



La versión digital de esta tesis está protegida por la Ley de Derechos de Autor del Ecuador.

Los derechos de autor han sido entregados a la "ESCUELA POLITÉCNICA NACIONAL" bajo el libre consentimiento del (los) autor(es).

Al consultar esta tesis deberá acatar con las disposiciones de la Ley y las siguientes condiciones de uso:

- Cualquier uso que haga de estos documentos o imágenes deben ser sólo para efectos de investigación o estudio académico, y usted no puede ponerlos a disposición de otra persona.
- Usted deberá reconocer el derecho del autor a ser identificado y citado como el autor de esta tesis.
- No se podrá obtener ningún beneficio comercial y las obras derivadas tienen que estar bajo los mismos términos de licencia que el trabajo original.

El Libre Acceso a la información, promueve el reconocimiento de la originalidad de las ideas de los demás, respetando las normas de presentación y de citación de autores con el fin de no incurrir en actos ilegítimos de copiar y hacer pasar como propias las creaciones de terceras personas.

Respeto hacia sí mismo y hacia los demás.

ESCUELA POLITÉCNICA NACIONAL

**FACULTAD DE INGENIERÍA ELÉCTRICA Y
ELECTRÓNICA**

**IMPLEMENTACIÓN DE UN SISTEMA DE CAPTURA DE IMÁGENES
TÉRMICAS Y SU ALMACENAMIENTO EN LA WEB**

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

CHRISTIAN RENÉ TERÁN NARVÁEZ

(christianconh07@hotmail.com)

DIRECTOR: ING. RICARDO XAVIER LLUGSI CAÑAR, MSc.

(ricardo.llugsi@epn.edu.ec)

Quito, marzo 2018

DECLARACIÓN

Yo, Christian René Terán Narvárez, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

Christian René Terán Narvárez

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Christian René Terán Narváez, bajo mi supervisión.

Ing. Ricardo Llugin Cañar, MSc.

DIRECTOR DEL TRABAJO DE TITULACIÓN

AGRADECIMIENTOS

A mis padres, René y Antonia, por el gran sacrificio que han realizado a lo largo de todo este tiempo, por todos sus consejos y por el ejemplo de lucha constante que han mantenido para fortalecer la unión familiar.

A mis hermanas Fernanda y Karla, por su presencia incondicional en momentos difíciles.

A mis sobrinos Matias, Isabela y Martina, por ser la motivación que día a día me impulso a conseguir este gran objetivo.

Al Ingeniero Ricardo Llugsí, por el gran apoyo y recomendaciones que me supo brindar en momentos oportunos, para la culminación de este proyecto.

A todos esos amigos que estuvieron en momentos buenos y malos. Muchas gracias, porque de una u otra manera impulsaron a conseguir este logro: Eduardo (tío), Daniel (goofy), Mauricio (pakito), Jairo (pongo), Luis (rashu), Franklin (thor), Israel (irra), Alexis (cremosito), Wilmer (piedra), Carlos (yuyu), Jorge (pata), Alejo(alejo), Andrés (abuelo); les quedo agradecido profundamente.

Como no olvidar a las personas que estuvieron en el mismo camino durante toda esta etapa universitaria: Deivid, Napo, Fernando, Francisco, José, Camilo, David, Gabo.

De igual manera a todos ustedes "PP's", que conformaron esa fuerza y coraje necesario para terminar con todo este proceso.

Finalmente, un especial agradecimiento a ti Negra, por ser la persona que estuvo a toda hora impulsándome a conseguir este maravilloso reto, solo tú conoces lo acontecido durante este trayecto. Te quedo inmensamente agradecido.

Christian



DEDICATORIA

Lo más importante en la vida siempre será la FAMILIA, este duro proceso y a la vez un gran reto, va dedicado a todos ustedes.

