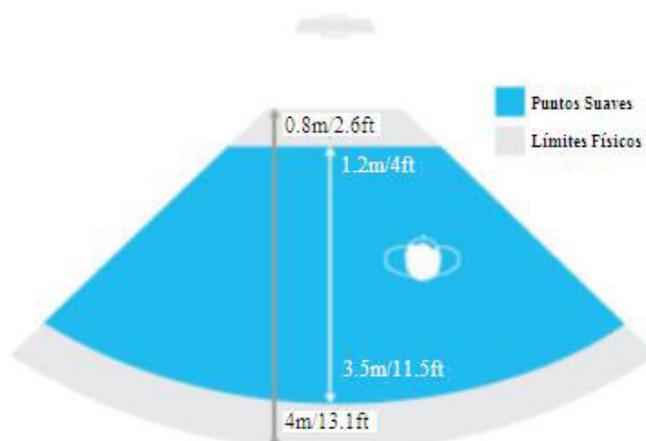
**Figura 28 Campo de visión vertical del sensor Kinect**  
Fuente: (Melero, 2014)

## 2. Visión de profundidad

En la Figura 29 y la figura 30 se puede apreciar los rangos de visión de profundidad para los dos modos que maneja Kinect, considerando las limitaciones físicas que son los rangos reales del área de cobertura (0.4m a 3m) y los puntos suaves que son las zonas óptimas de captura (0.8m a 2.5m).



**Figura 29 Rango de Profundidad Modo Near**  
Fuente: (Melero, 2014)



**Figura 30 Rango de Profundidad Modo Default**

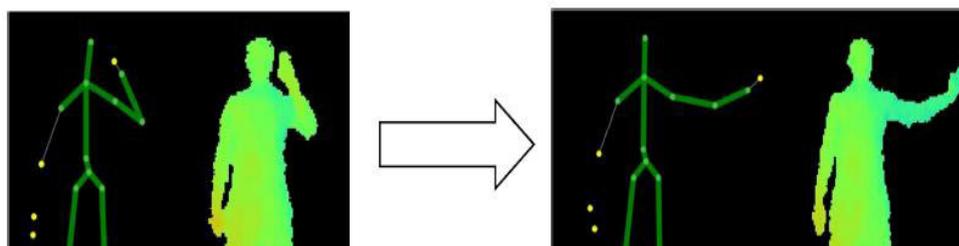
Fuente: (Melero, 2014)

### 2.4.7 Sistema de reconocimiento de gestos

A continuación se detallan las tres formas que tiene el dispositivo Kinect para reconocer los gestos que el usuario realiza:

#### Reconocimiento Basado en Coordenadas

Es el sistema más básico que por su nivel de efectividad es utilizado para reconocer gestos sencillos. El sistema analiza los gestos a ser usados y los codifica mediante comparaciones entre las coordenadas del cuerpo para verificar que se cumplan las condiciones necesarias para cada gesto. El sistema resulta ineficaz si la complejidad del gesto aumenta. La Figura 31 muestra el desplazamiento que realiza la mano derecha cuando parte desde el hombro hasta extender el brazo. El sistema compara las coordenadas X e Y que se realizan al inicio y al final del movimiento.



**Figura 31 Desplazamiento de la mano derecha**

Fuente: (Robledo, 2012)

### **Reconocimiento basado en modelos ocultos de Markov.**

Este sistema es muy eficiente para reconocer gestos y posturas. Se basa en examinar un modelo estadístico de acuerdo a la información de entrada. La información es una sucesión de coordenadas que siguen un patrón específico para cada gesto. Para que la codificación del modelo sea efectiva se necesita conocimientos estadísticos para elegir el conjunto de datos de entrada, ya que un conjunto de datos mal elegido podría hacer que los parámetros extraídos por el modelo no fueran los reales (Robledo, 2012, p. 11).

### **Reconocimiento de gestos DTW (Dynamic Time Warping)**

El sistema tiene la ventaja de obtener un porcentaje de acierto muy alto, aun variando la velocidad de realización del gesto. El sistema utiliza técnicas para reconocer patrones lineales en el tiempo que se produzcan a velocidades diferentes de la velocidad en la que fueron capturados. El principal problema que tiene este modelo es que, aunque abstrae muy bien la velocidad de ejecución del gesto, tiene más problemas con las desviaciones sobre el mismo, y, dado que lo interesante es conseguir un reconocimiento de patrones de alta universalidad, esta alternativa pierde algo de interés en su aplicación (Robledo, 2012, p. 14).

## **CAPÍTULO III**

### **IMPLEMENTACIÓN DE LA INTERFAZ DE CONTROL**

El presente capítulo está enfocado en mostrar todo el proceso que se realizó desde el diseño inicial de la interfaz de control hasta la implementación de la misma.

#### **3.1 ANÁLISIS PARA LA IMPLEMENTACIÓN DE LA INTERFAZ**

Para el diseño de la interfaz se necesita tener claro la perspectiva general del sistema, teniendo en cuenta su funcionamiento básico y las restricciones a las cuales se rige el dispositivo Kinect como el cuadricóptero. En vista de ello, a continuación se describe al sistema, así como se listan sus requisitos y demás consideraciones necesarias para que el sistema se funcione efectivamente.

##### **3.1.1 Descripción del sistema**

La interfaz de control se basa en una aplicación que mediante el dispositivo Kinect permite al usuario la interacción con el cuadricóptero para controlar sus movimientos. La aplicación fue desarrollada en Visual Studio 2013 con el lenguaje de programación C#- WPF. La interfaz de control funciona con los movimientos que el operador realiza con sus manos para que el cuadricóptero pueda elevarse, descender, desplazarse a la izquierda, desplazarse a la derecha, hacia adelante o hacia atrás.

##### **3.1.2 Requisitos del sistema**

Los elementos para que el funcionamiento del sistema cumpla su cometido son los siguientes:

**Dispositivo Kinect.-** Aparato que captura los movimientos del usuario.

El sistema requiere que el dispositivo Kinect se encuentre dentro de un ambiente controlado para la toma de señales, esto implica que no debe existir exposición al sol, ambientes de lluvia y luz reflectante hacia la cámara.

**Computador.-** Dispositivo encargado del procesamiento de la información, debe estar equipado con un procesador Core i3 o superior, para que la aplicación funcione sin ningún problema.

**SDK de Microsoft.-** Kit de desarrollo de software que posee las librerías para el funcionamiento del dispositivo Kinect.

**Placa de control.-** Circuito Electrónico encargado del manejo de señales desde el computador hacia el controlador remoto.

**Controlador Remoto.-** Dispositivo que envía las señales a través de una red inalámbrica WiFi al cuadricóptero.

**Cuadricóptero Phantom 3 Standard.-** UAV que recibe la información del usuario a través de la interfaz de control.

El cuadricóptero requiere que no existan condiciones climáticas adversas para su uso, tales como lluvia, nieve, niebla o velocidades de viento mayores a 10 m/s, se lo debe utilizar en espacios abiertos.

### 3.1.3 Diagrama de bloques del sistema

En la siguiente figura se puede identificar un diagrama de bloque del sistema propuesto:



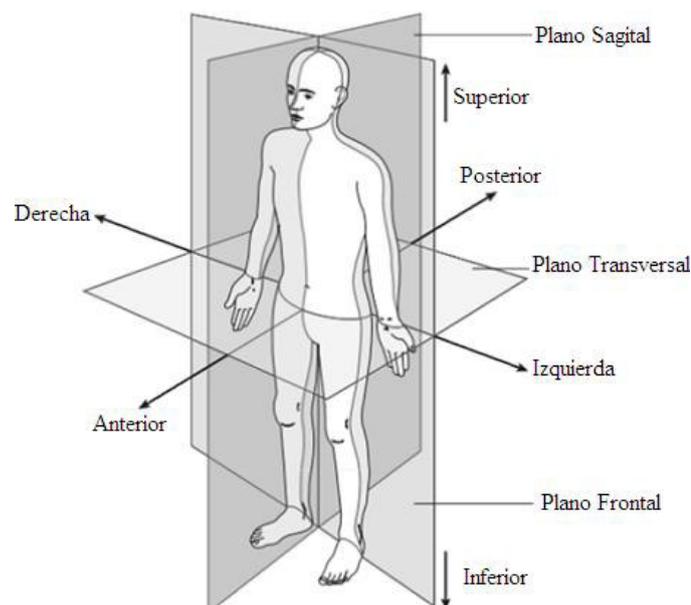
**Figura 32 Diagrama de bloques del sistema**

## 3.2 ANÁLISIS DE LOS TIPOS DE MOVIMIENTOS PARA CONTROLAR EL CUADRICÓPTERO

Para analizar las características propias de cada persona se estudia el movimiento del cuerpo humano, con la finalidad de utilizar técnicas en el desarrollo de aplicaciones de interacción en tiempo real.

### 3.2.1 Análisis biomecánico

La aplicación de métodos y técnicas de la ciencia mecánica que analiza el movimiento de los organismos vivos es la biomecánica. En este sentido, la descripción del movimiento anatómico humano explica el movimiento entre los huesos y el rango de cada articulación en tres planos del cuerpo (Cuenca, Delgado, & Chuya, 2013). Así, el movimiento se identifica en tres tipos de plano respecto al cuerpo humano. El primero es el plano frontal o coronal, que refiere a la parte anterior y posterior. Seguido, se encuentra el plano transversal, que refiere a al nivel superior e inferior. Finalmente, el plano sagital o lateral identifica la izquierda y derecha del cuerpo referido (Ibídem, p. 8). La figura número 33 muestra la división de los planos anatómicos.

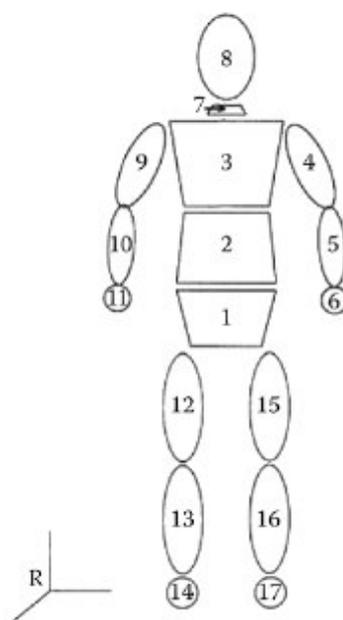


**Figura 33 División de los planos anatómicos**

Fuente: (Cuenca, Delgado, & Chuya, 2013)

### 3.2.2 Modelo biomecánico

Al igual que toda materia, el cuerpo de un ser humano tiene capacidad de moldeado respecto a cada una de sus partes, lo que depende de las necesidades e intereses del movimiento (Cuenca, Delgado, & Chuya, 2013, p. 9). Los nombres de cada segmento del modelo de la figura 34 se encuentran en la tabla 8.



**Figura 34 Estructura de un modelo biomecánico**

Fuente: (Cuenca, Delgado, & Chuya, 2013)

**Tabla 8**  
**Segmentos del modelo biomecánico**

Nº	SEGMENTOS	
R	Sistema de referencia inercial	9 Parte superior del brazo derecho
1	Pelvis(parte inferior del cuerpo)	10 Antebrazo derecho
2	Cintura(parte medio del cuerpo)	11 Mano derecha
3	Tórax(parte superior del cuerpo)	12 Muslo derecho
4	Parte superior del brazo izquierdo	13 Pierna derecha
5	Antebrazo izquierdo	14 Pie derecho
6	Mano izquierda	15 Muslo izquierdo
7	Cuello	16 Pierna izquierda
8	Cabeza	17 Pie Izquierdo

Fuente: (Cuenca, Delgado, & Chuya, 2013)

Los segmentos que participan en la movilidad se agrupan en cuatro segmentos de las extremidades inferiores y se muestran en la siguiente tabla:

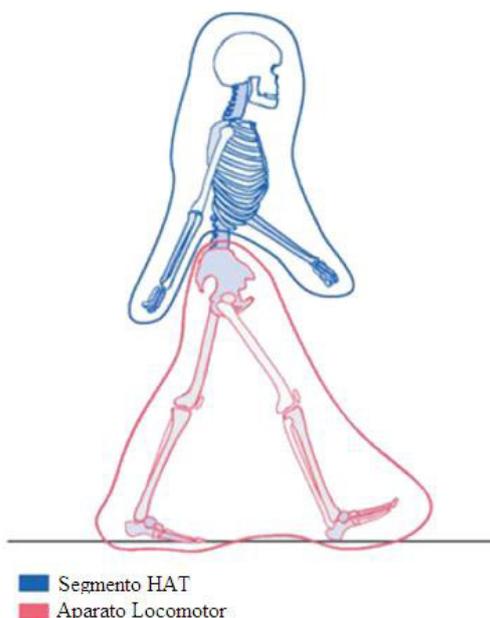
**Tabla 9**  
**Segmentos que participan en la movilidad**

<b>Segmentos</b>	<b>Descripción</b>
Cintura	Transmite la fuerza que se ejerce sobre ella y la adaptación de la pelvis, que se inclina o se balancea en respuesta al apoyo unilateral.
Cadera y muslo	Proporcionan al miembro inferior la posibilidad de desplazamiento en las tres direcciones del espacio, siendo siempre el muslo el segmento más móvil.
La rodilla y pierna	Aseguran la solidez del movimiento en la locomoción vertical, proporcionan también estabilidad en la sucesión de las fases de apoyo y suspensión durante el desplazamiento y el salto
Las diferentes partes del pie	Tienen como función la adaptación al suelo en las diferentes formas de locomoción.

**Fuente:** (Cuenca, Delgado, & Chuya, 2013)

### **3.2.3 Segmentación del Cuerpo Humano**

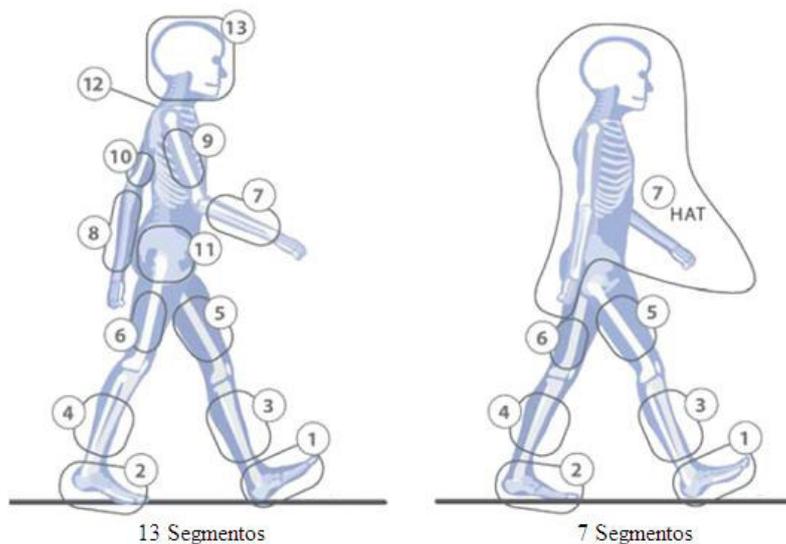
El análisis de movimiento del cuerpo humano comprende un conjunto de segmentos unidos entre sí, en el cuál, de acuerdo al concepto de Perry (citado en Cuenca, Delgado, & Chuya, 2013), sugiere que hay que considerar el segmento de carga y los segmentos del aparato locomotor. En el segmento de carga se localiza la cabeza, los brazos y el tronco. Al conjunto de estos segmentos se denomina segmento HAT. Los segmentos del aparato locomotor incluyen: la pelvis y extremidades inferiores (pie, pierna y muslo). Estos segmentos se articulan en el tobillo, rodilla, cadera y la unión lumbosacra, tal como se muestra en la figura 35.



**Figura 35 Segmentos del cuerpo humano**

**Fuente:** (Cuenca, Delgado, & Chuya, 2013)

El modelo actual de siete segmentos tiene dos pies (1, 2), dos piernas (3, 4), dos muslos (5, 6), y el segmento HAT (7). Con la evolución tecnológica actual se tiene un nuevo modelo en donde el cuerpo se divide en trece segmentos (figura 36), en este modelo la parte superior consta de seis partes adicionales al modelo anterior. Las partes que se incorporan son: la cadera (11), el tronco (12), brazo (9, 10) y antebrazo (7, 8), y cabeza (13).



**Figura 36 Segmentación actual del cuerpo humano**

**Fuente:** (Cuenca, Delgado, & Chuya, 2013)

### **3.2.4 Captura de movimiento**

La captura de movimiento es un proceso según el cual se obtienen movimientos del plano real y se los representa de forma digital en un sistema a través de un conjunto de puntos o coordenadas (Cuenca, Delgado, & Chuya, 2013, pág. 13). El conjunto de puntos principales en la persona se los ubica en las áreas de mayor información de sus movimientos. Dichos puntos hacen referencia a las partes rígidas de la persona, tales como las diferentes articulaciones que posee el cuerpo humano. Para la captura de los movimientos, los nuevos sistemas utilizan la tecnología actual que se basa en algoritmos, los que analizan la entrada de datos entregados por las cámaras e identifican las figuras humanas para la obtención del seguimiento.

Los dispositivos actuales que utilizan los videojuegos pueden capturar los movimientos de los usuarios y, gracias a sus costos y fiabilidad en los datos obtenidos, hacen que el estudio del movimiento humano sea más factible. Para la captura de los movimientos que realiza el usuario se utiliza el dispositivo Kinect. Como primer paso es necesario analizar los diferentes modelos de este dispositivo para escoger el que más se acople a las necesidades del proyecto. Además, es necesario elegir adecuadamente el kit de desarrollo de software que permita capturar los datos del dispositivo Kinect.

#### **3.2.4.1 Requisitos para la Captura de Movimientos**

Los requisitos necesarios del dispositivo Kinect y sus librerías se detallan a continuación:

##### *Dispositivo Kinect*

- Cámara RGB
- Sensor de profundidad
- Arreglo de micrófonos

- Ajuste de ángulo de elevación ( $\pm 27^\circ$ )
- Ángulo de visión ( $57^\circ$  horizontal y  $43^\circ$  vertical)
- Voltaje de alimentación 12 V
- Rango de distancia de 0.8m a 4m

Librerías que deberá tener el SDK

- Reconocimiento de las articulaciones que forma el esqueleto humano
- Reconocimiento de Voz

### 3.2.4.2 Selección del dispositivo Kinect

Actualmente, se dispone comercialmente tres tipos de modelos: el Kinect para XBOX 360, el Kinect para XBOX ONE y el Kinect para Windows. Los tres modelos poseen las librerías para su funcionamiento, así como sus respectivos adaptadores para la PC. En esta etapa se evalúan los tres modelos para escoger el más adecuado para el proyecto. En la tabla 10 se muestra las características más importantes de los tres modelos.

**Tabla 10**  
**Características de los modelos de Kinect**

<b>Características</b>	<b>XBOX 360</b>	<b>XBOX ONE</b>	<b>WINDOWS</b>
Cámara RGB	Si	Si	Si
Sensor de Profundidad	Si	No	Si
Arreglo de Micrófonos	Si	Si	Si
Ajuste de ángulo de elevación( $\pm 27^\circ$ )	Si	No	Si
Ángulo de visión ( $57^\circ$ horizontal y $43^\circ$ vertical)	Si	No	Si
Rango de distancia (0.8 m a 4m)	Si	No	Si
Detección de dedos y muñecas	No	Si	No
Detección de músculos	No	Si	No
Costo	\$150	\$150	\$250

Como se puede observar en la tabla 10, los modelos de Kinect que más se adaptan al proyecto son el XBOX 360 y el de Windows, ya que ambos poseen las dos características principales para el desarrollo de la interfaz,

como son el sensor de profundidad y el arreglo de micrófonos. A este punto, se debe considerar que para el desarrollo de cualquier aplicación el costo que se invierta en la misma es de vital importancia, por tal motivo se eligió el XBOX 360.

### 3.2.4.3 Selección del Kit de Desarrollo de Software

El siguiente paso luego de escoger el modelo de Kinect es elegir, adecuadamente, el software de desarrollo. En el capítulo anterior se analizó en la tabla 7 las ventajas y desventajas de utilizar el SDK de Microsoft respecto al SDK OPENNI/NITE. Además, se pudo comprobar que el SDK más adecuado para el proyecto era el de Microsoft. En la tabla 11 se muestra a manera de resumen la comparación de las principales características del SDK de código abierto frente al SDK de Microsoft.

**Tabla 11**  
**Comparación de los SDK Microsoft vs Open Source**

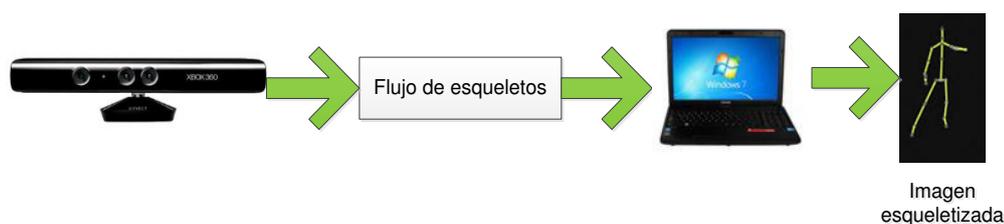
<b>Característica</b>	<b>SDK Microsoft</b>	<b>Open Source</b>
Facilidad de Adquisición	Si	No
Facilidad de Instalación	Si	No
Acceso a la Documentación	Si	No
Soporte a múltiples Sensores	Si	No
Sistema Operativo Windows	Si	Si
Precisión y velocidad en el reconocimiento de articulaciones	Si	No

La documentación que existe actualmente acerca del SDK de Microsoft lo convierte en la mejor alternativa de uso, ya que cuenta con las herramientas y el soporte adecuado para facilitar el desarrollo del proyecto. El SDK de Microsoft es el elegido porque, además de las características de la tabla 11, brinda la posibilidad de acceder a los datos de profundidad, audio y esqueletos necesarios para la manipulación en la implementación de la interfaz. Otra ventaja que se obtiene al utilizar el hardware y el software de la misma empresa es la estabilidad del sistema y la compatibilidad con todas las actualizaciones que sean necesarias en el transcurso de la

implementación. Cabe mencionar que, aunque la alternativa de código abierto sirve para trabajar en multiplataformas, su uso no es de alta frecuencia en otras plataformas ya que la interfaz fue desarrollada en Windows.

### 3.2.4.4 Diagrama de bloques utilizado en la captura de Movimientos

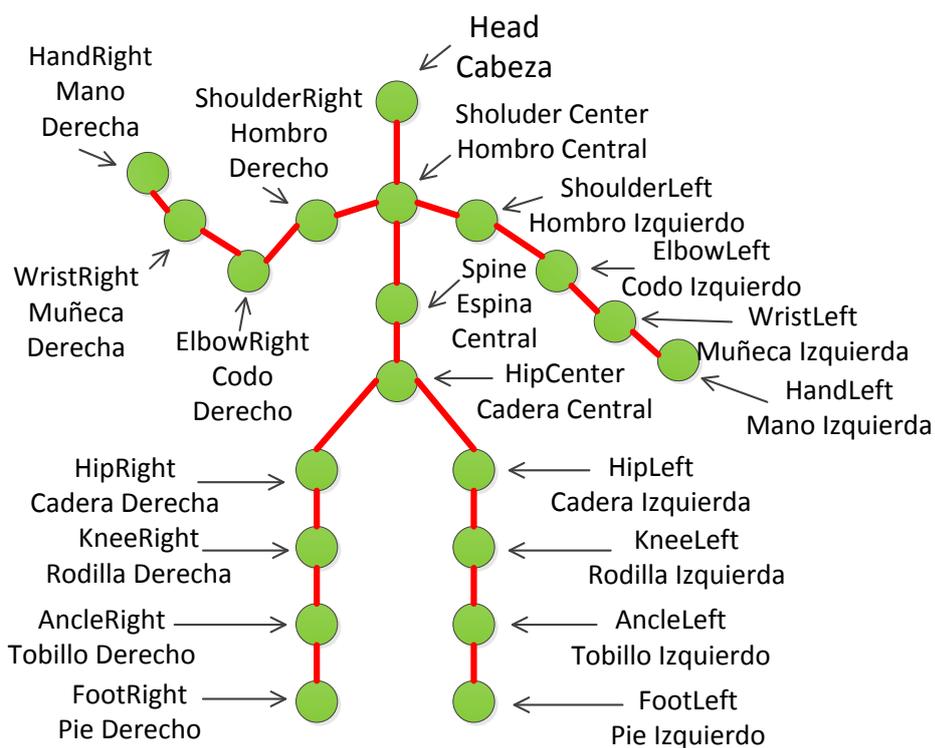
El siguiente paso luego de elegir el modelo del dispositivo Kinect y las librerías es explicar el proceso que realiza el dispositivo en la captura de los datos de movimiento por parte del usuario. El Kinect, por ejemplo, utiliza las cámaras de profundidad para la captura de datos del esqueleto humano. El SDK de Microsoft tiene funciones que nos permiten acceder a los flujos de datos que capturan las cámaras del dispositivo Kinect.



**Figura 37 Diagrama de captura de movimientos**

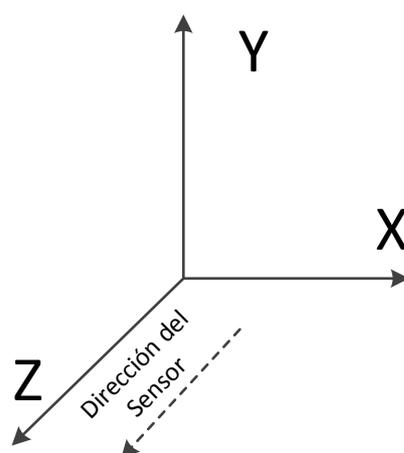
### Flujo de Esqueletos

El SDK elegido reconoce al cuerpo humano a través de 20 articulaciones y 19 huesos. Los datos son obtenidos mediante la combinación del Emisor IR y la cámara de profundidad del dispositivo Kinect. Para trabajar con el esqueleto los desarrolladores utilizan los nombres en inglés de los huesos y articulaciones, que por su denominación en inglés son *bones* y *joints*. La Figura 38 muestra los nombres de las 20 articulaciones que componen el esqueleto humano y que se encuentran disponibles en el SDK.



**Figura 38 Articulaciones disponibles en el SDK de Microsoft**

Otra de las funciones que brinda el SDK es la posibilidad de obtener los datos de las articulaciones en 3D. En la figura 39 se muestra el plano de coordenadas X, Y, Z que utiliza el Kinect para detectar las coordenadas de las articulaciones.



**Figura 39 Plano de Referencia de las Coordenadas de las Articulaciones.**

### 3.3 DISEÑO DEL HARDWARE PARA EL PROCESAMIENTO DE INFORMACIÓN

Para el diseño del circuito electrónico, que conforma la placa de control, se utilizó el programa de simulación y diseño Proteus, según el que se consideró el uso de los pines necesarios para la conexión entre el controlador remoto, la placa de control y la computadora.

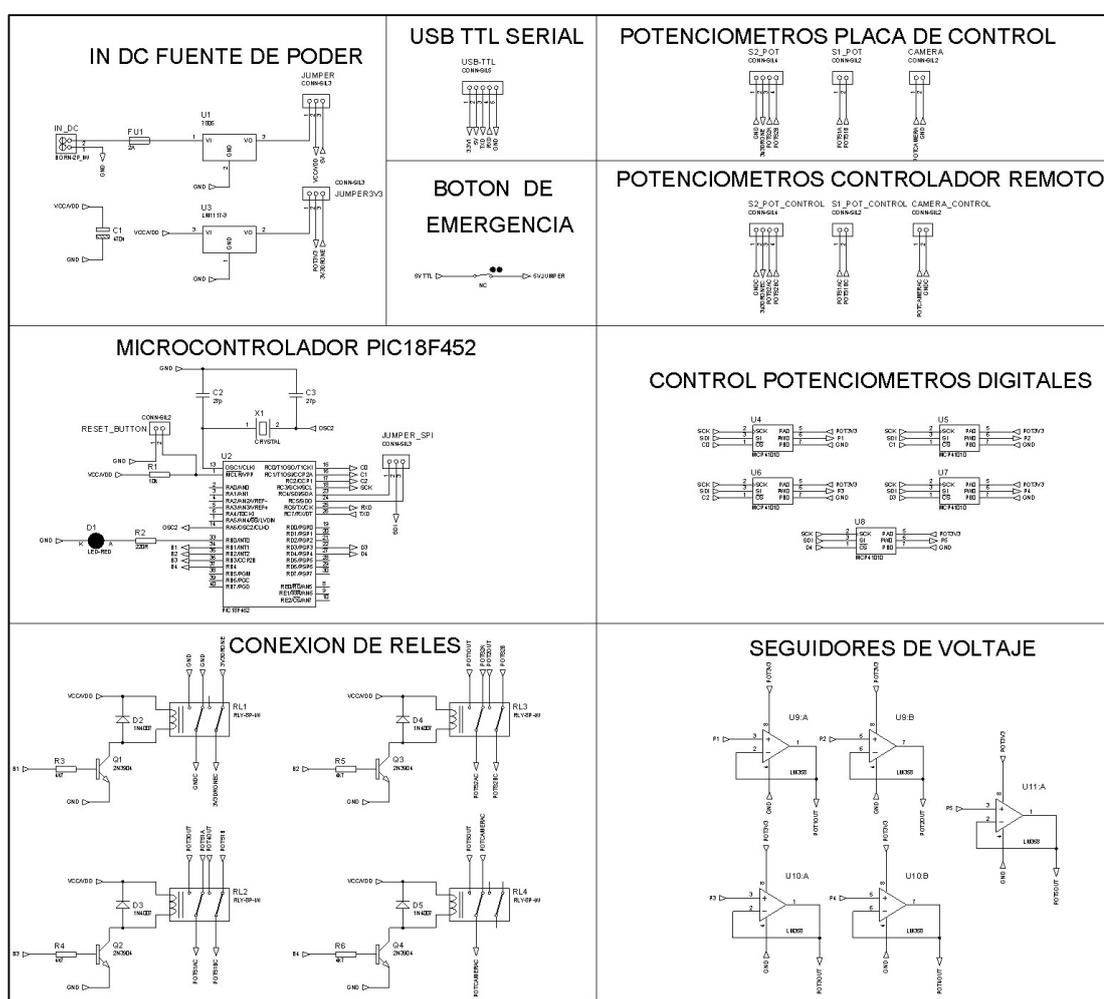
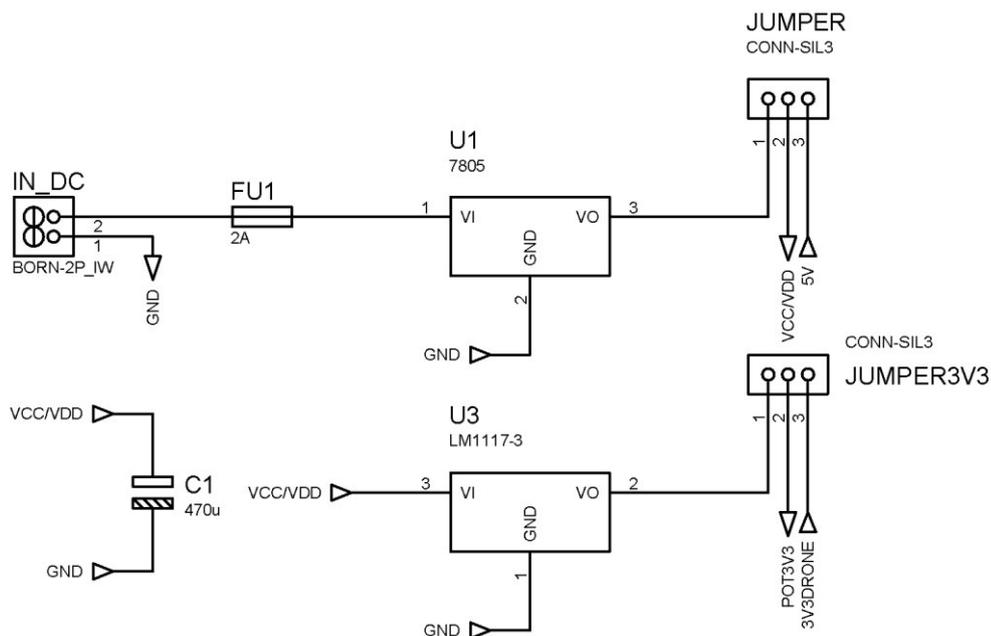


Figura 40 Diseño Electrónico de la conexión entre elementos de la placa

A continuación se explica cómo funciona cada etapa del circuito electrónico.

### 3.3.1 Etapa IN DC FUENTE DE PODER

El diseño de esta etapa comprende la entrada de alimentación de voltaje para los componentes que conforman la placa de control. En la figura 41 se muestra el diseño electrónico y en la tabla 12 se muestra los componentes y una breve descripción del papel que realiza cada uno en el circuito,



**Figura 41 Diseño Electrónico de la etapa de alimentación**

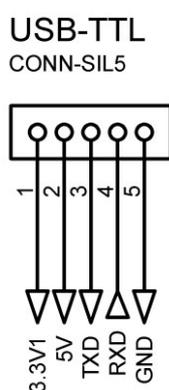
**Tabla 12**  
**Componentes etapa IN DC FUENTE DE PODER**

Componentes	Descripción
IN_DC	Conector para la entrada de voltaje
FU1	Conector para un fusible (en caso de ser necesario)
U1 (7805)	Regulador de Voltaje de 5 Voltios
JUMPER	Permite seleccionar una fuente de entrada por la bornera DC o utilizar la fuente USB del computador
C1	Capacitor para estabilizar voltajes de la fuente
U3 (LM1117-3)	Regulador de voltaje de 3.3V
JUMPER3V3	Permite seleccionar un voltaje para los potenciómetros digitales desde el controlador remoto del cuadricóptero o desde la fuente.

El JUMPER está en la posición 2-3 para que el circuito electrónico se alimente con los 5V que suministra la entrada USB del computador.

### 3.3.2 Etapa USB TTL SERIAL

Para el diseño de esta etapa se utiliza un conector FTDI, de aquí se obtiene los voltajes de 3.3V y 5V para alimentar a los componentes de la placa. Esta etapa permite la comunicación serial entre la computadora y la placa de control. El FTDI es un conversor USB a TTL.



**Figura 42 Diseño Electrónico de la etapa USB TTL SERIAL**

### 3.3.3 Etapa BOTON DE EMERGENCIA

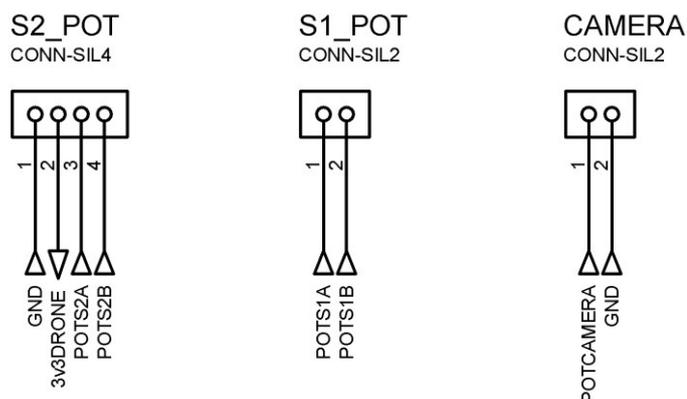
El diseño de esta etapa comprende el uso de un interruptor normalmente cerrado el cual está conectado entre la etapa USB TTL SERIAL y la etapa de alimentación. Éste sirve, específicamente, para cortar la energía de la placa de control cuando se tiene alguna falla en el sistema o cuando el usuario pierde el control del cuadricóptero. Cuando el interruptor es presionado cambia a la posición de normalmente abierto y entra en funcionamiento únicamente el controlador remoto del cuadricóptero.



**Figura 43 Diseño Electrónico de la etapa BOTON DE EMERGENCIA**

### 3.3.4 Etapa POTENCIOMETROS PLACA DE CONTROL

Esta etapa comprende el diseño de los conectores para los cables que vienen desde controlador remoto del cuadricóptero.



**Figura 44 Diseño Electrónico de la etapa POTENCIOMETROS PLACA DE CONTROL**

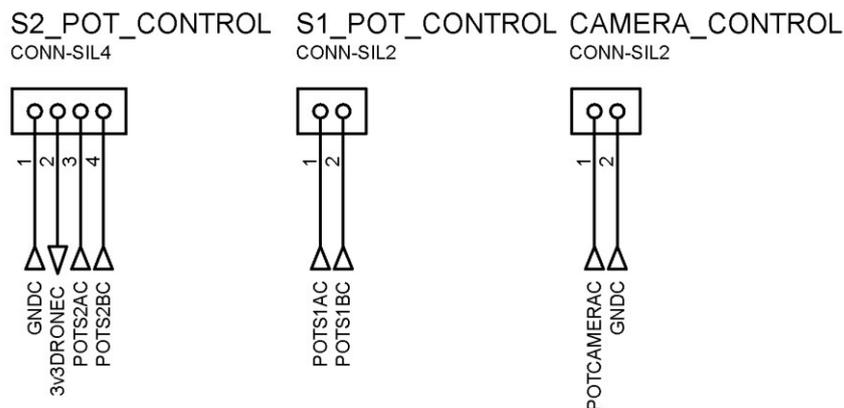
**Tabla 13**

**Componentes etapa POTENCIOMETROS PLACA DE CONTROL**

Componentes	Descripción
S2_POT	Conector al potenciómetro S2 del controlador remoto
S1_POT	Conector al potenciómetro S1 del controlador remoto
CAMERA	Conector al potenciómetro de la cámara del controlador remoto

### 3.3.5 Etapa POTENCIOMETROS CONTROLADOR REMOTO

Esta etapa comprende el diseño de los conectores para los cables que salen desde la Placa de Control Electrónica (PCE).



**Figura 45 Diseño Electrónico de la etapa POTENCIOMETROS CONTROLADOR REMOTO**

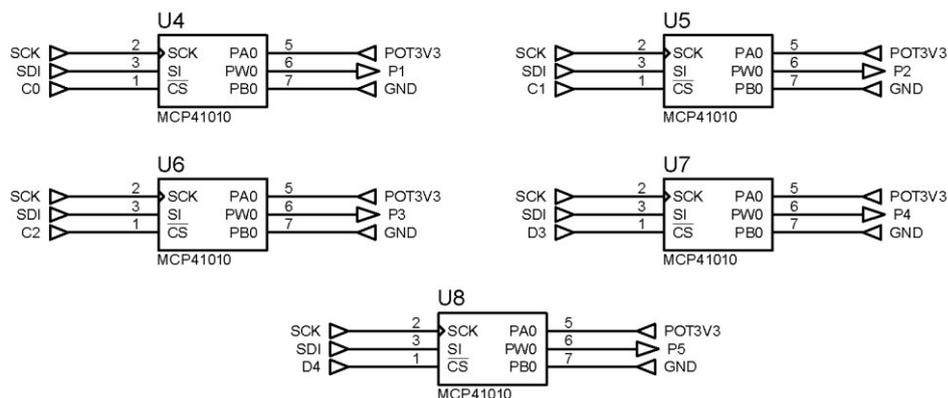
Tabla 14

**Componentes etapa POTENCIOMETROS CONTROLADOR REMOTO**

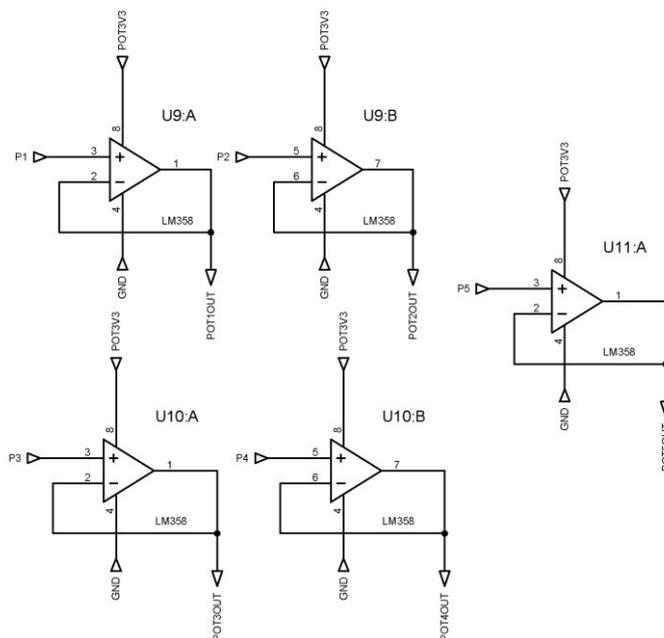
Componentes	Descripción
S2_POT_CONTROL	Conector al potenciómetro S2 desde la PCE
S1_POT_CONTROL	Conector al potenciómetro S1 desde la PCE
CAMERA CONTROL	Conector a la cámara desde la PCE

**3.3.6 Etapa CONTROL POTENCIOMETROS DIGITALES**

Para el diseño de esta etapa se utilizan los potenciómetros digitales MCP41010 (U4, U5, U6, U7 y U8). Estos potenciómetros trabajan con el Protocolo Síncrono Serial (SPI) en una red Bus de potenciómetros. Lo que selecciona la comunicación entre cada potenciómetro es el chip select al cuál están conectados a los pines P1, P2, P3, P4 y P5.

**Figura 46 Diseño Electrónico de la etapa CONTROL POTENCIOMETROS DIGITALES****3.3.7 Etapa SEGUIDORES DE VOLTAJE**

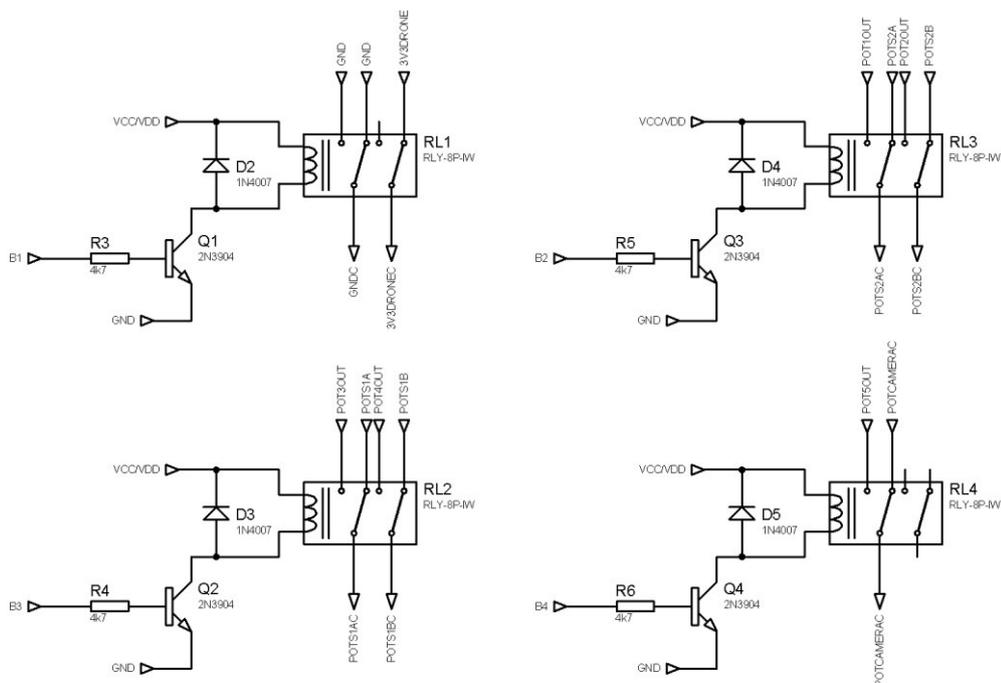
Para el diseño de esta etapa se utilizan los amplificadores operacionales LM358 (U9, U10 y U11) conectados en modo seguidor de tensión para fortalecer la señal de salida de los potenciómetros digitales.



**Figura 47 Diseño Electrónico de la etapa SEGUIDORES DE VOLTAJE**

### 3.3.8 Etapa CONEXIÓN DE RELES

El diseño de esta etapa comprende el uso de los relés dobles que funcionan a 5V para seleccionar entre el funcionamiento del controlador remoto o el funcionamiento de la placa de control electrónica.