Christian

CONTENIDO

DECLARACIÓN.....	I
CERTIFICACIÓN.....	II
AGRADECIMIENTOS	III
DEDICATORIA	IV
CONTENIDO	V
ÍNDICE DE FIGURAS	IX
ÍNDICE DE TABLAS	XIV
ÍNDICE DE CÓDIGOS	XV
ÍNDICE DE ECUACIONES	XVI
RESUMEN.....	XVII
PRESENTACIÓN	XVIII
CAPÍTULO 1: FUNDAMENTOS TEÓRICOS.....	1
1.1 INTRODUCCIÓN.....	1
1.2 RADIACIÓN ELECTROMAGNÉTICA.....	2
1.2.1 CAMPO ELÉCTRICO	2
1.2.2 CAMPO MAGNÉTICO	2
1.2.3 LONGITUD DE ONDA	3
1.3 ESPECTRO ELECTROMAGNÉTICO	3
1.3.1 TIPOS DE RADIACIÓN ELECTROMAGNÉTICA.....	4
1.4 RADIACIÓN INFRARROJA.....	7
1.4.1 INFRARROJO LEJANO	8
1.4.2 INFRARROJO MEDIO	8
1.4.3 INFRARROJO CERCANO	8
1.5 LUZ VISIBLE.....	9

1.6	TERMOGRAFÍA.....	10
1.6.1	VENTAJAS DE LA TERMOGRAFÍA	11
1.6.2	TERMÓMETRO DE INFRARROJOS.....	11
1.6.3	CÁMARAS TÉRMICAS.....	12
1.6.4	FACTORES DE INTERPRETACIÓN DE LAS IMÁGENES TÉRMICAS 16	
1.6.5	APLICACIONES DE LA TERMOGRAFÍA.....	18
1.7	PLACA RASPBERRY PI	22
1.7.1	Tipos de Placa Raspberry Pi	23
1.7.2	Ventajas de la Placa Raspberry Pi.....	24
1.7.3	Desventajas de la Placa Raspberry Pi.....	24
1.8	SISTEMA OPERATIVO RASPBIAN	25
1.9	TECNOLOGÍAS MÓVILES.....	26
1.9.1	TECNOLOGÍA MÓVIL 2.5G	28
1.9.2	TECNOLOGÍA MÓVIL 3G	29
CAPÍTULO 2: DISEÑO E IMPLEMENTACIÓN DEL SISTEMA		32
2.1	SISTEMA DE CAPTURA Y VISUALIZACIÓN DE IMÁGENES TÉRMICAS.....	32
2.1.1	PLACA RASPBERRY PI 2B	33
2.1.2	SENSOR DE CÁMARA TÉRMICA.....	36
2.1.3	PANTALLA	39
2.1.4	BATERÍA	41
2.1.5	SOFTWARE.....	42
2.1.6	CARCASA	45
2.2	SISTEMA DE TRANSMISIÓN DE DATOS HACIA INTERNET	45
2.2.1	MÓDEM USB	46
2.2.2	PROTOCOLO DE TRANSFERENCIA DE ARCHIVOS	48

2.2.3	COMPUTADOR	48
2.2.4	SERVIDOR ALOJADO EN LA NUBE	49
2.2.5	APLICACIÓN WEB	50
2.3	IMPLEMENTACIÓN DEL PROTOTIPO DE CÁMARA TÉRMICA	50
2.3.1	CONFIGURACIÓN DE PLACA RASPBERRY PI 2B	51
2.3.2	CONFIGURACIÓN DE PANTALLA TÁCTIL.....	53
2.3.3	CONFIGURACIÓN DE SENSOR DE CÁMARA TÉRMICA	57
2.3.4	CONFIGURACIÓN DE MÓDEM USB	61
2.3.5	CONFIGURACIÓN DE INTERFAZ GRÁFICA	67
2.3.6	CONFIGURACIÓN DE APLICACIÓN WEB	85
2.3.7	IMPLEMENTACIÓN DE FUENTE DE ALIMENTACIÓN	91
2.3.8	IMPLEMENTACIÓN DE CARCASA Y ENSAMBLAJE	93
CAPÍTULO 3: PRUEBAS DE FUNCIONAMIENTO Y ANÁLISIS DE RESULTADOS		98
3.1	SUGERENCIAS DE USO DEL PROTOTIPO DE CÁMARA TÉRMICA... ..	98
3.2	PRUEBA 1: SIMULACIÓN DE INCENDIO	99
3.3	PRUEBA 2: DIFERENCIA DE TEMPERATURA.....	101
3.4	PRUEBA 3: REFLEXIÓN DE MATERIALES	102
3.5	PRUEBA 4: INSPECCIÓN DE ESTRUCTURAS	103
3.6	PRUEBA 5: INSPECCIÓN DE HUMEDAD	104
3.7	PRUEBA 6: INSPECCIÓN ELÉCTRICA.....	105
3.8	PRUEBA 7: INSPECCIÓN DE EQUIPOS ELECTRÓNICOS	105
3.9	PRUEBA 8: INSPECCIÓN MECÁNICA.....	106
3.10	PRUEBA 9: VISIÓN NOCTURNA	107
3.11	PRUEBA 10: TRANSFERENCIA DE ARCHIVOS	108
CAPÍTULO 4: CONCLUSIONES Y RECOMENDACIONES		110
4.1	CONCLUSIONES	110

4.2 RECOMENDACIONES.....	111
REFERENCIAS.....	113
ANEXOS.....	122
ANEXO A: INSTALACIÓN DEL SISTEMA OPERATIVO RASPBIAN.....	122
ANEXO B: INSTALACIÓN Y CONFIGURACIÓN DEL ENTORNO DE DESARROLLO INTEGRADO QT CREATOR.	127
ANEXO C: FLIR LEPTON® Long Wave Infrared (LWIR) Datasheet.	131
ANEXO D: CÓDIGO FUENTE DE INTERFAZ GRÁFICA DE USUARIO.	132

ÍNDICE DE FIGURAS

CAPITULO 1

Figura 1.1 Perpendicularidad entre campo eléctrico y campo magnético [3]	2
Figura 1.2 Características de una Onda [6]	3
Figura 1.3 Espectro Electromagnético [4]	4
Figura 1.4 Tipos de Radiación Electromagnética [7]	5
Figura 1.5 Araña en Visión Infrarroja [10]	7
Figura 1.6 Tipos de Infrarrojos [12].....	9
Figura 1.7 Espectro de Luz Visible [13].....	9
Figura 1.8 Termografía de un motor con sus distintas temperaturas [19]	10
Figura 1.9 Termómetro de Infrarrojos [24]	12
Figura 1.10 Comparación entre Termómetro de IR y Cámara Termográfica [23].	12
Figura 1.11 Taza metálica con cinta adhesiva (“taipe” negro) [16]	13
Figura 1.12 Componentes de una Cámara Térmica [16].	13
Figura 1.13 Estructura de un Microbolómetro de Silicio-Amorfo (a-Si) [20].....	14
Figura 1.14 Cámara Térmica con microbolómetro no refrigerado [21].	14
Figura 1.15 Resoluciones de cámara de 640x480 y 320x240 pixeles [23].	15
Figura 1.16 Conductividad térmica de un material [26]	16
Figura 1.17 Medición de la emisividad de un objeto con ayuda de cinta adhesiva [28].	17
Figura 1.18 Temperatura reflejada sobre el vidrio de una ventana [22].	17
Figura 1.19 Mantenimiento predictivo en base a imágenes térmicas [23]	18
Figura 1.20 Aplicación de la termografía en motores [22].	19
Figura 1.21 Termografía de una tubería [22]	20
Figura 1.22 Imagen térmica de una instalación eléctrica [15]	20
Figura 1.23 Aplicación de la Termografía en instalaciones eléctricas de baja tensión [23].	21
Figura 1.24 Cámara Térmica utilizada por bomberos [29].	21
Figura 1.25 Detección de nivel de líquido en depósitos de almacenamiento [23].	22
Figura 1.26 Detección de llamas [23].	22
Figura 1.27 Placa Raspberry Pi y Logotipo de la Fundación [31]	23
Figura 1.28 Entorno de trabajo de Raspbian [35].	26

Figura 1.29 Estructura de un Sistema Celular [37]	27
Figura 1.30 Estándares Celulares de Tecnología Móvil [38].	27
Figura 1.31 Red GPRS [42].	28