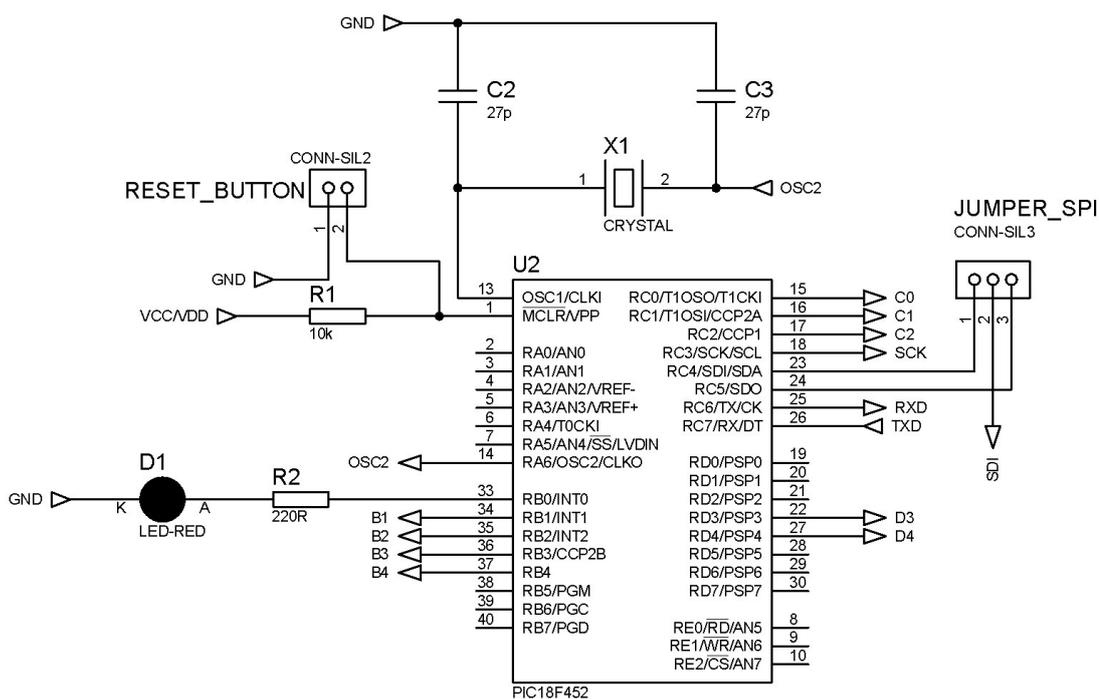
**Figura 48 Diseño Electrónico de la etapa CONEXIÓN DE RELES**

**Tabla 15**  
**Componentes etapa CONEXIÓN DE RELÉS**

Componentes	Descripción
RL1,RL2,RL3,RL4	Relés que permiten conmutar entre controlador remoto, o el manejo desde la PCE
Q1,Q2,Q3,Q4	Transistores que activan los relés de polarización NPN
D2,D3,D4,D5	Diodos de protección, son útiles para evitar reseteo de la tarjeta
R3,R4,R5,R6	Resistencias base de los transistores Q1,Q2,Q3,Q4

### 3.3.9 Etapa MICROCONTROLADOR PIC 18F452

El diseño de esta etapa comprende el uso del microcontrolador 18F452 el cuál fue elegido gracias a su firmware que nos permite trabajar con el Protocolo Síncrono Serial (SPI).

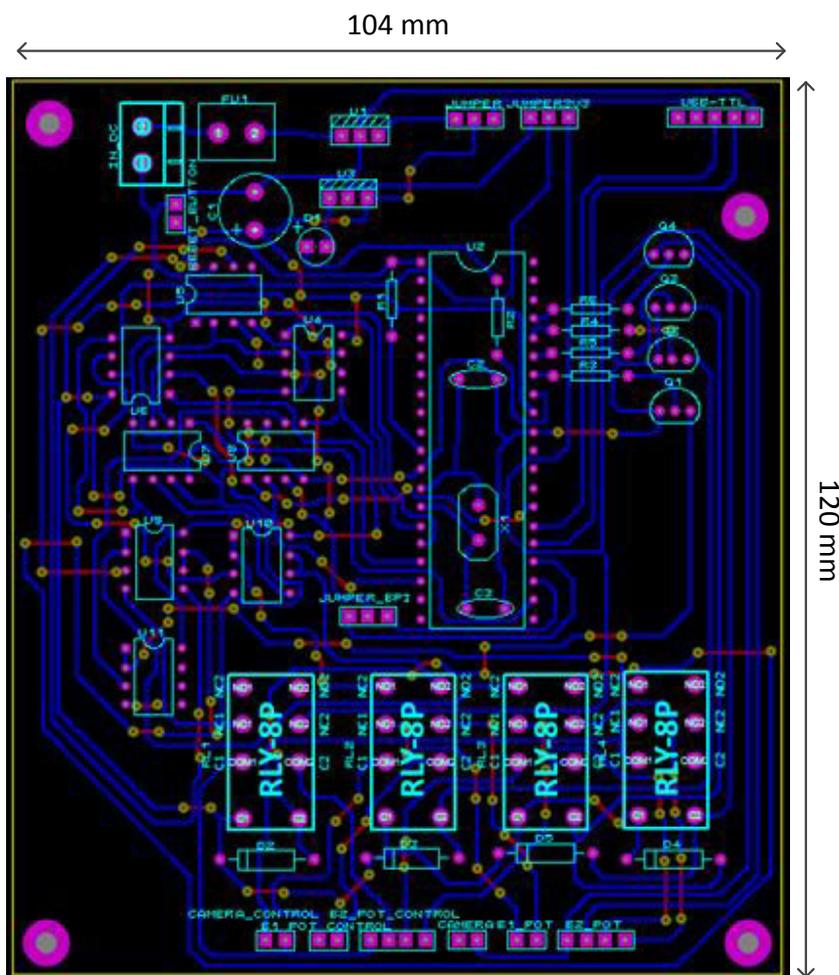


**Figura 49 Diseño Electrónico de la etapa MICROCONTROLADOR PIC 18F452**

**Tabla 16**  
**Componentes etapa MICROCONTROLADOR PIC 18F452**

Componentes	Descripción
U2 (PIC18F452)	Microcontrolador, utilizado para manejar las salidas de los potenciómetros digitales
RESET BUTTON	Botón de reset para todo el microcontrolador
X1	Cristal que da los tiempos de trabajo al microcontrolador
C2,C3	Capacitores para el cristal X1, la especificación del datasheet sugiere entre 22 pF y 27 pF
JUMPER_SPI	Permite seleccionar el funcionamiento del Protocolo Síncrono Serial para que trabaje como MOSI o MISO
D1	Led para identificar funcionamiento de encendido correcto del microcontrolador

### 3.3.10 Diseño de la placa de control



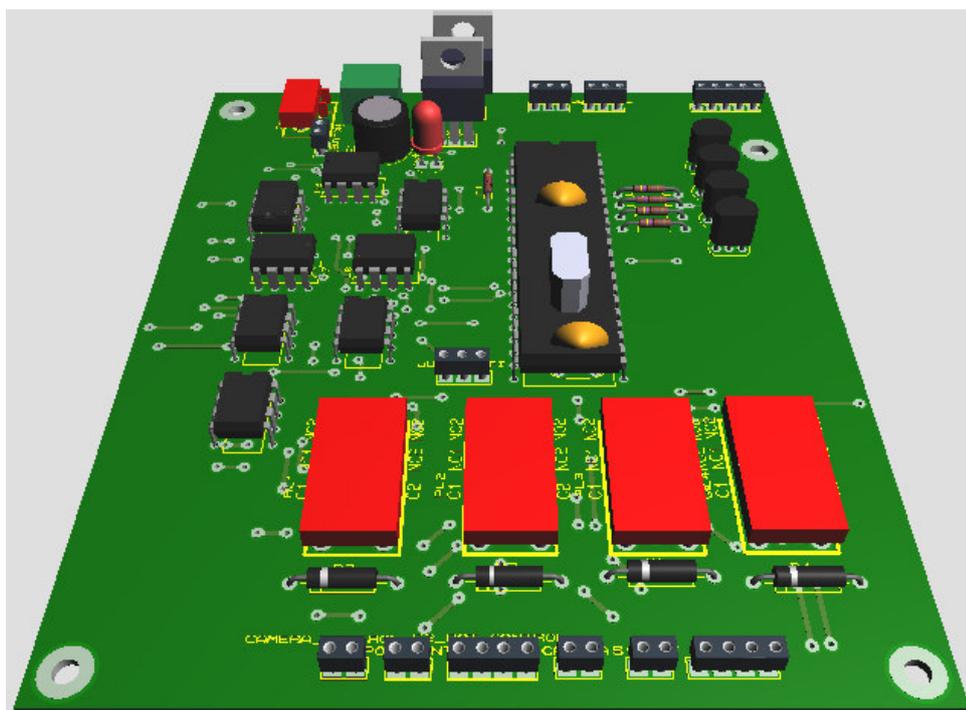
**Figura 50** Diseño de la placa de control

## ELEMENTOS

Los elementos electrónicos para la placa de control se describen en la tabla 17

**Tabla 17**  
**Elementos de la Placa de Control**

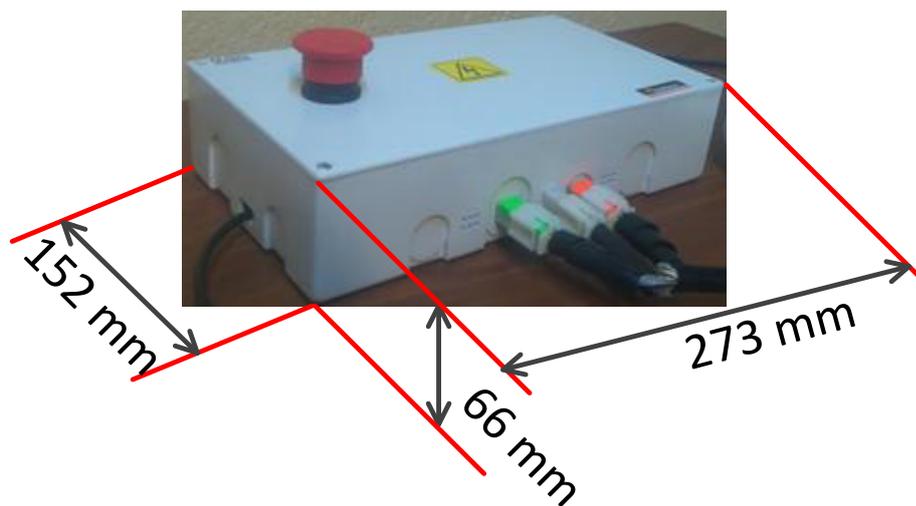
<b>CANTIDAD</b>	<b>DESCRIPCION</b>
1	Capacitor electrolítico 470 uF/35V
2	Capacitores cerámicos 22pF
1	Resistencia de 10 KΩ
1	Resistencia de 220Ω
4	Resistencias 4,7 KΩ
1	LM7805
1	PIC18F452
1	LM1117
5	MCP41010
3	LM358
4	TRANSISTORES 2N3904
1	LED 5MM NORMAL
4	DIODOS 1N4007
1	FUSIBLE 1A
1	ZOCALO FUS CHASIS
5	CABLES MOLEX 2 P
2	CABLES MOLEX 4 P
1	BORNERA 2P
4	JUMPERS
4	RELES 5V/8P
1	CONECTOR MACHO SIMPLE
1	CONECTOR MACHO EN L
1	USB - TTL
1	ZOCALO 40P
8	ZOCALOS 8P
1	PLACA PCB
3	CABLES ARDUINO



**Figura 51 Vista 3D del diseño completo de la tarjeta de control**

### 3.3.11 Dimensiones de la caja para la placa de control

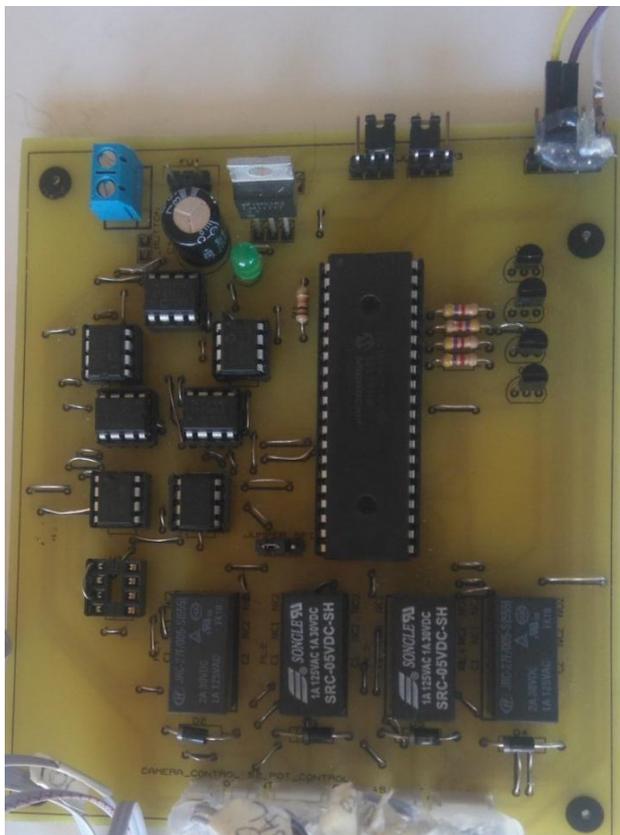
La caja donde se coloca la tarjeta de control es de un material plástico industrial y tiene las siguientes dimensiones.



**Figura 52 Dimensiones de la caja para la tarjeta de control**

### 3.3.12 Implementación del circuito electrónico

La figura 53 muestra la tarjeta de control terminada y la figura 54 muestra la caja de control implementada



**Figura 53 Vista la tarjeta de control implementada**



**Figura 54 Caja de control implementada**

### 3.4 DISEÑO DE LA ETAPA DE PROCESAMIENTO DE INFORMACIÓN

El diseño de esta etapa comprende la implementación de la interfaz gráfica de usuario, que es conocida por su denominación en inglés como *Graphical User Interface* (GUI). Es la etapa primordial del proyecto donde se reconocen y validan los movimientos que realiza el operador. Además, se procesa la información, misma que es enviada a la placa electrónica que se comunica con el controlador remoto y, por ende, con el cuadricóptero.

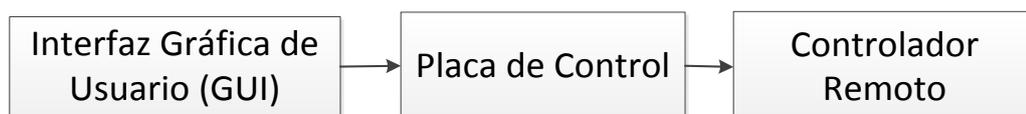
#### 3.4.1 Requisitos de la Etapa de Procesamiento de Información

Los requerimientos que debe cumplir la interfaz de control son los siguientes:

- Interfaz gráfica amigable y fácil de usar para el usuario.
- Interactuar con el dispositivo Kinect
- Reconocer y procesar los movimientos del usuario
- Conexión con la placa de control electrónica

#### 3.4.2 Diagrama de bloques de la Etapa de Procesamiento de Información

La figura 55 presenta el diagrama de bloques de la etapa de procesamiento de información



**Figura 55 Diagrama de bloques de la etapa de procesamiento de información**

## Software

Interfaz Gráfica de Usuario, o GUI por sus siglas en inglés, es la que recibe la información proporcionada por el dispositivo Kinect y la procesa para que pueda ser enviada al cuadricóptero a través de la placa de control y el controlador remoto.

## Hardware

La Placa de Control es el circuito electrónico que recibe la información de la interfaz de control. Se encarga del manejo de señales desde el computador hacia el controlador remoto.

El Controlador Remoto es el que envía los datos procesados por la placa de control hacia el cuadricóptero a través de una red inalámbrica WiFi.



**Figura 56 Hardware para el procesamiento de información**

### 3.4.3 Requisitos de la Interfaz Gráfica de Usuario (GUI)

La interfaz de control analiza los movimientos que realiza el usuario a través del dispositivo Kinect. Éste los procesa y verifica para determinar que el movimiento realizado sea el correcto para la manipulación del cuadricóptero. La interfaz gráfica de usuario presenta una pantalla donde se muestra el flujo del esqueleto del operador, mismo que permite ver las

coordenadas de las manos en los ejes X, Y, Z y muestra el estado en el que se encuentra el cuadricóptero. Además, éste consta de tres sliders los cuales representan el movimiento de las manos para controlar el desplazamiento del cuadricóptero. Con ello el dispositivo pueda elevarse, descender, desplazarse a la izquierda y derecha, hacia adelante y atrás. Para que el funcionamiento de la aplicación sea el correcto se deben tener en cuenta los siguientes requerimientos:

- **Requerimientos Funcionales.-** Describen el funcionamiento del sistema
- **Requerimientos No Funcionales.-** Son criterios usados para juzgar la operación del sistema como por ejemplo: rendimiento, interfaz, recursos, operación, comprobación, seguridad, calidad, documentación y mantenimiento.

La especificación de requisitos de software (ERS) de acuerdo con algunas características definidas en el estándar IEEE 830-1998 debe ser completa, inequívoca, consistente y correcta. La tabla 18 muestra los requisitos funcionales y no funcionales que se debe tomar en cuenta para efectuar una adecuada especificación de los requisitos de software.

**Tabla 18**  
**Nomenclatura de los Tipos de Requisitos de la Aplicación**

<b>Identificador</b>	<b>Tipo de Requisito</b>
<b>RF</b>	Requisito Funcional
<b>RNFR</b>	Requisito No Funcional de Rendimiento
<b>RNFI</b>	Requisito No Funcional de Interfaz
<b>RNFO</b>	Requisito No Funcional de Operación
<b>RNFC</b>	Requisito No Funcional de Comprobación
<b>RNFD</b>	Requisito No Funcional de Documentación
<b>RNFM</b>	Requisito No Funcional de Mantenimiento
<b>RNFU</b>	Requisito No Funcional de Usabilidad
<b>RNFS</b>	Requisito No Funcional de Soporte

Fuente: (Sánchez E. , 2012)

## REQUISITOS FUNCIONALES

**Tabla 19**  
**Requisitos Funcionales de la Aplicación.**

Identificador	Título	Descripción
RF01	Iniciar Aplicación	El operador debe iniciar la aplicación
RF02	Ver Opciones de aplicación en el Menú de Acceso	La pantalla despliega la lista de opciones iniciales del sistema (Ayuda, Información, Interfaz)
RF03	Cerrar Aplicación	El sistema cierra la aplicación

**Tabla 20**  
**Requisitos Funcionales de la Opción “Interfaz”**

Identificador	Título	Descripción
RF04	Serie	Detecta si el puerto serial está abierto
RF05	Kinect	Detecta si el dispositivo está conectado
RF06	Centros	Detecta que ambas manos se encuentren en la posición central
RF07	Encendido	Detecta que ambas manos se eleven y regresen a la posición central
RF08	Control	Detecta que las manos derecha e izquierda realicen movimientos adecuados
RF09	Aterrizar	Detecta que las manos se encuentren abajo

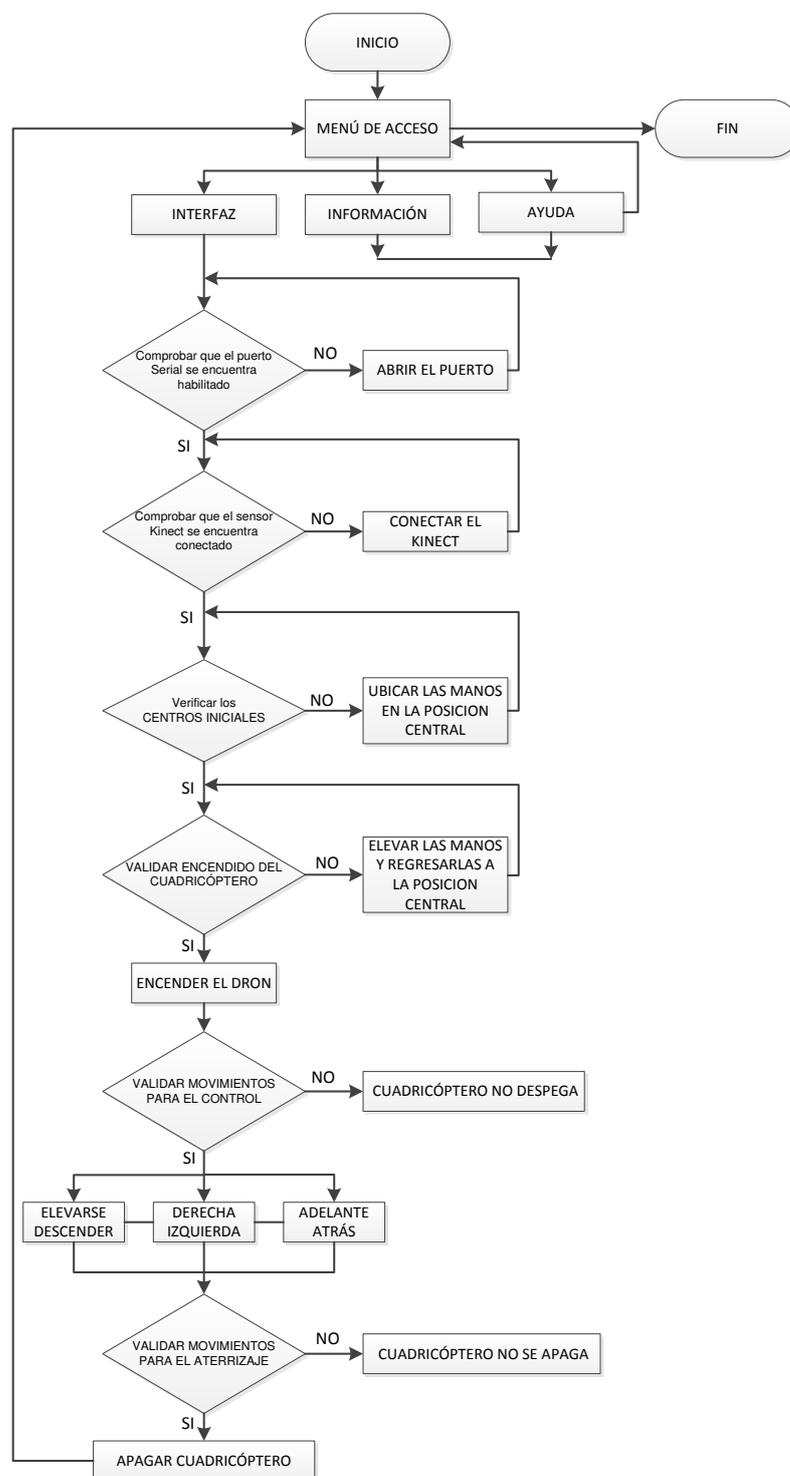
## REQUISITOS NO FUNCIONALES

**Tabla 21**  
**Requisitos No Funcionales de la Aplicación.**

Identificador	Título	Descripción
RNFR01	Usuarios del sistema	El sistema admite un usuario
RNFI01	Interfaz Gráfica Usuario	Aplicación con imágenes y colores
RNFI02	Contenido del Menú	Opciones claras para el usuario
RNFO01	Acceso a la Aplicación	A través de Visual Studio 2013
RNFU01	Facilidad de manejo	No se necesita conocimientos avanzados de computación
RNFS01	Plataforma	Sistema Operativo Windows 7 o superiores y SDK de Microsoft

### 3.4.4 Diagrama de Flujo de la Interfaz Gráfica de Usuario (GUI)

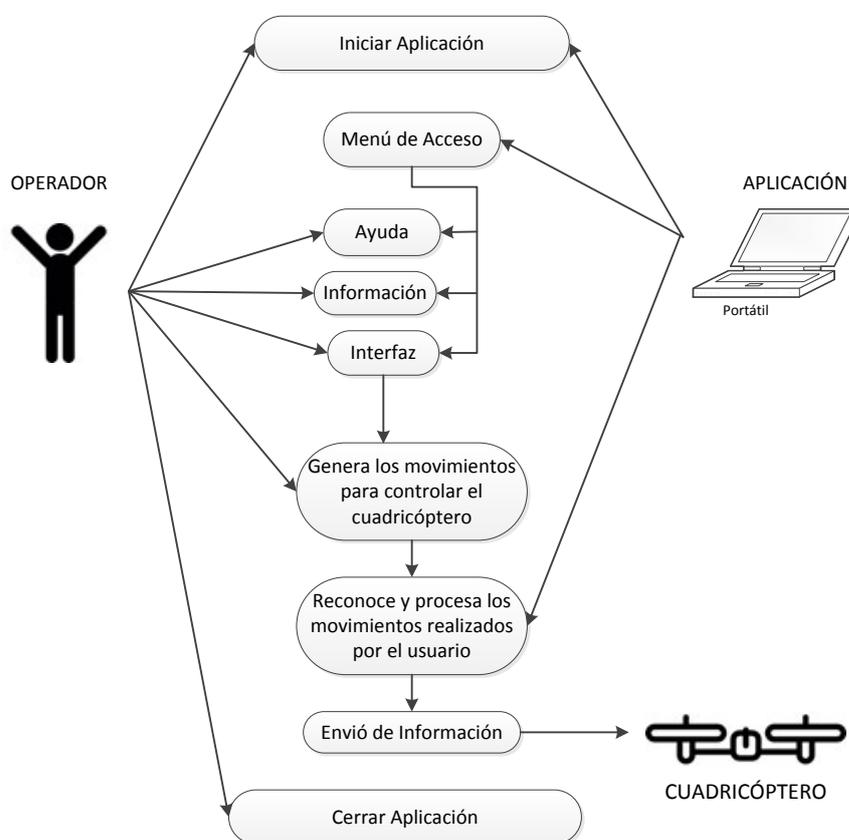
La figura 57 muestra el diagrama de flujo del funcionamiento de la aplicación.