CAPITULO 2

Figura 2.1 Placa Raspberry Pi 2B [44].	33
Figura 2.2 Distribución de pines GPIO [43]	34
Figura 2.3 Descripción de Pines GPIO [47]	35
Figura 2.4 Cámara Térmica Flir Lepton LWIR [47]	37
Figura 2.5 Pantalla Táctil conectada con la Raspberry Pi [51].	39
Figura 2.6 Batería Externa [55]	41
Figura 2.7 Logo del sistema operativo Raspbian [56]	43
Figura 2.8 NOOBS, la alternativa de instalación de Raspbian [56].	43
Figura 2.9 Sistemas Operativos alternativos para Raspberry Pi [56]	44
Figura 2.10 Módem Huawei E303 [58].	46
Figura 2.11 Computador Portátil ASUS X455LB [60].	49
Figura 2.12 Periféricos básicos para la configuración de la Raspberry Pi 2B [61].	51
Figura 2.13 Interfaz del programa SD Formatter 4.0 [61].	52
Figura 2.14 Diagrama de Flujo de Instalación del Sistema Operativo Raspbian. .	52
Figura 2.15 Conexión Pantalla - Raspberry Pi 2B.	53
Figura 2.16 Pantalla con errores en la imagen.	53
Figura 2.17 Configuración del archivo config.txt	54
Figura 2.18 Rangos de configuración de la pantalla.	54
Figura 2.19 Conexión de la cámara térmica con los pines GPIO de la Raspberry Pi 2B.	57
Figura 2.20 Conexión de cámara térmica y pantalla con la Raspberry Pi 2B.	58
Figura 2.21 Activación de la Interfaz SPI [65].	59
Figura 2.22 Activación de la Interfaz I2C [65].	59
Figura 2.23 Error del programa de pruebas Qt [65].	60
Figura 2.24 Imagen Térmica de prueba.	61
Figura 2.25 Módem USB "Huawei E303".	61

Figura 2.26 Estructura interna del módem USB con ranura para Tarjeta SIM.	62
Figura 2.27 Tarjeta SIM de la empresa “Claro”	62
Figura 2.28 Módem funcionando como Tarjeta de Red.	63
Figura 2.29 Cambio de modo de operación del módem USB.	63
Figura 2.30 Conexión a Internet mediante el protocolo ppp.	64
Figura 2.31 Configuración manual del módem USB.	64
Figura 2.32 Módem USB conectado a la red celular de “Claro”	65
Figura 2.33 Información de la conexión a la red celular de “Claro”.	65
Figura 2.34 Diagrama de Flujo de Instalación del IDE Qt Creator.	68
Figura 2.35 Diagrama de Flujo de Configuración del IDE Qt Creator.	69
Figura 2.36 Interfaz Gráfica de Usuario de la Cámara Térmica.	70
Figura 2.37 Creación de un nuevo Proyecto en Qt Creator.	71
Figura 2.38 Selección del Kit en Qt Creator.	72
Figura 2.39 Configuración de la Información de Clase del proyecto.	72
Figura 2.40 Estructura del Proyecto de Interfaz Gráfica.	73
Figura 2.41 Creación de cuenta de Amazon Web Services (AWS).	86
Figura 2.42 Sistema Operativo del servidor web.	87
Figura 2.43 Instancia t2.micro para el servidor web.	87
Figura 2.44 Asociación de IP Elástica con la Instancia.	88
Figura 2.45 Conexión al servidor mediante SSH.	89
Figura 2.46 Servidor Web Apache2.	89
Figura 2.47 Página de Inicio de la Aplicación Web.	90
Figura 2.48 Página de almacenamiento de Imágenes Térmicas.	91
Figura 2.49 Dimensiones de la Batería [55].	91
Figura 2.50 Cable de conexión Batería - Raspberry Pi 2B.	92
Figura 2.51 Cable de conexión Batería – Cargador.	92
Figura 2.52 Componentes del prototipo sobre la Plancha de Acrílico antes de cortar.	93
Figura 2.53 Diseño de las paredes de la carcasa.	93
Figura 2.54 Pared Frontal.	94
Figura 2.55 Pared Posterior.	94
Figura 2.56 Pared Lateral Izquierda.	95
Figura 2.57 Pared Lateral Derecha.	95

Figura 2.58 Pared Superior.....	96
Figura 2.59 Pared Inferior.	96
Figura 2.60 Disposición Interna de los equipos.	96
Figura 2.61 Modo de recarga de batería.	97
Figura 2.62 Prototipo de Cámara Térmica.	97
Figura 2.63 Descarga del paquete NOOBS [61].....	122
Figura 2.64 Archivos que contiene el paquete NOOBS.	123
Figura 2.65 Elección del sistema operativo Raspbian en la aplicación NOOBS [61].	123
Figura 2.66 Inicio del proceso de Instalación de Raspbian [61].....	124
Figura 2.67 Transcurso del proceso de Instalación de Raspbian [61].	124
Figura 2.68 Finalización del Proceso de Instalación de Raspbian [61].....	124
Figura 2.69 Entorno Gráfico de PIXEL sobre el sistema operativo Raspbian [61].	125
Figura 2.70 Interfaz gráfica del escritorio del sistema operativo Raspbian [61]. .	125
Figura 2.71 Instalación de las herramientas de compilación de Qt.	127
Figura 2.72 Instalación del IDE Qt Creator.	127
Figura 2.73 Programa Qt Creator instalado en la Raspberry Pi 2B.	127
Figura 2.74 Configuración de las opciones del IDE Qt Creator.	128
Figura 2.75 Configuración de Build & Run.	128
Figura 2.76 Configuración de la Versión de Qt Creator.....	129
Figura 2.77 Configuración del Compilador de Qt Creator.	129
Figura 2.78 Configuración del Depurador de Qt Creator.....	129
Figura 2.79 Configuración del Kit de Qt Creator.	130
Figura 2.80 Entorno de Desarrollo de Qt Creator.	130

CAPITULO 3

Figura 3.1 Inicio de la prueba de incendio.	100
Figura 3.2 Imagen Normal y Térmica de la prueba de incendio.	100
Figura 3.3 Escala de temperaturas según el color.	101
Figura 3.4 Caso #1 de diferencia de temperatura.	102
Figura 3.5 Caso #2 de diferencia de temperatura.	102

Figura 3.6 Imagen Normal y Térmica, reflexión del vidrio.	103
Figura 3.7 Imagen Normal y Térmica, inspección estructural.	104
Figura 3.8 Imagen Normal y Térmica, inspección de humedad.	104
Figura 3.9 Imagen Normal y Térmica, inspección eléctrica.	105
Figura 3.10 Imagen Normal y Térmica, inspección electrónica.	106
Figura 3.11 Imagen Normal y Térmica, inspección mecánica.	107
Figura 3.12 Imagen normal con luz visible.	108
Figura 3.13 Imagen Normal y Térmica, visión nocturna.	108
Figura 3.14 Directorio Local.	109
Figura 3.15 Directorio Remoto.	109

ÍNDICE DE TABLAS

CAPITULO 1

Tabla 1.1 Bandas del Espectro Electromagnético [8].	6
Tabla 1.2 Tipos de Placa Raspberry Pi [32]	23

CAPITULO 2

Tabla 2.1 Valores posibles del comando <code>hdmi_group</code> [63].	55
Tabla 2.2 Valores posibles del comando <code>hdmi_cvt</code> [63].	56
Tabla 2.3 Valores posibles del comando <code>hdmi_drive</code> [63].	57
Tabla 2.4 Significado del color de led del módem USB [67].	66

ÍNDICE DE CÓDIGOS

CAPITULO 2

Código 2.1 Condicionales del Preprocesador.....	73
Código 2.2 Declaración de librerías y clases en “mainwindow.h”.....	74
Código 2.3 Declaración de Punteros.....	75
Código 2.4 Declaración de Slots.....	76
Código 2.5 Fichero de Fuente Principal.....	77
Código 2.6 Declaración de librerías en “mainwindow.cpp”.....	78
Código 2.7 Modelo de color RGB888.....	79
Código 2.8 Modelo de color RGB888.....	79
Código 2.9 Título de la aplicación.....	79
Código 2.10 Posición de la Imagen Térmica.....	80
Código 2.11 Color de relleno (rojo) para indicar error de comunicación.....	80
Código 2.12 Actualización de Imagen Térmica.....	81
Código 2.13 Recepción de datos desde el sensor Flir Lepton.....	81
Código 2.14 Captura y envío de archivo de imagen térmica.....	82
Código 2.15 Botón conectado a la señal “guardarCaptura”.....	82
Código 2.16 Señal “archivo”.....	83
Código 2.17 Señal “reenviarCapturas”.....	83
Código 2.18 Script “enviater”.....	83
Código 2.19 Señales “apagaSistema” y “reinicioSistema”.....	84
Código 2.20 Script “apagater”.....	84
Código 2.21 Script “reiniter”.....	84
Código 2.22 Configuración de botones.....	85

ÍNDICE DE ECUACIONES

CAPITULO 1

Ecuación 1.1 Relación entre longitud de onda y frecuencia.....	4
Ecuación 1.2 Energía de un fotón.	4

RESUMEN

El trabajo que cumplen los bomberos para rescatar a personas atrapadas durante un incendio requiere realizarse a la mayor brevedad posible. Un factor para cumplir con este objetivo es una buena visión dentro de este ambiente. Por tal motivo se ha vuelto cada vez más necesario la utilización de tecnología que facilite una mejor visibilidad.