**Figura 57 Diagrama de Flujo de la Aplicación**

### 3.4.5 Diagrama de Casos de uso de la Interfaz Gráfica de Usuario

El modelo de casos de uso es un diagrama que presenta las funciones principales de la aplicación, así como lo quien forma parte de cada uno de los procesos. El diagrama de casos de uso especifica el comportamiento que debe tener cada proceso. La figura 58 es un bosquejo de cómo emplear el sistema. El principal actor dentro de todos los procesos es el operador, ya que se encarga de acceder a la aplicación y generar los movimientos para controlar el cuadricóptero.



**Figura 58 Diagrama de Casos de Uso**

Para explicar de manera clara las consideraciones del diagrama de casos de uso se presentan a continuación una serie de tablas que describen los aspectos más importantes de cada función. Cada tabla está compuesta de un Identificador, Nombre, Actor, Descripción, Precondiciones, Postcondiciones y el flujo que seguirá cada proceso.

**Tabla 22**  
**Modelo de Casos de Uso “Iniciar Aplicación”**

<b>Identificador</b>	MCU_01
<b>Nombre</b>	Iniciar aplicación
<b>Actor</b>	Operador
<b>Descripción</b>	El usuario ejecuta la aplicación
<b>Precondiciones</b>	Abrir la aplicación Tener conectado el Kinect
<b>Flujo Normal</b>	El usuario ejecuta la aplicación El sistema inicia la aplicación
<b>Postcondiciones</b>	Se carga la aplicación
<b>Requerimientos</b>	RF01

**Tabla 23**  
**Modelo de Casos de Uso “Menú de Acceso”**

<b>Identificador</b>	MCU_02
<b>Nombre</b>	Menú de Acceso
<b>Actor</b>	Operador
<b>Descripción</b>	Permite visualizar el menú de acceso
<b>Precondiciones</b>	Abrir la aplicación Tener conectado el Kinect
<b>Flujo Normal</b>	El usuario ejecuta la aplicación El sistema presenta el menú en pantalla
<b>Postcondiciones</b>	Se presenta la opción seleccionada
<b>Requerimientos</b>	RF02

**Tabla 24**  
**Modelo de Casos de Uso “Ayuda”**

<b>Identificador</b>	MCU_03
<b>Nombre</b>	Ayuda
<b>Actor</b>	Operador
<b>Descripción</b>	Permite obtener ayuda del funcionamiento de la aplicación
<b>Precondiciones</b>	Debe mostrar el entorno de la opción
<b>Flujo Normal</b>	El usuario elige la opción El sistema presenta el entorno de la opción en la pantalla
<b>Flujo Alternativo</b>	El usuario elige la otra opción El sistema presenta el entorno de la opción en la pantalla
<b>Postcondiciones</b>	Salir de la opción
<b>Requerimientos</b>	RF02

**Tabla 25**  
**Modelo de Casos de Uso “Información”**

<b>Identificador</b>	MCU_04
<b>Nombre</b>	Información
<b>Actor</b>	Operador
<b>Descripción</b>	Permite obtener información de la aplicación (Funcionamiento y Requerimientos)
<b>Precondiciones</b>	Debe mostrar la opción
<b>Flujo Normal</b>	El usuario elige la opción El sistema presenta el entorno de la opción en la pantalla
<b>Flujo Alternativo</b>	El usuario elige la otra opción El sistema presenta el entorno de la opción seleccionada en la pantalla
<b>Postcondiciones</b>	Salir de la opción
<b>Requerimientos</b>	RF03

**Tabla 26**  
**Modelo de Casos de Uso “Interfaz”**

<b>Identificador</b>	MCU_05
<b>Nombre</b>	Interfaz
<b>Actor</b>	Operador
<b>Descripción</b>	Permite ingresar a la interfaz de control del cuadricóptero
<b>Precondiciones</b>	Seleccionar la opción interfaz
<b>Flujo Normal</b>	Se detecta la comunicación serial Se detecta el dispositivo Kinect El usuario Genera los movimientos de control La aplicación recibe y procesa los movimientos realizados por el usuario
<b>Flujo Alternativo</b>	No se detecta la comunicación serial y aparece un mensaje de error No se detecta el dispositivo Kinect y se cierra la aplicación
<b>Postcondiciones</b>	Genera los movimientos por parte del usuario
<b>Requerimientos</b>	RF04,RF05

**Tabla 27**  
**Modelo de Casos de Uso “Genera los movimientos para controlar el cuadricóptero”**

<b>Identificador</b>	MCU_06
<b>Nombre</b>	Genera los movimientos para controlar el cuadricóptero
<b>Actor</b>	Operador
<b>Descripción</b>	Recibe los datos que el usuario realiza con la manos para controlar el cuadricóptero
<b>Precondiciones</b>	Seleccionar la opción interfaz
<b>Flujo Normal</b>	El usuario utiliza su mano izquierda para generar el movimiento de control vertical y su mano derecha para generar el movimiento de control horizontal y de profundidad
<b>Flujo Alternativo</b>	No se detecta los movimientos del usuario debido a que no se tiene conectado el Kinect o abierto el puerto de comunicación serial
<b>Postcondiciones</b>	Validar el Movimiento
<b>Requerimientos</b>	RF04,RF05

**Tabla 28**  
**Modelo de Casos de Uso “Reconoce y procesa los movimientos realizados por el usuario”**

<b>Identificador</b>	MCU_07
<b>Nombre</b>	Reconoce y procesa los movimientos realizados por el usuario
<b>Actor</b>	Aplicación
<b>Descripción</b>	Reconocer y procesar que los movimientos que realizo el usuario para controlar el cuadricóptero sean los adecuados
<b>Precondiciones</b>	El usuario debe generar los movimientos
<b>Flujo Normal</b>	La aplicación reconoce el movimiento para encender, elevar y desplazar el cuadricóptero de forma vertical, horizontal , hacia adelante y hacia atrás
<b>Flujo Alternativo</b>	La aplicación reconoce el movimiento para descender y aterrizar el cuadricóptero
<b>Postcondiciones</b>	Se envían los datos al cuadricóptero a través de la placa de control y el controlador remoto
<b>Requerimientos</b>	RF04, RF05, RF06, RF07, RF08 y RF09

**Tabla 29**  
**Modelo de Casos de Uso “Cerrar Aplicación”**

<b>Identificador</b>	MCU_08
<b>Nombre</b>	Cerrar aplicación
<b>Actor</b>	Operador
<b>Descripción</b>	Permite al usuario cerrar la aplicación
<b>Precondiciones</b>	La aplicación de estar disponible
<b>Flujo Normal</b>	El usuario selecciona el botón para salir de la aplicación
<b>Postcondiciones</b>	Se cierra la aplicación
<b>Requerimientos</b>	RF03

### 3.4.6 Diseño de diagramas de flujo para la aplicación

En esta sección se describe la lógica de programación mediante diagramas de flujo de las clases y métodos usados para la implementación de la interfaz gráfica de usuario. El lenguaje de programación que se escogió para el desarrollo de la aplicación es C#, ya que posee una aplicación denominada Windows Presentation Foundation (WPF). Según estudio realizado por Guacanes (s.f, p. 3), se dice que esta aplicación “puede transferir la mayor carga de trabajo posible a cualquier unidad de procesamiento de gráficos (GPU) disponible en el sistema”.

La aplicación WPF ayuda a que cada imagen dentro del sistema pueda ajustarse a la medida y resolución de la pantalla en la que se grafica. Esto sucede en vista que se emplea el uso de gráficos vectoriales. Además, gracias a su lenguaje de marcado de aplicaciones extensible (XAML), la aplicación permite separar en archivos diferentes la implementación lógica y el diseño de la interfaz, lo que ayuda en el proceso del desarrollo de la aplicación.

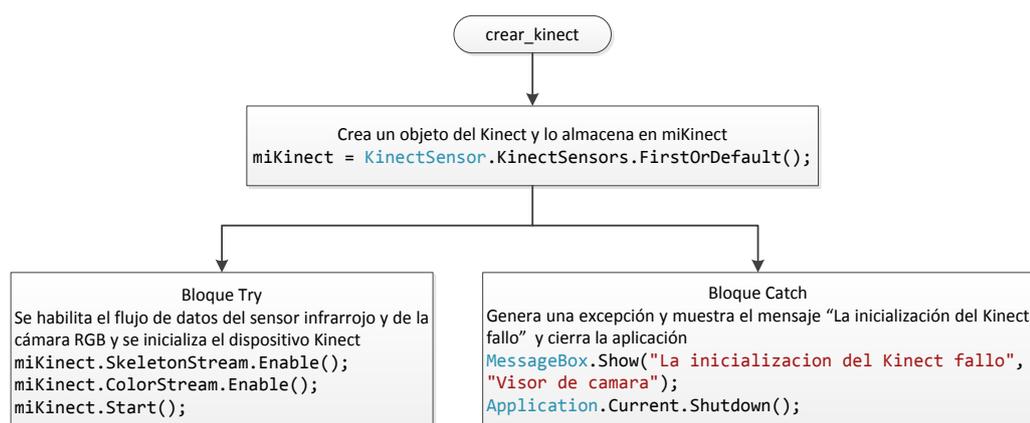
#### 3.4.6.1 Flujo de Datos del Sensor Kinect

Para tener acceso al flujo de datos que proporciona el dispositivo Kinect y poderlo utilizar es necesaria la ayuda de las librerías del SDK. La cámara

del Kinect captura datos de profundidad mientras que el arreglo de micrófonos captura datos de audio. Así, el dispositivo Kinect accede a los datos del esqueleto del usuario mediante la combinación de los valores obtenidos por los sensores de profundidad y el emisor de infrarrojos (IR). El flujo de datos del dispositivo Kinect se activa creando un objeto de la clase *KinectSensor*. Esta clase permite reconocer si el sensor se encuentra conectado y listo para su uso, iniciando la captura de la información proveniente del sensor que se llama al método `Start()`.

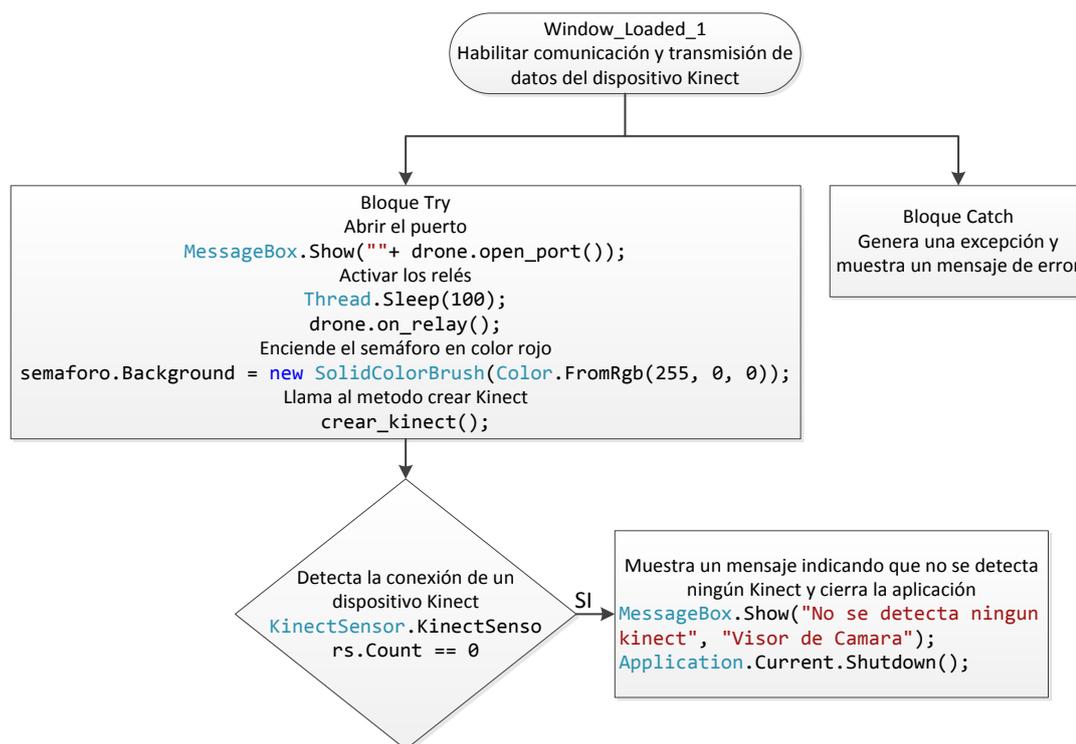
Dentro de la clase *KinectSensor* se puede especificar los diferentes parámetros que se desea obtener, tales como la información de profundidad y del esqueleto, así como se puede definir el número de *frames* por segundo que se desea mostrar. Al tener más de un sensor conectado a la PC se debe especificar cual se va a utilizar. Para establecer que se detecte el primer dispositivo que se encuentra conectado se crea un objeto del Kinect y se lo almacena en la variable `miKinect`. La figura 59 muestra el diagrama de flujo del método `crear_kinect` que sirve para crear el objeto, elegir el primer sensor conectado y habilitar el sensor para la captura de datos.

### Método crear\_kinect



**Figura 59 Diagrama de flujo del método crear\_kinect**

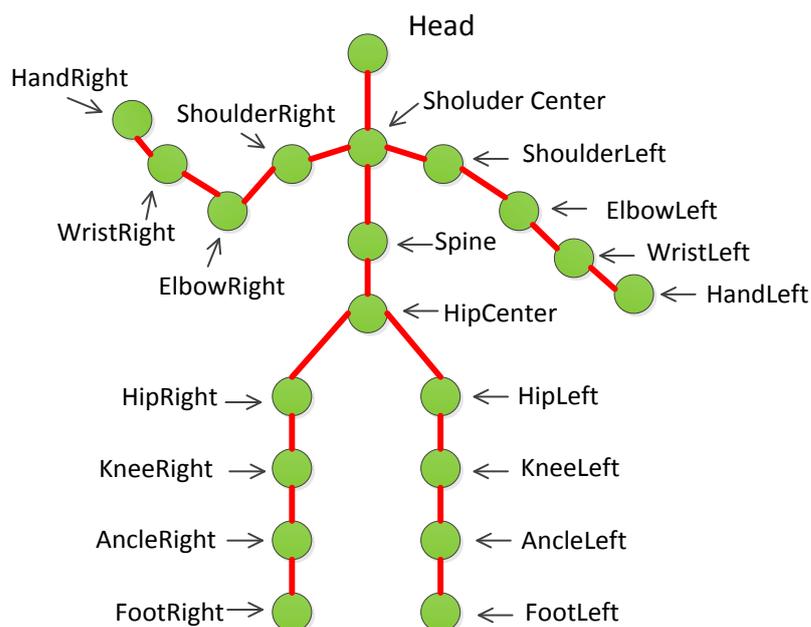
La figura 60 muestra el diagrama de flujo utilizado para iniciar el proceso de la transmisión de datos del dispositivo Kinect.



**Figura 60** Algoritmo utilizado para habilitar el flujo de datos

### 3.4.6.2 Flujo de Datos del Esqueleto

El envío de *frames* constantes con las coordenadas 3D de las articulaciones del esqueleto humano son almacenadas con la función `SkeletonTrackingState`. Esta función permite almacenar en variables de tipo `double` las coordenadas en X, Y, Z. Las articulaciones reconocidas por el dispositivo Kinect son consideradas con su respectivo nombre en inglés como se muestra en la figura 61.



**Figura 61 Nombres de las Articulaciones Reconocidas por el dispositivo Kinect**

El flujo de datos de las articulaciones *Joints* muestra las posiciones de las manos del operador. Éste permite que la aplicación y el usuario puedan trabajar en conjunto a través de la interfaz de control. Las sentencias utilizadas para definir las coordenadas son las que se aprecian en la figura 62

```
// Brazo Izquierdo
agregarLinea(esqueleto.Joints[JointType.ShoulderCenter], esqueleto.Joints[JointType.ShoulderLeft]);
agregarLinea(esqueleto.Joints[JointType.ShoulderLeft], esqueleto.Joints[JointType.ElbowLeft]);
agregarLinea(esqueleto.Joints[JointType.ElbowLeft], esqueleto.Joints[JointType.WristLeft]);
agregarLinea(esqueleto.Joints[JointType.WristLeft], esqueleto.Joints[JointType.HandLeft]);

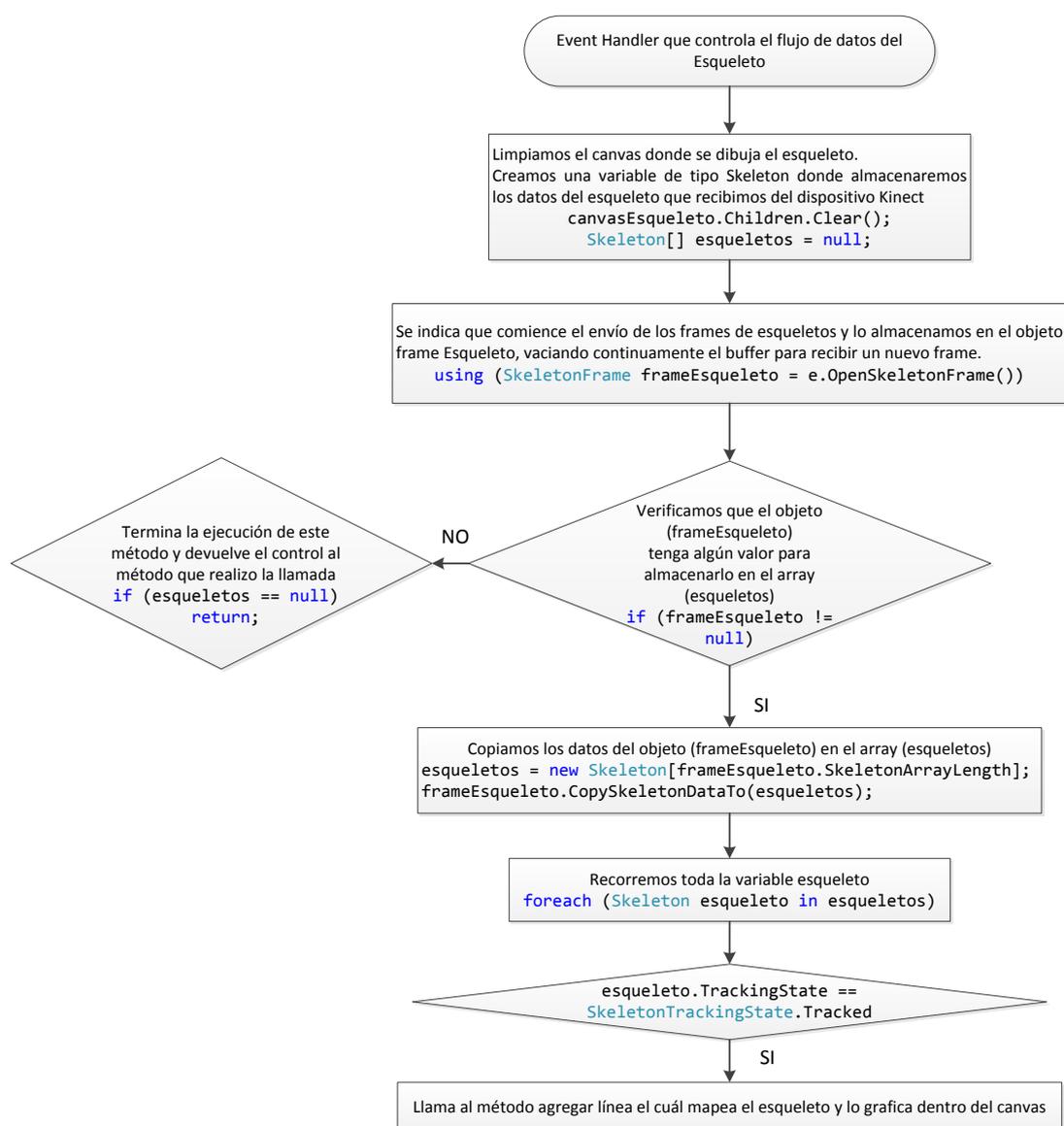
// Brazo Derecho
agregarLinea(esqueleto.Joints[JointType.ShoulderCenter], esqueleto.Joints[JointType.ShoulderRight]);
agregarLinea(esqueleto.Joints[JointType.ShoulderRight], esqueleto.Joints[JointType.ElbowRight]);
agregarLinea(esqueleto.Joints[JointType.ElbowRight], esqueleto.Joints[JointType.WristRight]);
agregarLinea(esqueleto.Joints[JointType.WristRight], esqueleto.Joints[JointType.HandRight]);
```

**Figura 62 Sentencias para definir las coordenadas de las articulaciones**

La tabla 30 muestra las funciones que utiliza el sensor Kinect para realizar el proceso que controlar el flujo de información del esqueleto.

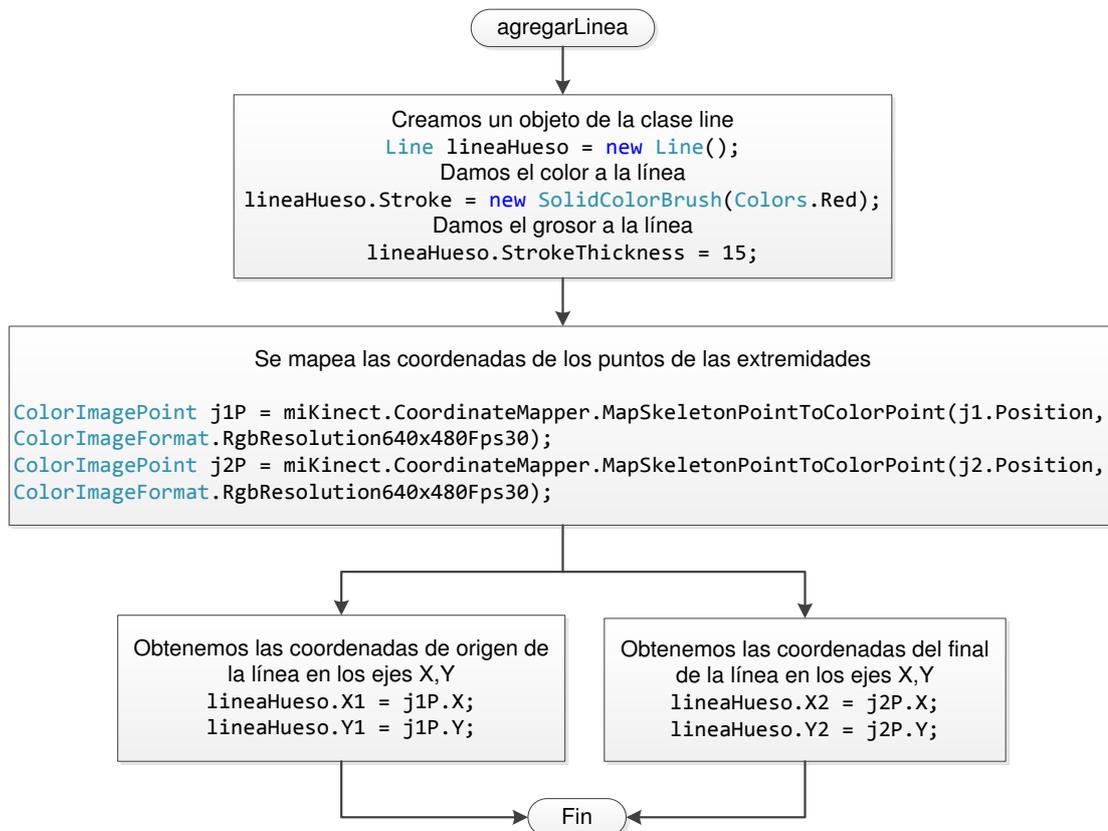
**Tabla 30**  
**Funciones para reconocer el flujo de esqueletos**

Función	Descripción
Skeleton	Provee la información para identificar al usuario y crear el esqueleto
SkeletonFrame	Posee información de la posición de las articulaciones reconocidas por el Kinect. Inicializa la captura del frame de esqueletos.
SkeletonTrackingState	Proporciona los estados de la función Skeleton. Identifica las articulaciones y efectúa el seguimiento del esqueleto



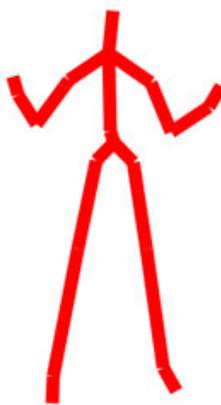
**Figura 63 Diagrama de Flujo del Algoritmo Utilizado para el Control de los Datos de Esqueletos**

## Método agregarLinea



**Figura 64 Diagrama de flujo del método agregarLinea**

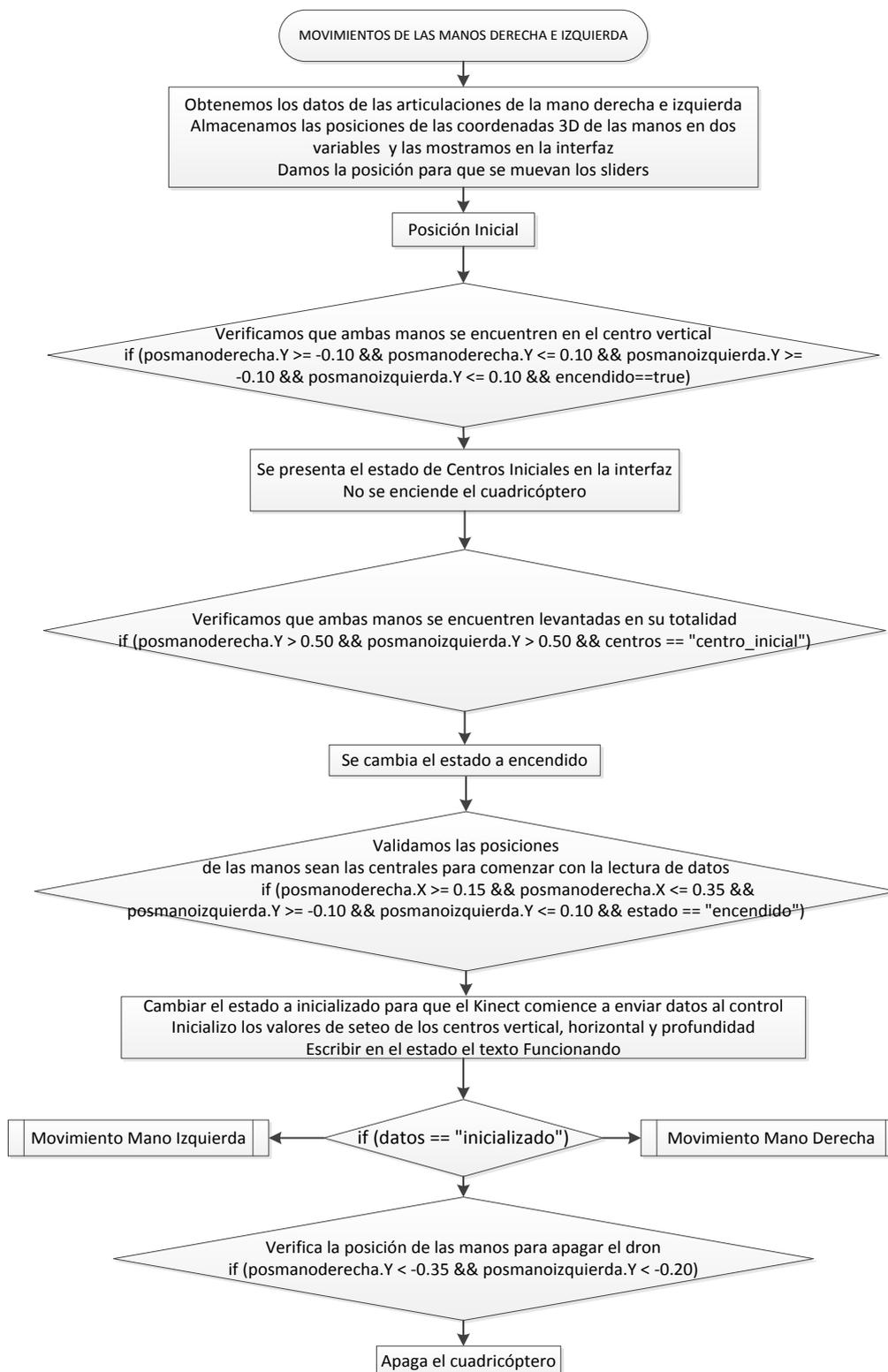
La figura 65 muestra el resultado de la esqueletización de las coordenadas del usuario.



**Figura 65 Imagen esqueletizada**

### 3.4.6.3 Reconocimiento de los Movimientos para controlar el cuadricóptero

La figura 66 muestra el algoritmo utilizado para reconocer los movimientos de las manos para controlar al cuadricóptero.



**Figura 66 Diagrama de flujo para reconocer los movimientos de las manos**

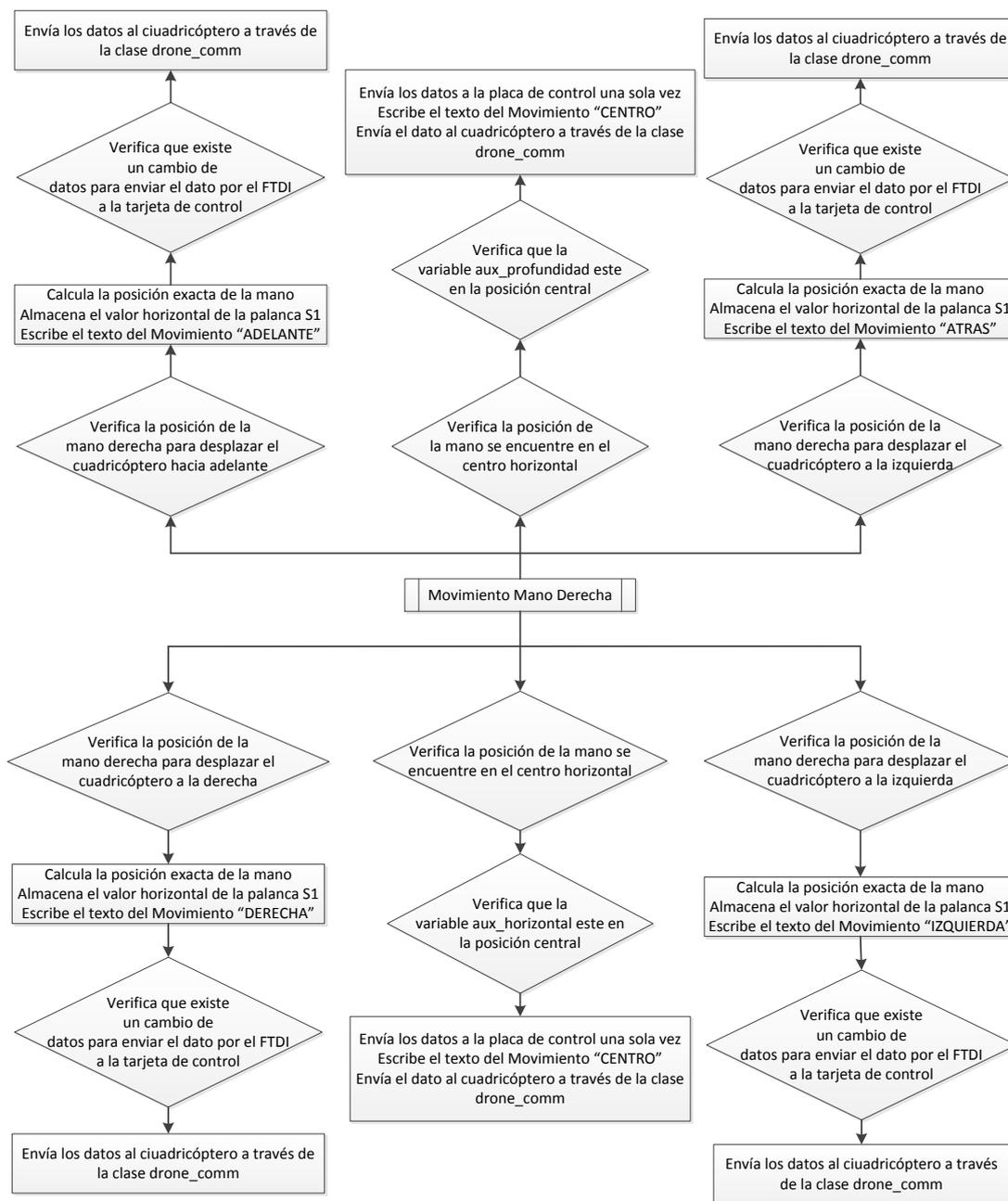


Figura 67 Movimientos mano derecha

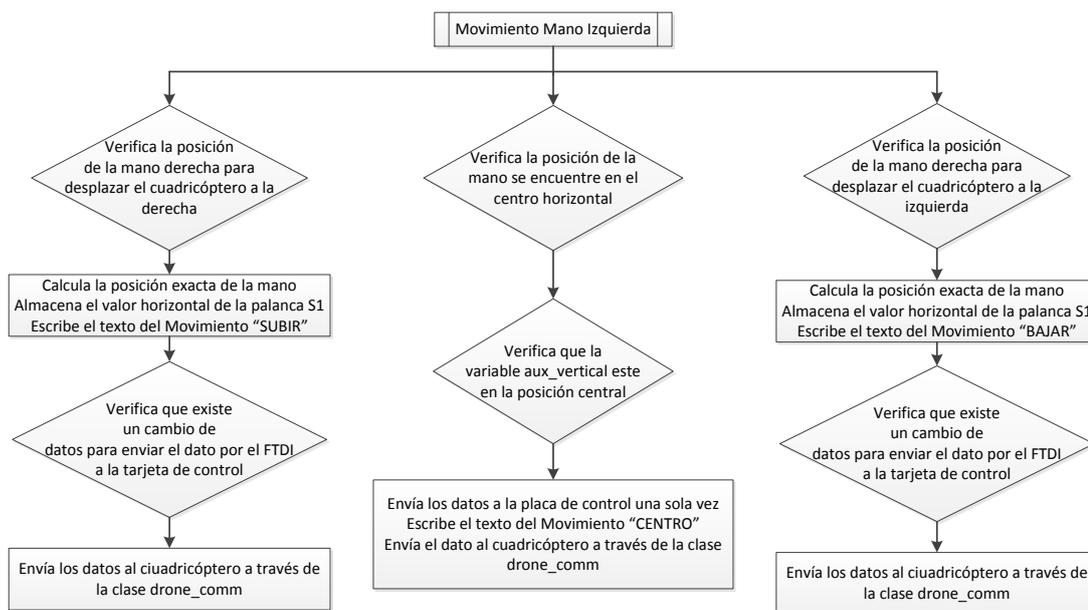


Figura 68 Movimientos mano izquierda

### 3.4.6.4 Clase drone\_comm

La figura 69 muestra el diagrama de flujo de la clase drone\_comm la cual sirve para establecer la comunicación Serial entre la computadora y la placa de control electrónica.

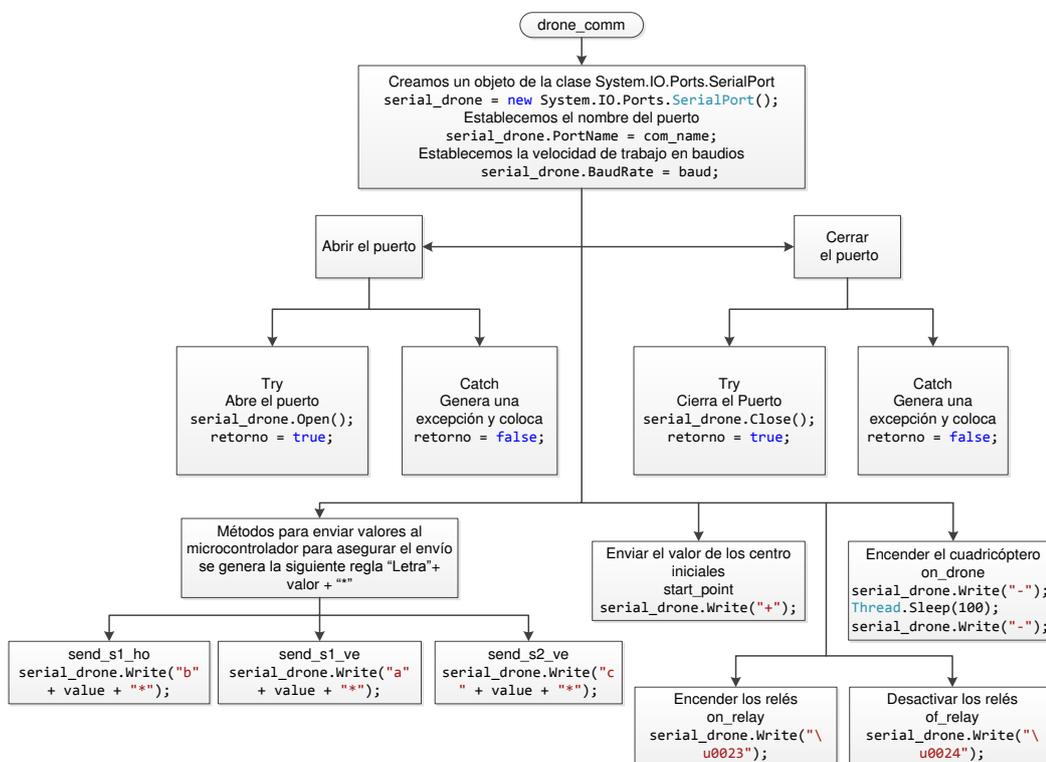
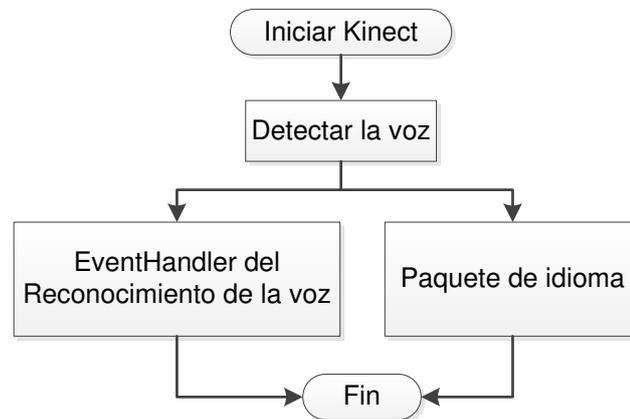


Figura 69 Diagrama de flujo de la Clase drone\_comm

### 3.4.6.5 Reconocimiento de la Voz

Para este proceso, se establece que la voz del usuario puede ser detectada por medio del arreglo de los micrófonos que tiene el dispositivo Kinect. Para ello, se debe usar las librerías de formato y reconocimiento de audio. Para acceder a la clase que permite reconocer la voz del usuario se accede a los métodos que poseen las librerías del SDK y se crea un objeto del tipo `SpeechRecognitionEngine`. La figura 70 muestra el diagrama de flujo del procedimiento que realiza el Kinect para detectar la voz.



**Figura 70 Diagrama de flujo del reconocimiento de voz**

El bloque que realiza el procedimiento para detectar la voz del usuario necesita saber el lenguaje que va a utilizar y las frases que va a detectar, los parámetros que se necesitan para la configuración de este método son:

- Formato de codificación
- Número de muestras por segundo
- Bits por muestra
- Número de canal
- Promedio de bytes por segundo
- Bytes que se van a transmitir

Las sentencias que se utiliza para el reconocimiento de la voz se pueden apreciar en la figura 71.

```
//(formato de Codificación PCM, muestras por segundo, bits por
muestra, numero de canal, promedio Bytes por segundo, alineacion
del bloque, bytes[] formato especifico de los datos)
speechengine.SetInputToAudioStream(_sensor.AudioSource.Start(),
new SpeechAudioFormatInfo(EncodingFormat.Pcm, 16000, 16, 1, 32000,
2, null));
speechengine.RecognizeAsync(RecognizeMode.Multiple);//Sincronizar
el reconocimiento de modo multiple
```

**Figura 71 Sentencias para reconocimiento de voz**

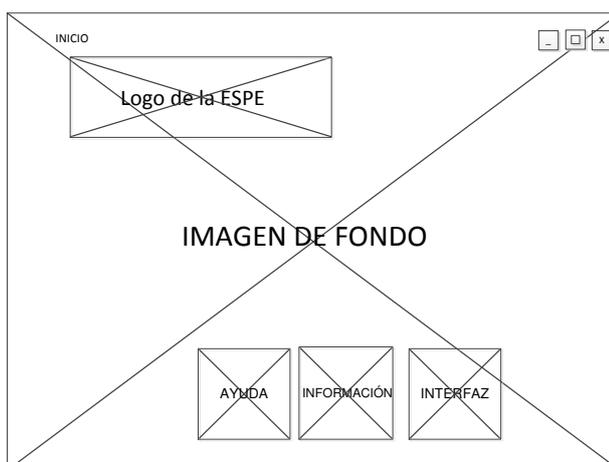
El bloque *EventHandler* nace desde el procedimiento para detectar la voz. Es en esta etapa donde se establece el rango de tolerancia de error para detectar las palabras o frases a utilizar, así como las acciones a ejecutar de acuerdo al comando establecido. El bloque que corresponde al paquete de idioma es donde se configura el lenguaje que se va a utilizar para el reconocimiento de la voz.

### 3.5 DISEÑO DE LA INTERFAZ GRÁFICA DE USUARIO (GUI)

El diseño de las interfaces que conforman la aplicación se llevó a cabo teniendo en cuenta que deben ser amigables, de fácil comprensión, intuitivas y coherentes con la información que presentan al usuario.

#### 3.5.1 Diseño de la Ventana “Menú de Acceso”

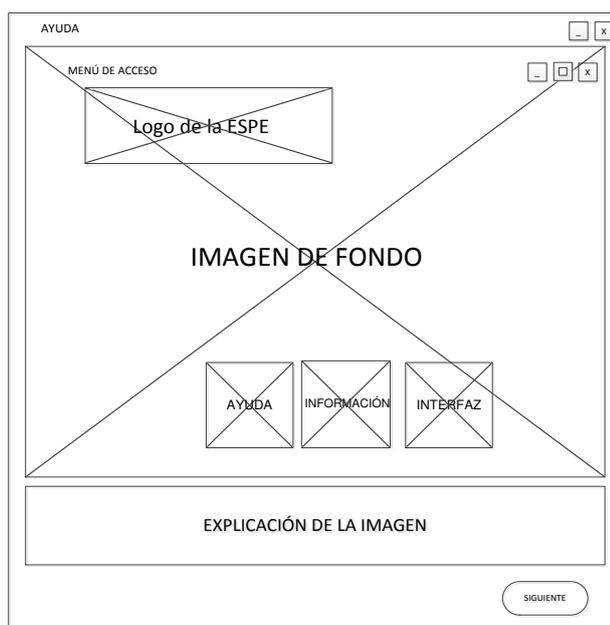
La ventana se presenta al iniciar la aplicación. Según se observa en la Figura 72, el diseño presenta tres botones para que el usuario pueda identificar cómo manejar la aplicación, presentar la información del sistema o ejecutar la interfaz. Cada botón esta personalizado con una imagen para hacer más intuitiva a la interfaz.