El empleo de cámaras térmicas es una alternativa ante este problema para obtener una mejor orientación dentro de un incendio. La visión infrarroja de la cámara térmica permite al bombero detectar rastros de puntos calientes a través del humo generado por las llamas. De esta manera se posibilita a los bomberos un eficiente desempeño en el rescate de posibles víctimas.

También es importante el uso de cámaras térmicas para labores de mantenimiento preventivo relacionadas con el incremento o disminución de temperatura, ya que a simple vista no es posible notar estos detalles.

En el presente proyecto se realiza el diseño e implementación de un prototipo de cámara térmica que permita realizar capturas de imágenes térmicas para ser enviadas a una aplicación web, mediante la red celular.

Para la obtención de las imágenes térmicas se utiliza el sensor de cámara térmica “LWIR Flir Lepton”, que detecta la radiación infrarroja y la convierte en un equivalente de colores. El prototipo contará con una Interfaz Gráfica de Usuario (GUI) para su funcionamiento.

Finalmente se somete el prototipo a varias pruebas de funcionamiento como: la simulación de un incendio, labores de mantenimiento preventivo y pruebas de transferencia de archivos.

PRESENTACIÓN

En el presente trabajo se muestra la implementación de un prototipo de cámara térmica que permita realizar capturas de imágenes para ser enviadas hacia un servidor web. El desarrollo de tal proyecto está compuesto de 4 capítulos detallados a continuación.

En el capítulo 1, se describe las características del espectro electromagnético, la radiación infrarroja, la luz visible, detalles acerca del estudio de la termografía, las prestaciones que ofrece la placa Raspberry Pi, así como el análisis de ciertas tecnologías móviles.

En el capítulo 2, se detallan las especificaciones técnicas que posee cada componente utilizado en el proyecto. Se plantea el diseño del prototipo conformado por el sistema de captura y visualización de imágenes térmicas, y el sistema para la transferencia de archivos al servidor web. Dentro de este capítulo también consta la implementación de cada componente del prototipo, especificando el procedimiento para su respectiva configuración, así como la programación correspondiente utilizada para su funcionamiento.

En el capítulo 3, se presentan los pasos necesarios para una correcta utilización del equipo. También se muestran los distintos tipos de pruebas realizadas para incendios y tareas de mantenimiento preventivo, así como para la transferencia de archivos; de esta manera se confirma el buen funcionamiento y capacidad del equipo.

En el capítulo 4, se detallan las conclusiones obtenidas luego del desarrollo del prototipo de cámara térmica, así como las posibles recomendaciones que puedan añadirse para mejorar la capacidad y funcionamiento del equipo en trabajos de investigación posteriores.

CAPÍTULO 1: FUNDAMENTOS TEÓRICOS

1.1 INTRODUCCIÓN

En este capítulo se detalla las pautas principales sobre el estudio de la radiación electromagnética, enfocando la temática sobre aspectos relacionados con la luz visible, así como también aspectos con los que cuenta la luz infrarroja. Se expondrá un análisis sobre el espectro electromagnético mostrando los distintos tipos de radiaciones que existen en la naturaleza y específicamente tratando el rango de longitudes de onda en que se encuentra la radiación infrarroja.

Este tipo de radiación es de mucho interés para el presente proyecto, ya que las imágenes térmicas logran ser interpretadas en función de la luz infrarroja. Se pretende analizar sus respectivas características, así como la forma en la cual el ojo humano puede percibir este tipo de radiaciones, basándose en la observación de dichas imágenes térmicas.