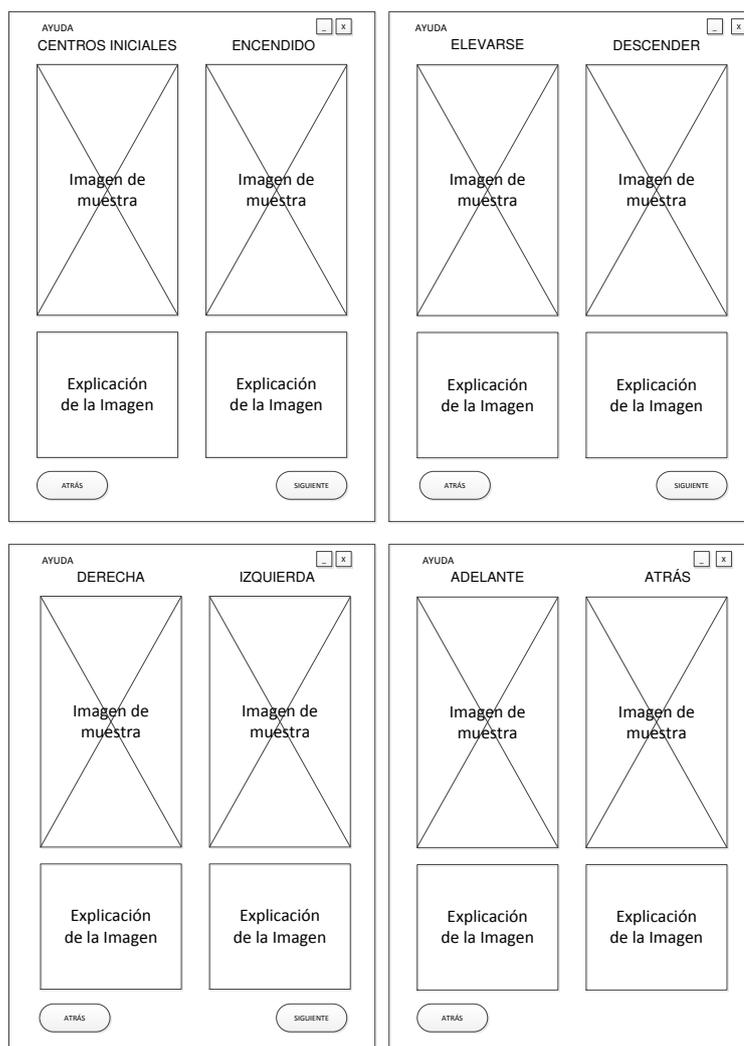


**Figura 72 Diseño de la ventana “Menú de Acceso”**

### 3.5.2 Diseño de las Ventanas de “Ayuda”

Presenta un corto manual para que el usuario pueda utilizar la aplicación y sepa cómo realizar los movimientos para controlar el cuadricóptero. Esta parte de la aplicación consta de varias ventanas las cuales poseen dos botones para desplazarse a la siguiente o regresar a la anterior, dentro de cada ventana se presentan imágenes con su respectiva explicación para hacer a la interfaz más amigable con el usuario.

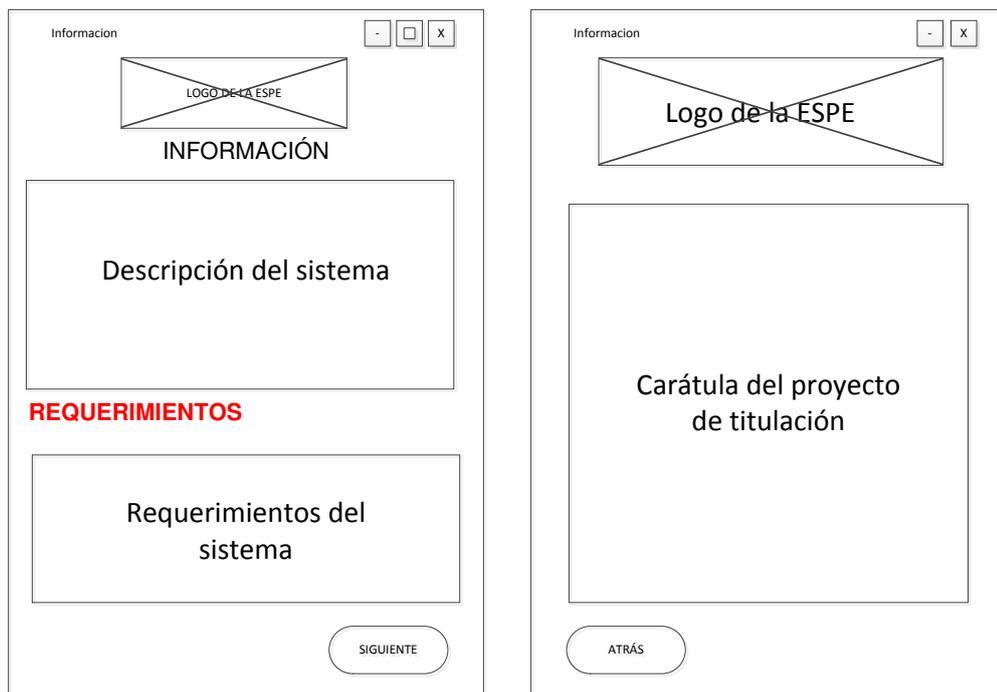




**Figura 73 Diseño de las ventanas de “Ayuda”**

### 3.5.3 Diseño de las Ventanas de “Información”

Ésta presenta una breve descripción del sistema, así como sus requerimientos. Esta parte de la aplicación consta de dos ventanas. La primera ventana muestra la información del sistema y posee dos botones. El primer botón sirve para regresar al Menú de Acceso. El otro sirve para acceder a la segunda ventana donde se muestra información del proyecto de titulación.



**Figura 74 Diseño de las ventanas de “Información”**

### 3.5.4 Diseño de la Ventana “Interfaz”

Éste presenta un área donde se visualiza la imagen esqueletizada del usuario. Está compuesto de tres sliders, los cuales sirven para controlar el movimiento vertical, horizontal y de profundidad de la aeronave. La ventana muestra información del estado del cuadricóptero, así como de las coordenadas de las manos derecha e izquierda en los ejes X, Y, Z. Además, éste muestra las posiciones que toma cada slider de acuerdo al movimiento del usuario. Los datos que son enviados a la placa de control electrónica.

Este diseño posee tres botones. El primero sirve para abrir el puerto serial. El segundo permite activar los relés. Finalmente, el tercer botón permite desactivar los relés. Pese a que la interfaz realiza automáticamente los procedimientos, se colocan los botones en el caso de alguna falla del sistema. Adicional a esto, el sistema presenta un texblock que cambia de color cuando reconoce un comando de voz.



**Tabla 31**  
**Controladores de las ventanas de la aplicación**

Control	Descripción
Botón	Realiza una acción al hacer click sobre él
Label	Mostrar texto en la ventana
TextBox	Permite ingresar datos y mostrar texto
Image	Presentar imágenes
Canvas	Lienzo para dibujar puntos, líneas, elipses, etc
Slider	Obtiene un valor numérico al arrastrar el cursor
Grid	Área cuadriculada flexible formada por filas y columnas
TexBlock	Mostrar pequeñas cantidades de contenido dinámico

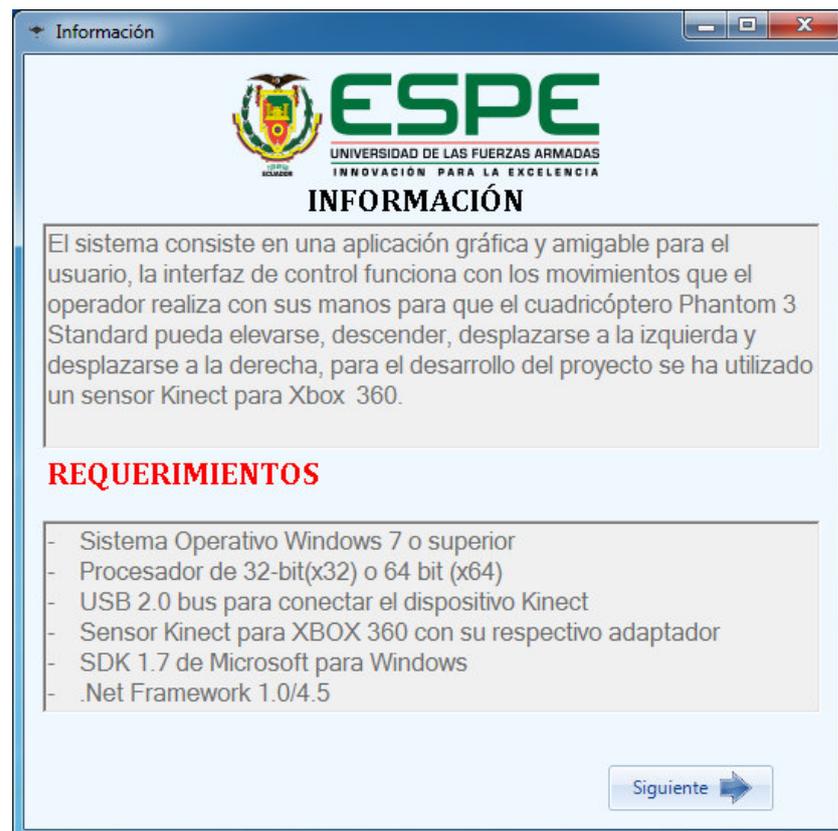
Las ventanas de la interfaz implementadas se muestran en las siguientes figuras:



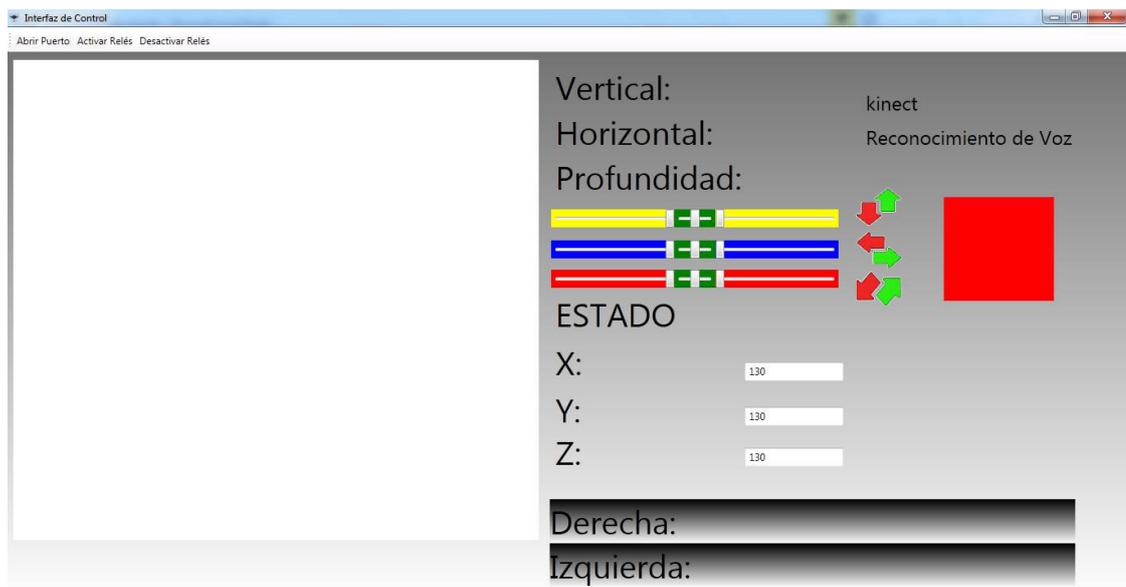
**Figura 76 Ventana “Menú de Acceso”**



Figura 77 Ventanas de "Ayuda"



**Figura 78 Ventanas de “Información”**



**Figura 79 Ventana “Interfaz”**

### 3.5.6 Elementos del sistema para la interfaz de control

A continuación se presenta los elementos y recursos que se utilizó para la implementación del sistema.

#### 3.5.6.1 Recursos y Herramientas

Para la implementación del sistema se utilizó los siguientes recursos y Herramientas:

#### Software

- Sistema Operativo Windows 7
- Kit de Desarrollo de Software (SDK) para Kinect version 1.7
- Visual Studio 2013 (Microsoft .NET Framework 4.5)
- Lenguaje de programación C Sharp (C#)
- Proteus Design Suite 8.0
- Aplicación DJI GO

## Hardware

- Computadora Toshiba Satellite con un procesador Core i3 2.4 GHz, memoria RAM de 2 GB.
- Dispositivo Kinect para XBOX 360
- Placa de control electrónica
- Smartphone
- Cuadricóptero Phantom 3 Standard con su controlador remoto



**Figura 80 Elementos del sistema**

## CAPÍTULO IV

### PRUEBAS DE FUNCIONAMIENTO

El presente capítulo está enfocado en realizar pruebas del funcionamiento de la interfaz de control, para verificar que la aplicación funciona sin ningún problema se realiza pruebas de la distancia y altura a la cual el dispositivo Kinect reconoce los movimientos realizados por el usuario, se prueba la funcionalidad que ofrece la interfaz y se evalúa el manejo de la aplicación por parte de varios usuarios dentro de un ambiente controlado.

#### 4.1 PRUEBAS DE FUNCIONAMIENTO DEL DISPOSITIVO KINECT

En estas pruebas se validan las alturas apropiadas para el funcionamiento adecuado de la interfaz de control y la distancia a la cual debería estar ubicado el usuario con relación al dispositivo Kinect para que el reconocimiento de las manos sea captado sin ningún problema.

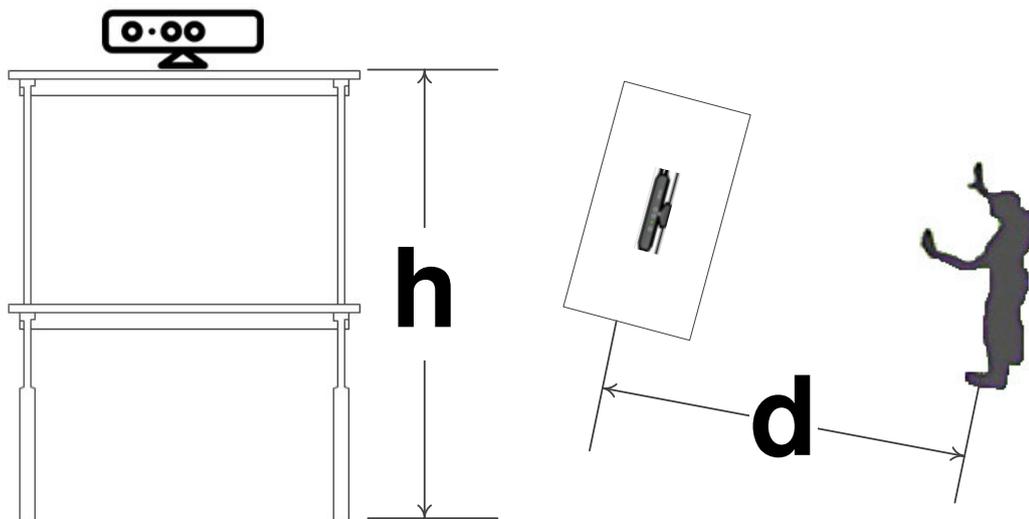


Figura 81 Pruebas de altura y distancia

**Tabla 32**  
**Pruebas de altura y distancia**

<b>PRUEBA</b>	<b>DISTANCIA (m)</b>	<b>ALTURA (m)</b>	<b>OBSERVACION</b>
1	1,53	1,24	Mano Izquierda a la altura del hombro Mano Derecha a la altura del Pecho
2	1,52	1,21	Mano Izquierda a la altura del hombro Mano Derecha a la altura del hombro
3	1,53	1,18	Mano Izquierda a la altura del Pecho Mano Derecha centro inicial
4	1,52	1,15	Mano Izquierda muy cerca al centro inicial Mano Derecha centro inicial
5	1,52	1,12	Mano Izquierda centro inicial Mano Derecha centro inicial
6	1,56	1,09	Mano Izquierda centro inicial Mano Derecha centro inicial

De acuerdo a las pruebas realizadas a diferentes alturas se comprobó que la altura idónea para ubicar al dispositivo Kinect es de 1,12m desde el nivel del suelo, en esta posición el usuario debe estar parado frente del dispositivo a una distancia de 1,52m.

## **4.2 PRUEBAS DE FUNCIONAMIENTO DE LA APLICACIÓN**

En estas pruebas se valida que todas las ventanas de la interfaz de control cumplan con los diseños del capítulo anterior y que funcionen de manera correcta.

### **2.1.6 Ventana Menú de Acceso**

Esta prueba consiste en verificar el funcionamiento de los botones Ayuda, Información e Interfaz que se presentan al ejecutar la aplicación.

## Prueba del Botón Ayuda

La figura 82 muestra el funcionamiento de las ventanas del botón Ayuda se puede abrir varias ventanas de acuerdo a la necesidad del usuario.

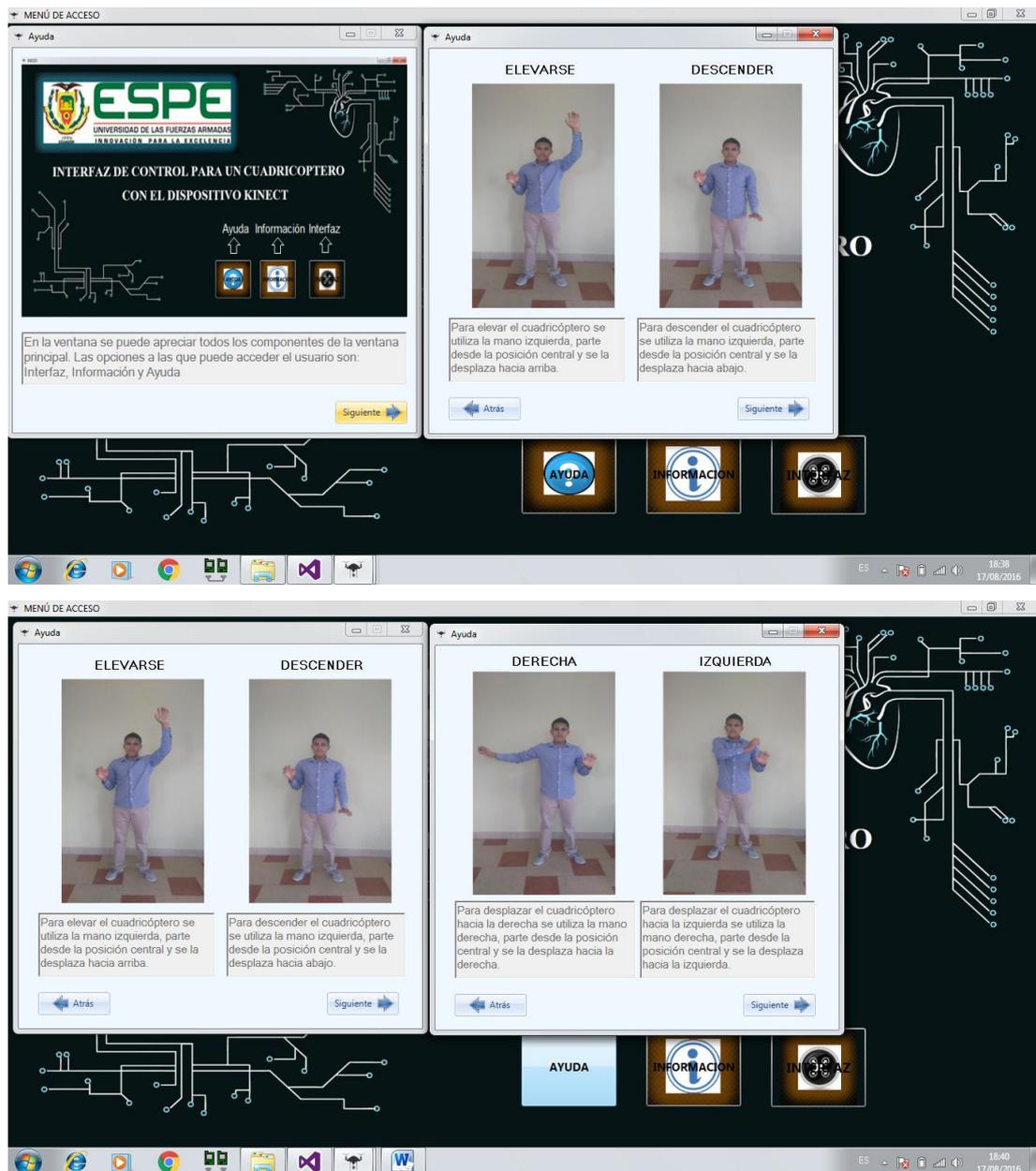
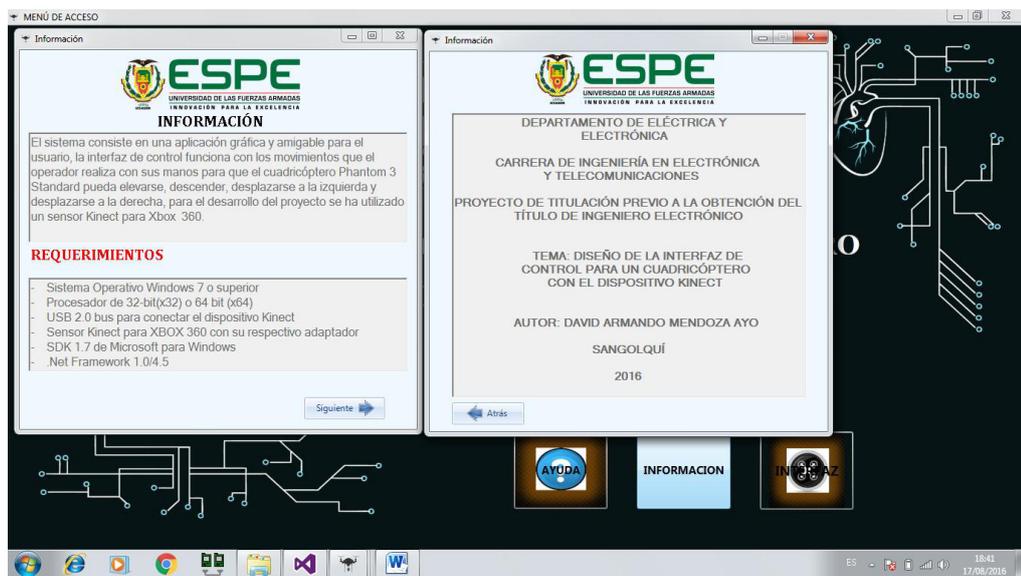


Figura 82 Pruebas botón Ayuda

## Prueba del Botón Información

La figura 83 muestra el funcionamiento de las ventanas del botón Información y se verifica que funcionan sin ningún problema.



**Figura 83 Pruebas botón Información**

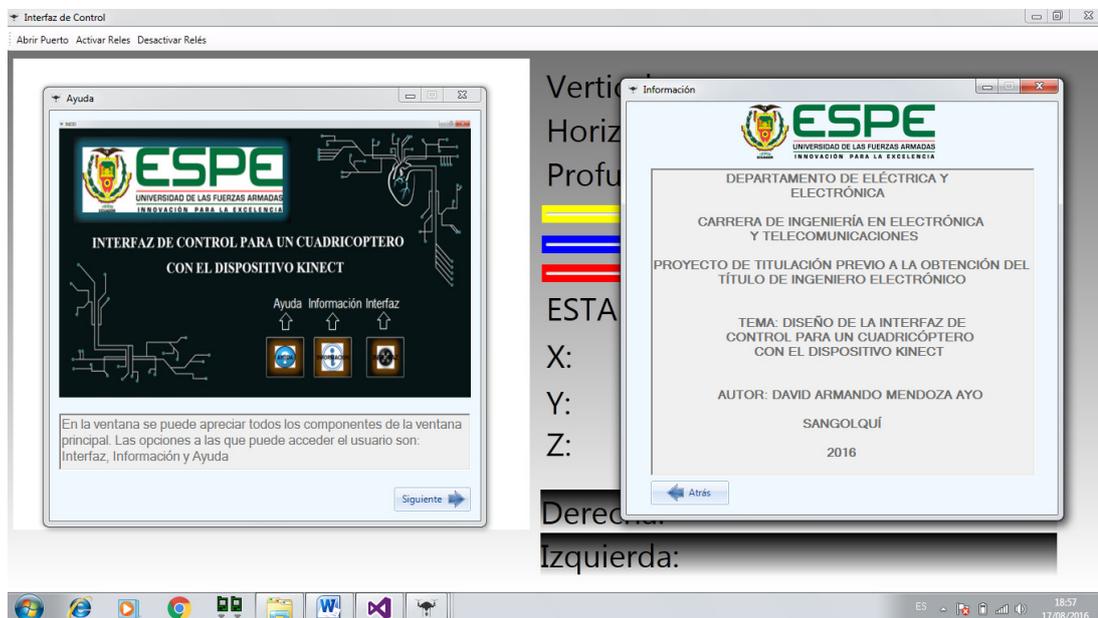
### Prueba del Botón Interfaz

La figura 84 muestra el funcionamiento de la ventana del botón Interfaz y se verifica que funciona sin ningún inconveniente.



**Figura 84 Pruebas del botón Interfaz**

Se puede abrir las ventanas de ayuda e información al mismo tiempo que la ventana de interfaz.



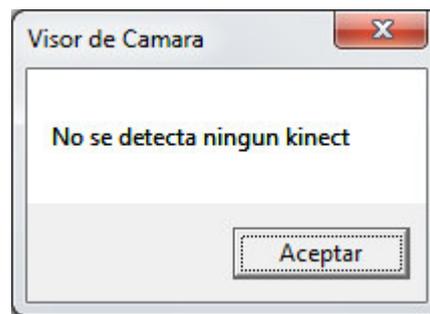
**Figura 85 Pruebas de los botones de la ventana menú acceso**

**Tabla 33  
Pruebas de los botones de la aplicación**

BOTÓN	FUNCIONAMIENTO	OBSERVACIÓN
Ayuda	OK 	Se pueden abrir varias ventanas al mismo tiempo
Información	OK 	Se pueden abrir varias ventanas al mismo tiempo
Interfaz	OK 	Ninguna

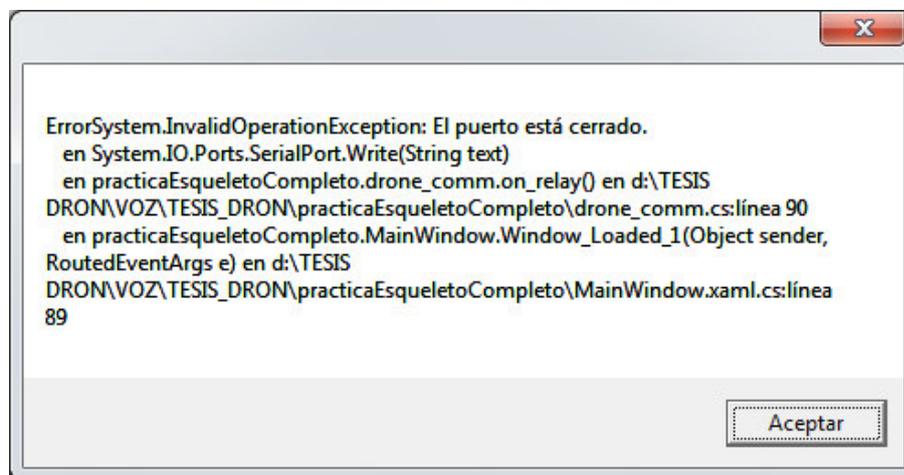
#### 4.2.1 Prueba de la interfaz sin conexión del dispositivo Kinect

Esta prueba sirve para validar si se ha detectado o no el dispositivo Kinect, si el dispositivo no se encuentra conectado se presenta el mensaje de la figura 86



**Figura 86 Prueba de detección del dispositivo Kinect**

De acuerdo a la programación la interfaz genera una excepción y cierra la aplicación.



**Figura 87 Prueba de excepción dispositivo Kinect**

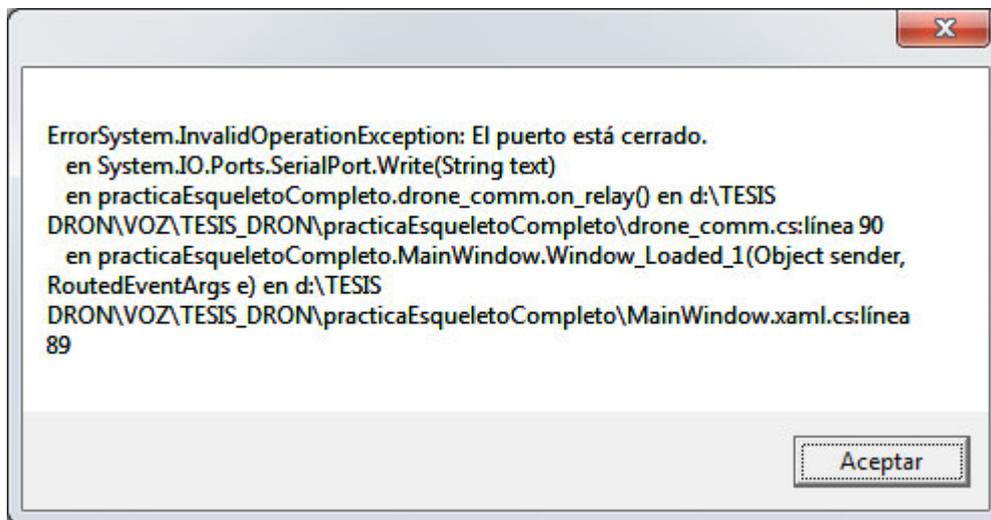
### 2.1.7 Prueba de la interfaz sin conexión de la placa de control

Una vez que se conecta el dispositivo Kinect es necesario conectar la placa de control en caso de que no se conecte se muestra el mensaje de la figura 88



**Figura 88 Prueba de detección de la placa de control**

De acuerdo a la programación se genera una excepción y se despliega la interfaz.



**Figura 89 Prueba de excepción placa de control**

El usuario debe conectar el cable USB de la placa de control al computador y dar click en el botón abrir el puerto para que la comunicación serial pueda funcionar.



**Figura 90 Prueba Interfaz sin conexión placa de control**

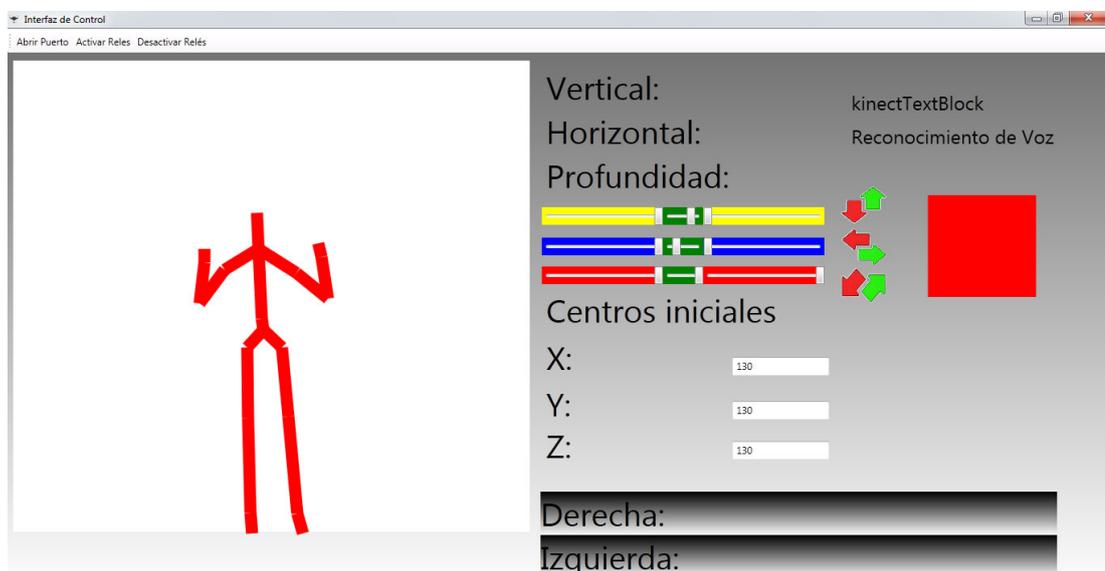
### 2.1.8 Prueba de la interfaz conectado el Kinect y la placa de control

En esta prueba se tiene conectado ambos elementos antes del inicio de la aplicación y como se puede ver en la figura 91 se presenta el mensaje “True” el cuál informa que el puerto serial se abrió con éxito y se reconoció satisfactoriamente al dispositivo Kinect.



**Figura 91 Prueba de reconocimiento del Kinect y la placa de control**

La figura 92 muestra la ventana de la interfaz, la cual se encuentra funcionando con el dispositivo Kinect y la placa de control conectados al computador



**Figura 92 Prueba de la interfaz conectado el Kinect y la placa de control**

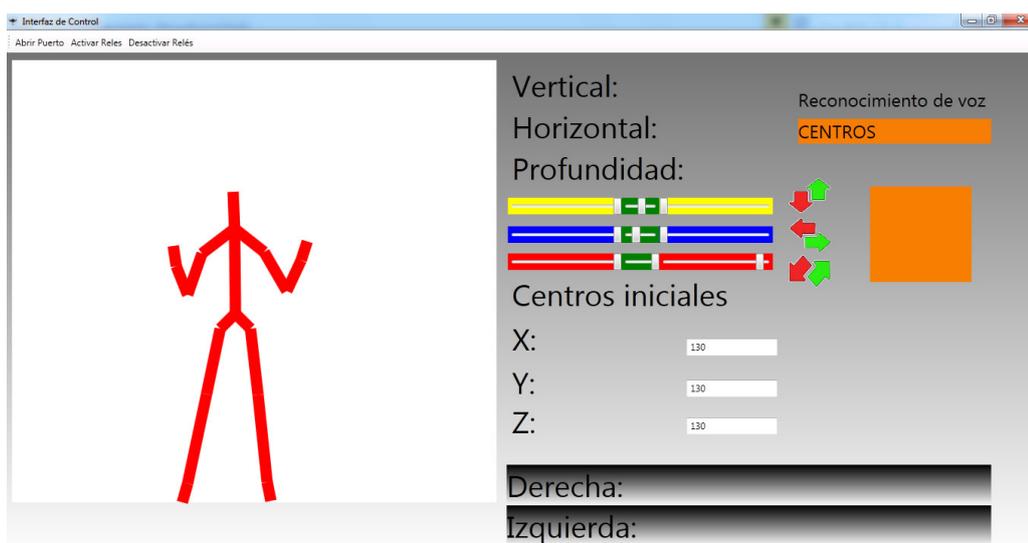
La figura 93 muestra la ventana de la interfaz cuando el usuario se encuentra ubicado correctamente como se vio en la primera parte de este

capítulo, el usuario coloca las manos en la posición central para activar el reconocimiento de la voz.



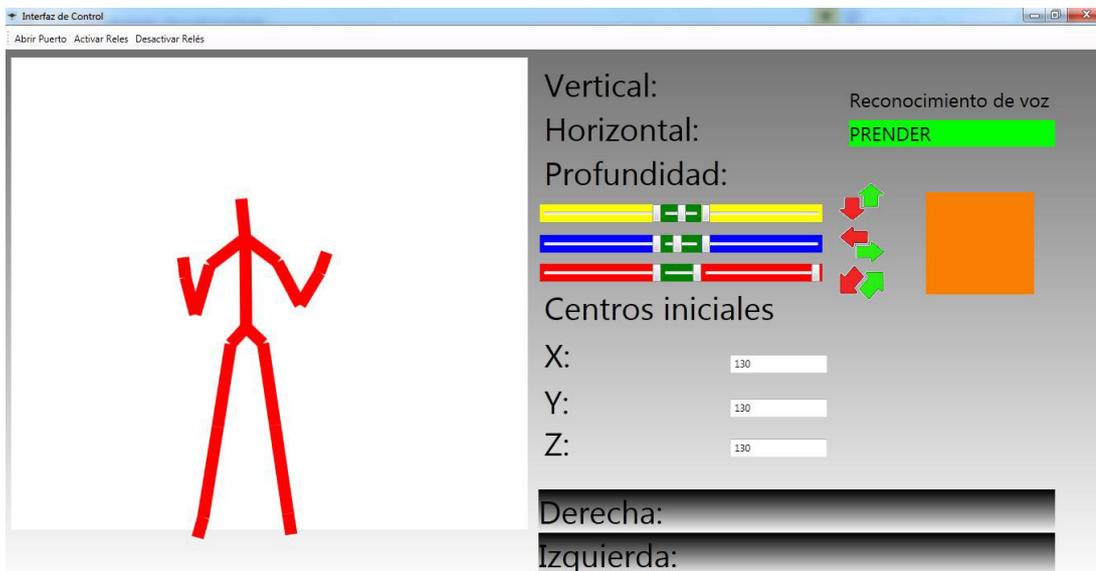
**Figura 93 Prueba de la interfaz con el usuario**

La figura 94 muestra la ventana de la interfaz en la cual el usuario activa el textblock a través del comando de voz CENTROS y el color de este cambia a tomate.



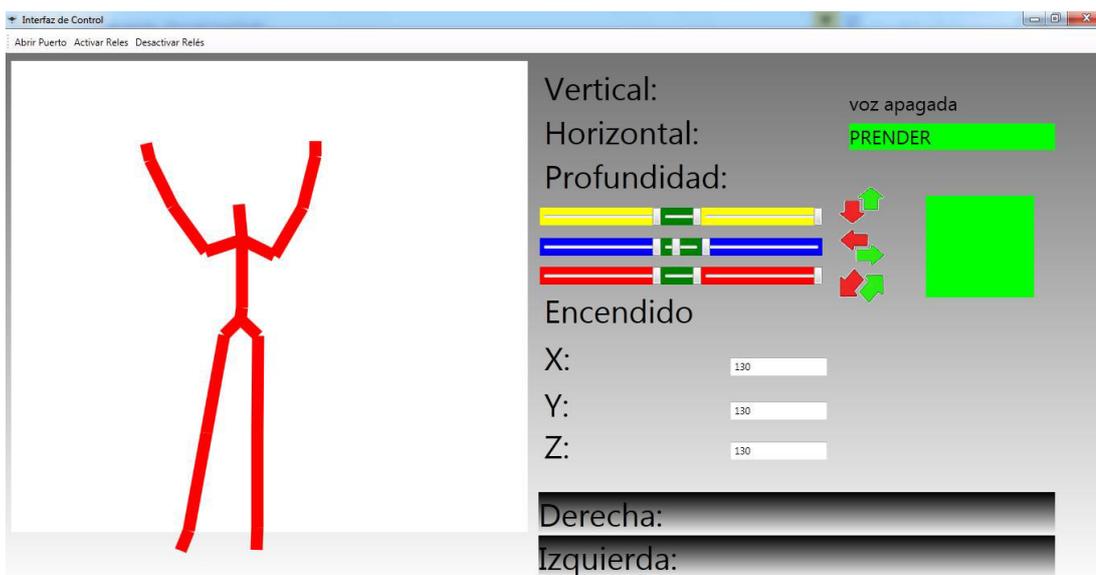
**Figura 94 Prueba de la interfaz comando de voz CENTROS**

La figura 95 muestra la ventana de la interfaz en la cual el usuario activa el comando de voz PRENDER pero el texblock no cambia de color debido a que no se tiene ambas manos levantadas, el cuadricóptero continua apagado.



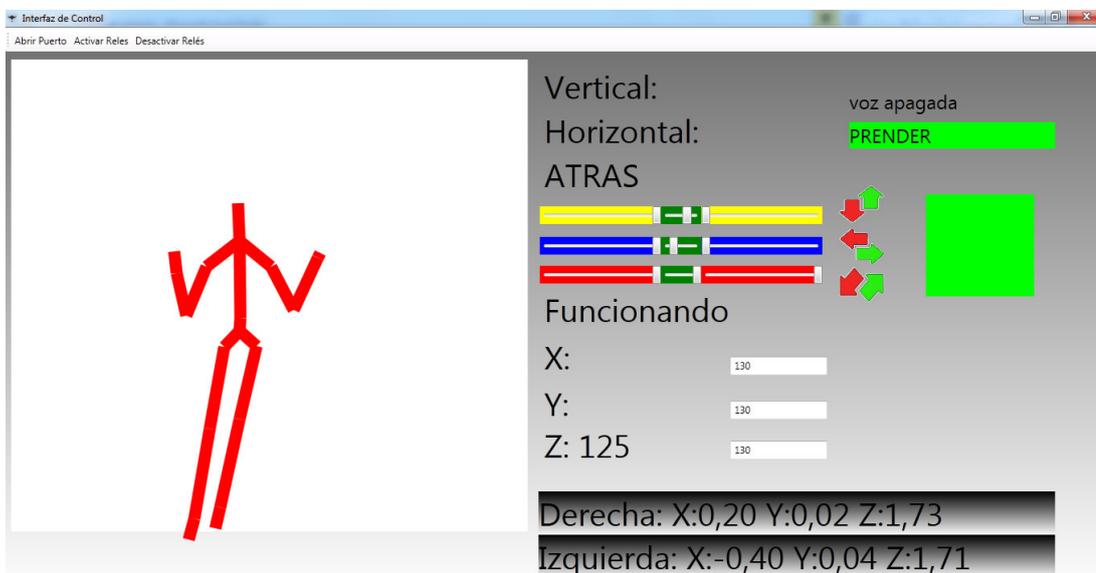
**Figura 95 Prueba de la interfaz comando de voz PRENDER**

La figura 96 muestra la ventana de la interfaz en la cual el usuario levanta las manos para encender al cuadricóptero, el estado de la aeronave cambia a encendido.



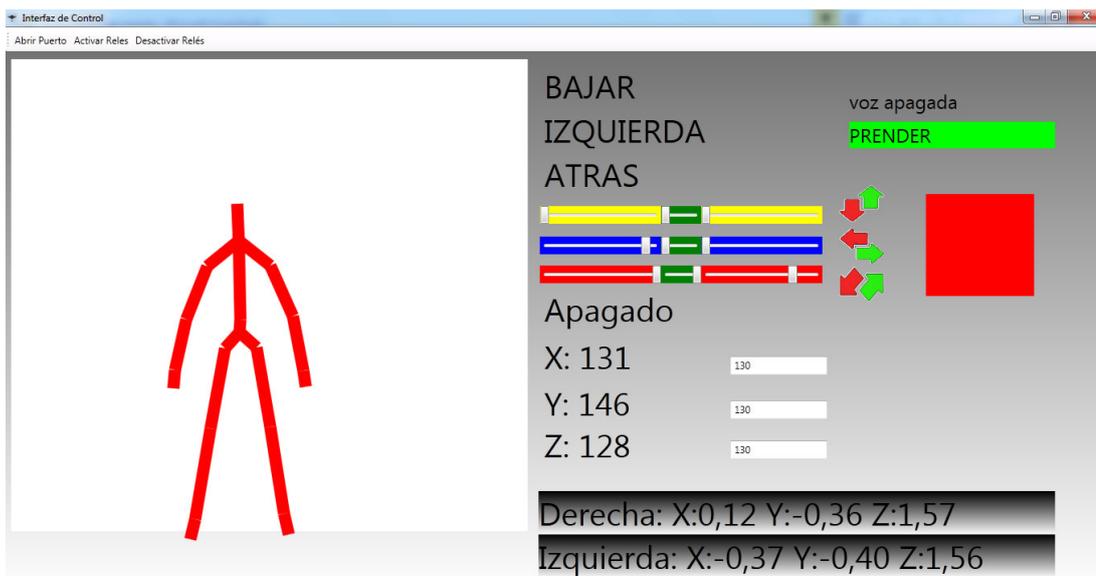
**Figura 96 Prueba de la interfaz en estado Encendido**

La figura 97 muestra la ventana de la interfaz cuando el usuario regresa las manos a la posición central, el sistema entra en funcionamiento y se puede utilizar las manos para controlar al cuadricóptero.



**Figura 97 Prueba de la interfaz en estado Funcionando**

La figura 98 muestra la ventana de la interfaz cuando el usuario procede a aterrizar la aeronave y apagar el sistema.

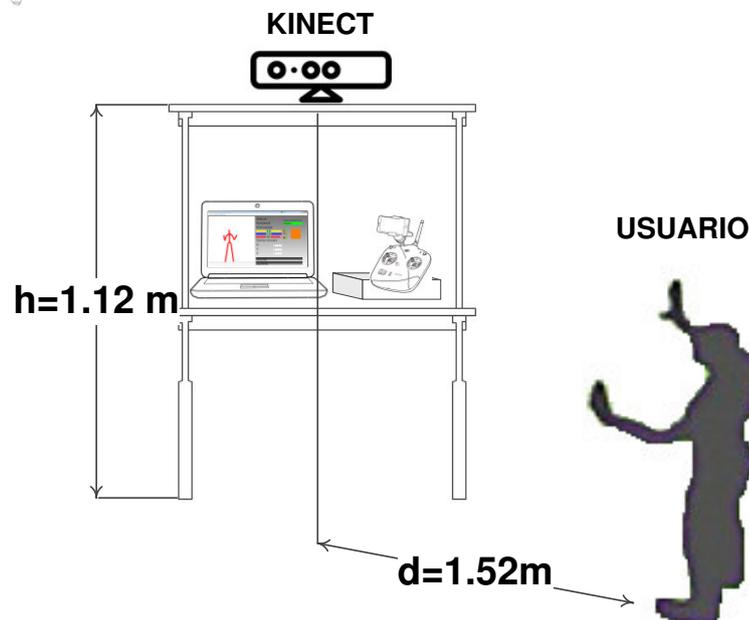


**Figura 98 Prueba de la interfaz en estado Apagado**

### 4.3 PRUEBAS DE COMUNICACIÓN CON EL CUADRICÓPTERO

Con estas pruebas se evalúa el control del cuadricóptero mediante la interfaz de control.

#### CUADRICÓPTERO



**Figura 99 Escenario de la prueba**

En el escenario planteado en la figura 99 el dispositivo Kinect está ubicado a una altura de 1.12m desde el nivel del piso con un ángulo de elevación de  $0^\circ$  y el usuario está ubicado frente al dispositivo Kinect a una distancia de 1.52m, con esto se tendrá una mejor maniobrabilidad en los movimientos para controlar al cuadricóptero.

La prueba fue realizada en un ambiente controlado es decir donde no se tenía exposición a ninguna interferencia externa que afecte al dispositivo Kinect.

Las pruebas de comunicación de la interfaz con el cuadricóptero se realizaron con 6 personas con el objetivo de validar el funcionamiento de la aplicación con diversos usuarios, se tomó en cuenta el número de intentos que el usuario realizó para controlar satisfactoriamente a la aeronave así como el tiempo que le tomó en realizar este proceso.