También se expone todo lo relacionado a las características que poseen las partes del prototipo de cámara térmica; es decir, detallar los componentes principales que conforman el proyecto como: la Tarjeta Raspberry Pi, el módulo de cámara térmica LWIR¹ (*Long Wave Infrared*, Infrarrojo de Onda Larga), la pantalla táctil LCD² (*Liquid Cristal Display*, Pantalla de Cristal Líquido), el módulo para la comunicación inalámbrica, y especificar el tipo de batería con la cual se mantendrá operativo el prototipo.

Uno de los componentes en los cuales se tendrá gran énfasis en cuanto a su funcionamiento y operación es el módulo de cámara térmica LWIR, ya que será el sensor encargado de interpretar las radiaciones infrarrojas para transformarlas en equivalencias de colores, y así permitir que la vista humana pueda percibir este tipo de radiación, desplegando dichas imágenes sobre la pantalla táctil.

¹ LWIR, es el tipo de onda electromagnética con que trabaja el sensor de cámara térmica para poder detectar la radiación infrarroja.

² En las pantallas LCD, se utiliza cristales líquidos para activar o desactivar los píxeles y así revelar un color específico en base a cierto nivel de electricidad que le sea inducido.

1.2 RADIACIÓN ELECTROMAGNÉTICA

Se conoce como radiación electromagnética a la interacción que existe entre los campos eléctricos y magnéticos, propagándose a través del espacio en forma de ondas portadoras de energía. Las ondas electromagnéticas no necesitan de un medio físico para poder ser propagadas, es decir, no requieren de ningún material para desplazarse. En cuanto a la dirección, las ondas del campo eléctrico son perpendiculares a las ondas del campo magnético [1].

1.2.1 CAMPO ELÉCTRICO

Está compuesto por las fuerzas de carácter eléctrico, el cual se produce por la diferencia de voltaje, y entre mayor sea la magnitud de este voltaje, mayor será el campo eléctrico obtenido [2].

1.2.2 CAMPO MAGNÉTICO

Está constituido por las fuerzas magnéticas, se origina de las corrientes eléctricas y su magnitud depende de la intensidad de esta corriente. Presentan su mayor valor en el origen, es decir, en la fuente que origina esta corriente eléctrica. Estas ondas son capaces de atravesar los medios físicos como las paredes, pero al aumentar su distancia se disminuirá la intensidad de este campo magnético [2].

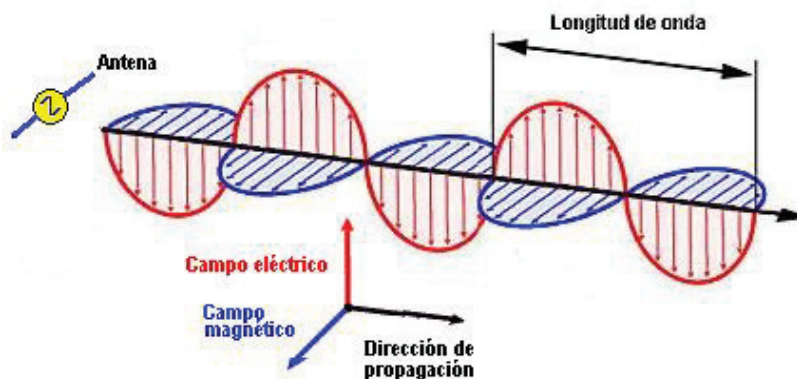


Figura 1.1 Perpendicularidad entre campo eléctrico y campo magnético [3]

En una onda electromagnética cuando las líneas de fuerza del campo eléctrico son perpendiculares a la superficie de La Tierra se trata de una polarización

vertical en tanto que si dichas fuerzas son paralelas a la superficie terrestre se tiene una polarización horizontal [3].

1.2.3 LONGITUD DE ONDA

El término longitud de onda es la distancia recorrida por una onda en un determinado intervalo de tiempo; los distintos tipos de radiaciones que existen están diferenciadas en el espectro electromagnético mediante su respectiva longitud de onda [4].