**Tabla 34**  
**Pruebas de funcionamiento Interfaz con diferentes usuarios**

Nº del Usuario	Nº Intentos	Duración (min)
1	3	2,58
2	4	3,05
3	6	4,26
4	3	2,33
5	8	6,14
6	9	7,02

#### 4.3.1 Escenario de Evaluación

Se ha considerado evaluar al sistema dentro de un espacio controlado con el objetivo de que el dispositivo Kinect funcione sin ningún inconveniente. La figura 100 muestra las dimensiones del espacio utilizado para las pruebas de funcionamiento del sistema.



**Figura 100 Dimensiones del Espacio controlado**

Se utilizó una carpa de 2.5m de largo X 2.5m de ancho y con una altura de 2.7m, con el objetivo de probar el funcionamiento de la interfaz durante cualquier hora del día se procedió a colocar paredes en cada lado de la carpa para no tener interferencias en las cámaras del dispositivo Kinect.

#### 4.3.2 Evaluación del sistema

Para verificar el funcionamiento y la efectividad del sistema se evaluó a la interfaz gráfica de usuario en base a una encuesta que considera la heurística de NIELSEN la cual contempla: Utilización de un lenguaje entendible para el Usuario (1), Control y autonomía para el usuario (2), Visibilidad del estado de la interfaz (3), Estándares y consistencia (4), Minimización de la carga cognitiva (5), Prevención de errores(6), Diseño Minimalista (7), Ayuda y Documentación para el usuario(8).

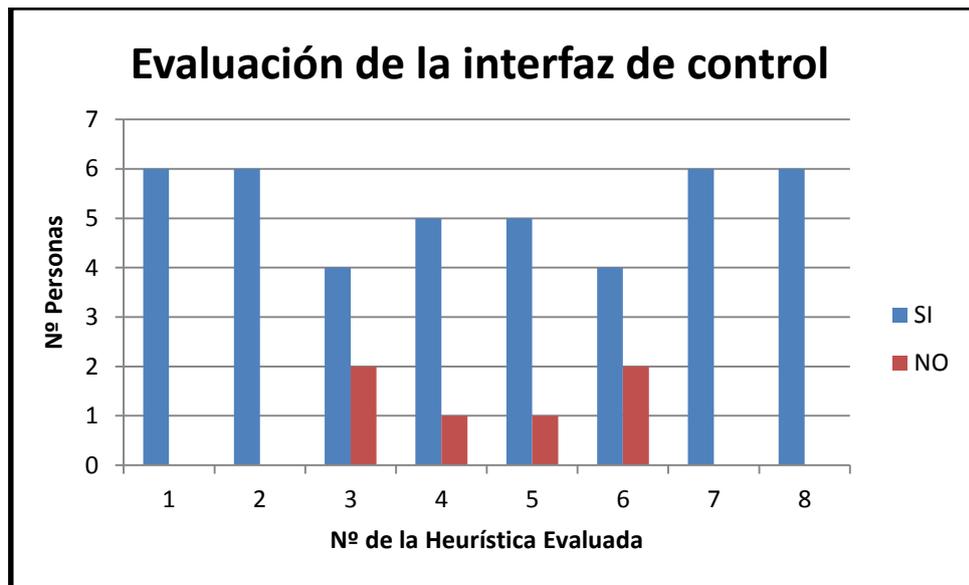
UNIVERSIDAD DE LAS FUERZAS ARMADAS ESPE			
EVALUACIÓN DE LA INTERFAZ DE CONTROL PARA UN CUADRICÓPTERO CON EL DISPOSITIVO KINECT			
Indicación.- Marque con una X el casillero de la respuesta que considere adecuada			
Nº	DESCRIPCIÓN DE LA PREGUNTA	SI	NO
1	¿La interfaz utiliza un lenguaje entendible para el usuario?		
2	¿La interfaz cuenta con las opciones para proveer al usuario una salida fácil?		
3	¿La interfaz muestra información a los usuarios de su estado?		
4	¿El usuario entiende el significado de las acciones de la interfaz?		
5	¿La interfaz tiene instrucciones visibles y están al alcance del usuario?		
6	¿La interfaz cuenta con prevención de errores y muestra información clara al usuario para que pueda realizar una determinada acción?		
7	¿La interfaz tiene diálogos estéticos y diseños minimalistas?		
8	¿La interfaz ofrece ayuda e información útil para el usuario?		

GRACIAS POR SU COLABORACIÓN

**Figura 101 Modelo de la encuesta**

## 4.4 RESULTADOS Y ANÁLISIS

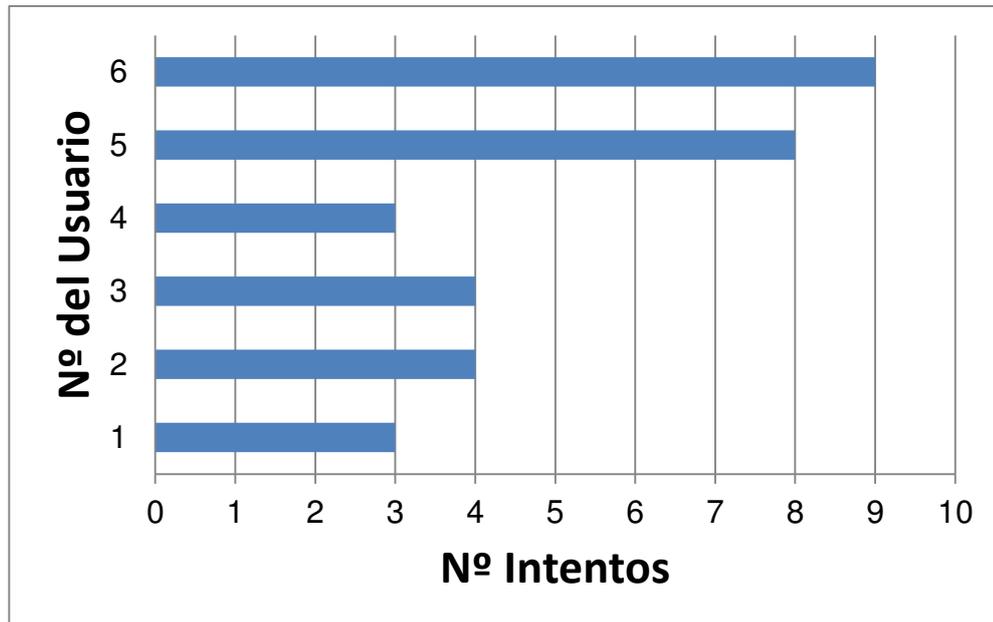
Los resultados obtenidos de la encuesta se presentan en la figura 102, el número de persona evaluadas fueron seis.



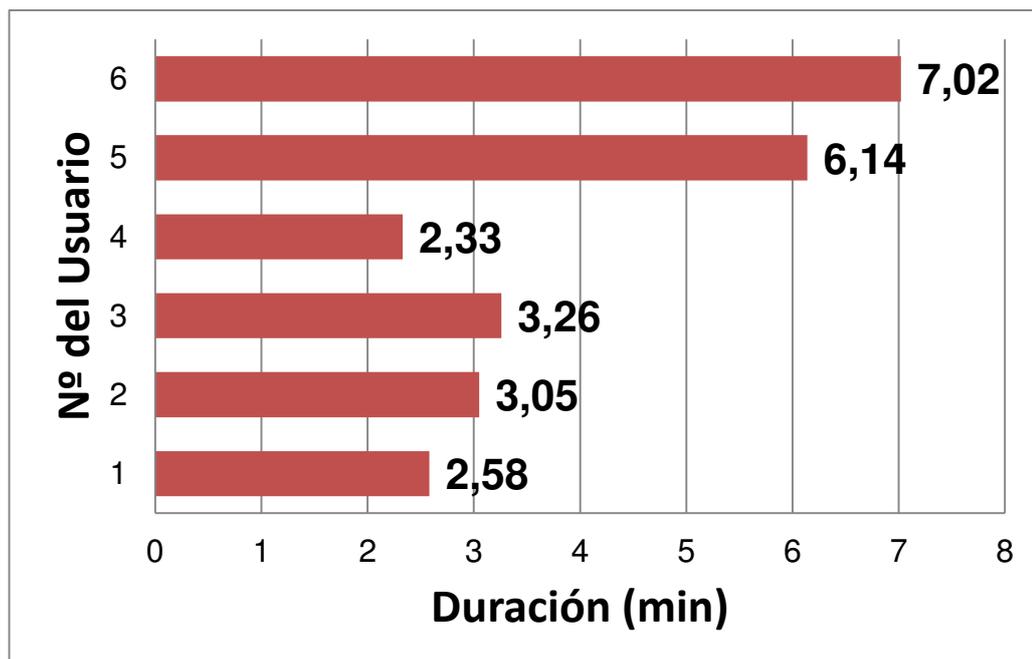
**Figura 102 Resultados de la encuesta**

Los resultados obtenidos de la encuesta en base a los parámetros de Nielsen muestran la conformidad del operador al utilizar la interfaz de control. El punto número 3 resulta algo limitado porque se tiene un pequeño retardo en el reconocimiento de la voz del usuario por parte del dispositivo Kinect. El punto número 4 y 5 no tuvo un resultado excelente ya que dos de los usuarios evaluados no tenían experiencia en computación. El resultado del punto número 6 se presentó porque los usuarios olvidaban conectar el dispositivo Kinect al computador.

El manejo de la interfaz es analizado con la información de la tabla 34, los resultados de las pruebas se representan gráficamente en la figura 103, la gráfica (a) se basa en el número de intentos que realizó el usuario hasta poder controlar el cuadricóptero sin ningún problema y la gráfica (b) se basa en el tiempo que le tomo al usuario en realizar dicha acción.



(a)



(b)

**Figura 103 Progreso de los usuarios en el control del cuadricóptero**

La figura 103(a) indica que los usuarios deben realizar como mínimo tres intentos para familiarizarse con los movimientos para controlar al cuadricóptero, en el caso de los dos últimos usuarios aunque no tenían mucho conocimiento en computación y tuvieron que intentar un mayor

número de veces hasta ganar experiencia en el manejo de la interfaz al final de la prueba podían controlar al cuadricóptero sin ninguna dificultad.

La figura 103(b) muestra que el tiempo que le toma a un usuario en manejar la interfaz de control no supera los 4 minutos, aunque dentro de los usuarios evaluados se tuvo dos personas que les tomo más tiempo el familiarizarse con la aplicación, los resultados obtenidos muestran que existe un progreso cada vez que el usuario utiliza la interfaz es decir que con el uso constante de la aplicación se puede dominar fácilmente los movimientos para que el cuadricóptero realice una tarea específica dentro de un área determinada.

## **CAPÍTULO V**

### **CONCLUSIONES Y RECOMENDACIONES**

#### **5.1 CONCLUSIONES**

El cuadricóptero Phantom 3 Standard, gracias a sus características técnicas, es catalogado dentro de la tercera generación de vehículos aéreos no tripulados, mismo que está a la par con modelos de última generación y su costo, en comparación con otros de alta gama, es menor.

El uso del dispositivo Kinect, como elemento primordial en desarrollo del trabajo de investigación, permite a las personas un manejo más intuitivo del cuadricóptero. A través de la creación de una interfaz de fácil uso se logra simplificar el trabajo humano, gracias a los componentes que integran el Kinect como el sensor de profundidad y el sensor de infrarrojos (emisor y receptor) se logró armar un plano tridimensional del usuario con el objetivo de capturar el esqueleto, reconocerlo y posicionarlo en el plano, gracias a la información que captura el arreglo de micrófonos se logró poner en funcionamiento la aplicación a través de comandos de voz específicos.

Para programar toda la serie de aplicativos de la interfaz de control entra en juego el Kit de desarrollo de software (SDK) que es provisto por Microsoft el cuál es de vital importancia para el desarrollo de la aplicación, mediante pruebas realizadas con diferentes versiones se llegó a la conclusión que la versión 1.7 nos proporciona una mayor estabilidad al trabajar en Visual Studio 2013, una vez escogida la versión adecuada del SDK se realizó un estudio en el cuál se logró habilitar las funciones para inicializar al dispositivo Kinect y obtener información de datos de profundidad , esqueleto y audio necesarios para el desarrollo del proyecto.

En vista que las extremidades superiores son la parte de mayor empleabilidad de cualquier tarea cotidiana que realiza un ser humano, se

utilizaron los movimientos de las manos para controlar al cuadricóptero, específicamente, se empleó el recurso de reconocimiento basado en coordenadas. Esto a decir verdad, ya que su nivel de efectividad es alto para el reconocimiento de gestos sencillos, como por ejemplo el levantar la mano o desplazarla en posición horizontal.

El reconocimiento de voz para el desarrollo del proyecto no presenta mayor recurso en el manejo de la interfaz, ya que el arreglo de micrófonos que tiene el dispositivo Kinect es muy sensible al ruido.

Previo al diseño de la interfaz, se investigó el funcionamiento del cuadricóptero con el objetivo de validar las operaciones previas a cada vuelo. La primera es la calibración de la brújula para cada nueva ubicación de vuelo. Seguido, se debe realizar una prueba de arranque de motores para validar que funcionan con normalidad. Finalmente, la aplicación DJI GO debe estar conectada al controlador remoto, con el propósito de obtener un rendimiento óptimo del sistema.

Para lograr una efectividad en el empleo del cuadricóptero previo a cada vuelo es indispensable asegurar que el controlador remoto, la batería de vuelo inteligente y el dispositivo móvil se encuentren completamente cargados. Además, se debe verificar que las hélices están correcta y firmemente montadas al cuadricóptero y que las condiciones climáticas para el uso de la aeronave no sean adversas, en especial si se tiene una velocidad de viento mayor a 10m/s. El cuadricóptero debe ser movilizado en espacios abiertos.

La implementación de una placa de control electrónica, que sirve de puente de información entre el ordenador, desde la interfaz de control, y el cuadricóptero, a través de la red inalámbrica de su controlador remoto, permite al usuario interactuar con la aplicación a través de diferentes movimientos de las manos.

A través de las pruebas de la interfaz de control se logró obtener la altura adecuada a la cual el dispositivo Kinect reconoce sin ningún inconveniente los movimientos del usuario ( $h=1,12\text{m}$ ) así como la distancia a la cual el usuario debe ubicarse en relación con el dispositivo Kinect ( $1,52\text{m}$ ), durante el desarrollo de la aplicación se varió el ángulo de inclinación del Kinect pero no se obtuvo ninguna mejora en relación a la captura de los movimientos.

Las primeras pruebas de funcionamiento de la interfaz con el cuadricóptero se empezaron a evaluar a partir de las 18:00 porque a partir de esa hora no se tenía exposición al sol que afecte a las cámaras del dispositivo, con el objetivo de evaluar a la interfaz de control a cualquier hora del día se procedió a diseñar un ambiente controlado en el cuál el dispositivo Kinect pueda captar los movimientos del usuario sin ningún inconveniente,

## **5.2 RECOMENDACIONES**

Es recomendable que antes de desarrollar cualquier aplicación con el dispositivo Kinect se verifique su funcionamiento con las diferentes versiones del SDK de Microsoft para la versión de Visual Studio en la cual se va a programar la aplicación, además se debe revisar que el cable que funciona como adaptador se encuentre trabajando correctamente, es decir que entregue el voltaje necesario para el funcionamiento del dispositivo Kinect.

Se recomienda verificar que tanto el dispositivo Kinect como la placa de control se encuentren correctamente colocados al computador antes de iniciar la aplicación con esto se obtiene un mejor rendimiento de la interfaz de control.

Se recomienda no volar la aeronave cerca de zonas en las que existan altos niveles de electromagnetismo ya que esto puede afectar a la brújula de a bordo y a la señal GPS, hay que considerar que el rendimiento de la

aeronave depende de factores ambientales como la densidad del aire y la temperatura.

Al momento de realizar las pruebas se pudo constatar que la autonomía de vuelo que proporciona la batería es de aproximadamente 25 minutos pero como precaución hay que considerar que la interfaz de control y el cuadricóptero funcionan sin ninguna novedad alrededor de 20 minutos continuos en el aire para obtener información de la batería el usuario debe revisar la aplicación DJI GO.

Para trabajos futuros, se recomienda el empleo de un dron que contenga una plataforma de código abierto, para que se pueda acceder a la trama de datos que es enviada desde el controlador remoto hacia la aeronave y de esta manera evitar el uso de circuitos externos para realizar este proceso.

Se recomienda ampliar la información existente para el manejo del dispositivo Kinect con el uso del kit de desarrollo de software de Microsoft, con motivo de diversificar las opciones para el desarrollo de un sistema similar frente a las opciones de código abierto. Esto permitirá que el dispositivo funcione en diferentes sistemas operativos. Asimismo, se recomienda emplear un espacio donde no se tenga luz externa que afecte a los diferentes sensores que posee el Kinect.

Se recomienda utilizar la aplicación propia del dron para elevarlo automáticamente hasta cierta altura, ya que la programación por voz de esta maniobra dentro de la interfaz presenta inconvenientes al momento del despegue de la aeronave y cuando el usuario toma control de la interfaz manualmente, produciendo un funcionamiento anómalo en el manejo del cuadricóptero.

Las pruebas de funcionamiento de la interfaz muestran que las personas que tuvieron alguna experiencia manejando el dispositivo Kinect pueden

utilizar la interfaz de control en menor tiempo en relación a personas inexpertas frente a este problema se recomienda para trabajos futuros investigar técnicas de reconocimiento más avanzadas con el objetivo de desarrollar una aplicación que reconozca mayor número de gestos y posturas con el objetivo de simplificar aún más el uso del dron.

## BIBLIOGRAFÍA

- ¿Qué son los Drones? Usos y Características (en Vídeo)*. (s.f.). Recuperado el 15 de 11 de 2015, de <http://www.tecnonauta.com/notas/1939-drones-que-es>
- AeroQuad*. (2016). Obtenido de <http://aeroquad.com>
- Carvajal, J., & Ramírez, J. (2003). *Análisis estadístico del cultivo y producción de la caña de azúcar*.
- Cortes, J. P. (s/a). *QUÉ ES UN DRONE*. Recuperado el 10 de 11 de 2015, de [http://www.academia.edu/7946946/QU%C3%89\\_ES\\_UN\\_DRONE](http://www.academia.edu/7946946/QU%C3%89_ES_UN_DRONE)
- Cuenca, M., Delgado, K., & Chuya, J. (2013). DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA PARA ELANÁLISIS DEL MOVIMIENTO HUMANO USANDO SENSORES KINECT.
- Dans, D. (02 de 07 de 2014). *La NASA estaría desarrollando un sistema de control aéreo para drones*. Recuperado el 15 de 11 de 2015, de <http://xombit.com/2014/09/nasa-sistema-control-aereo-drones>
- DJI. (2015). *PHANTOM 3 Guía de inicio rápido*.
- DJI. (2015). *PHANTOM 3 Standard Manual del usuario*.
- dji*. (05 de 02 de 2016). Recuperado el 10 de 04 de 2016, de <http://www.dji.com/es/product/phantom-3-standard/info>
- dji store*. (s.f.). Recuperado el 14 de 07 de 2016, de <http://store.dji.com/product/phantom-3-standard>
- dron, adaptación al español de drone*. (18 de 12 de 2014). Recuperado el 07 de 11 de 2015, de <http://www.fundeu.es/recomendacion/dron-adpatacion-al-espanol-de-drone/>
- Guacanes, L. (s.f). Estudio de la tecnología WPF para el desarrollo de aplicaciones de Escritorio y Web. 10.
- Ibañez, R., & Fanaro, Á. (2013). Herramienta para facilitar el desarrollo de aplicaciones basadas en Kinect. *16o Concurso de Trabajos Estudiantiles, EST 2013*.
- Inteligencia DYNAMICS*. (2012). Recuperado el 16 de 11 de 2015, de [www.iuavs.com/pages/aplicaciones\\_y\\_usos](http://www.iuavs.com/pages/aplicaciones_y_usos)

- Lorusso, E., Avila, D., Fasce, S., Pereira, G., Mazza, N., & Ierache, J. (2014). Avances en robótica situada aplica a la navegaciones autónoma de cuadricópteros. 818.
- Martínez Novo, J. (07 de Julio de 2015). Sistema de posicionamiento para un drone mediante odometría visual.
- Mathe, L., Samban, D., & Gómez, G. (2012). Estudio del funcionamiento del sensor Kinect y aplicaciones para bioingeniería. *Proyecto final de carrera de Ingeniería en Computación, Universidad Nacional de Córdoba, Córdoba, Veracruz.*
- Mayorga, R. (2009). *Sistema de Navegación para Vehículos Aéreos Cuadricópteros.*
- Melero, P. (2014). Teleoperación de un brazo robot mediante el sensor Kinect.
- Melero, P. (2014). Teleoperación de un brazo robot mediante el sensor Kinect.
- Mora, R. (2015). Políticas públicas y desarrollo rural en el Ecuador. Propuesta centrada en la gestión de las Juntas Parroquiales. 39.
- Nuño, J. (2012). RECONOCIMIENTO DE OBJETOS MEDIANTE SENSOR 3D.
- Oñate de Mora, M. (2015). Tipología de aeronaves pilotadas por control remoto. En *Dirección General de Industria, Energía y Minas de la Comunidad de Madrid, Los Drones y sus aplicaciones en la ingeniería civil* (págs. 49-57). Madrid, España: Consejería de Economía y Hacienda Comunidad de Madrid.
- Plan Nacional del buen Vivir.* (2013). Recuperado el 16 de 11 de 2015, de <http://plan.senplades.gob.ec/el-proceso-participativo>
- Que es un ROV.* (2014). Obtenido de <http://www.rovs.es/#!que-es-un-rov/coyg>
- Ramos, D. (2013). ESTUDIO CINEMÁTICO DEL CUERPO HUMANO MEDIANTE KINECT.
- Robledo, J. (2012). Prototipado de una aplicación interactiva mediante Kinect. 11-12.
- Rodriguez, Á. (2013). Sistema de entrenamiento con Kinect. 14.

- s/a. (2016). *Ingenius Worlds*. Obtenido de <http://www.ingeniusworlds.com/index.php/nosotros/vision>
- Samaniego, D. (2012). Sistema inteligente para reconocimiento de género mediante el sensor Kinect.
- Sánchez, A., & Olea, J. (2014). Diseño y construcción de una aeronave de ala rotativa para operaciones de seguridad fronteriza y respuesta a emergencias.
- Sánchez, E. (2012). Control de software educativo mediante kinect de Microsoft.
- Soares, C., Torres, J., Sobral, P., & Moreira, R. (2013). LoCoBoard: Low-Cost Interactive Whiteboard Using Computer. *Hindawi Publishing Corporation*, 1.
- Toribios Blázquez, J. (Septiembre de 2012). Modelado 3D de objetos a partir de la Kinect.

## **ANEXOS**