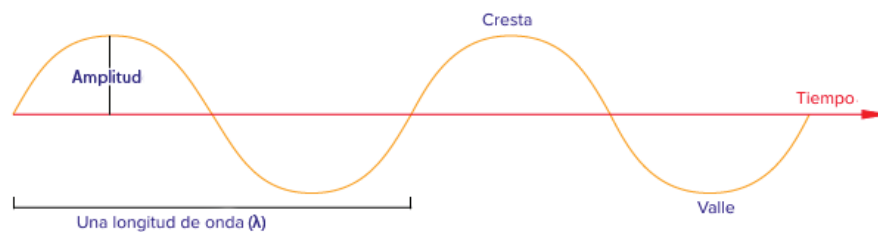


Figura 1.2 Características de una Onda [6]

Existen diferentes formas de energía electromagnética, una de las más conocidas por el ser humano y percibida naturalmente es la luz visible. Se tiene otras de igual relevancia como las ondas de radio, ondas térmicas, los rayos ultravioleta o los rayos X, todas estas formas de energía poseen su respectiva longitud de onda.

1.3 ESPECTRO ELECTROMAGNÉTICO

El espectro electromagnético es la distribución energética de todas las posibles formas de radiación que existen, dispuestas en una escala de menor a mayor longitud de onda, o también en valores de mayor a menor frecuencia respectivamente [7].

Todas las ondas electromagnéticas viajan a la velocidad de la luz en el vacío independientemente de su frecuencia y longitud de onda. Las radiaciones del espectro electromagnético presentan propiedades típicas de un movimiento ondulatorio tales como la difracción y la interferencia [6].

En una determinada forma de radiación electromagnética, la velocidad de la luz con valor de constante $c = 3 \times 10^8$ [m/s], se expresa mediante la relación que existe entre la longitud de onda (λ) y la frecuencia (f) [5] [6].

$$c = \lambda * f$$

Ecuación 1.1 Relación entre longitud de onda y frecuencia.

La radiación está compuesta de elementos conocidos como cuantos o fotones, los cuales presentan un valor de energía (E), determinado por la relación entre la constante de Planck $h = 6,626 \times 10^{-34}$ [J. s] y la frecuencia (f) [5].

$$E = h * f$$

Ecuación 1.2 Energía de un fotón.

Cada zona del espectro electromagnético tiene asignado un nombre en específico, sin embargo no existen divisiones o cortes exactos entre cada una de estas. El espectro electromagnético se extiende desde la zona con longitudes de onda corta y frecuencias altas (rayos cósmicos, rayos gamma, rayos X y el rango de los ultravioletas), hasta la zona con longitudes de onda larga y frecuencias bajas (ondas de radio, microondas y el rango de los infrarrojos) [5] [7].

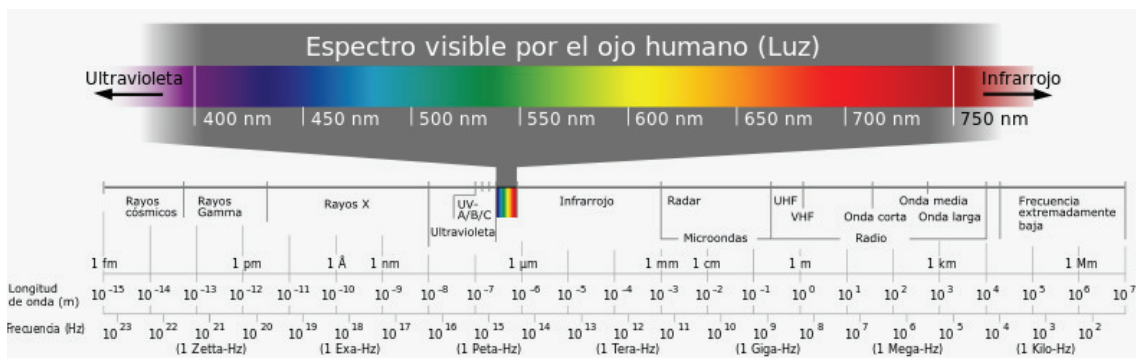


Figura 1.3 Espectro Electromagnético [4]

1.3.1 TIPOS DE RADIACIÓN ELECTROMAGNÉTICA

Las formas de energía que se muestran a lo largo del espectro electromagnético se encuentran distribuidas y diferenciadas mediante su longitud de onda, su

frecuencia asociada y su energía fotónica; estas tres variables se relacionan mediante las ecuaciones tratadas anteriormente [7].

En cualquier tipo de radiación el valor de su energía fotónica es directamente proporcional a la magnitud de su frecuencia. Por lo general los valores de longitud de onda se trabajan en el vacío ya que al actuar en un medio (materia) su valor se reduce [7].

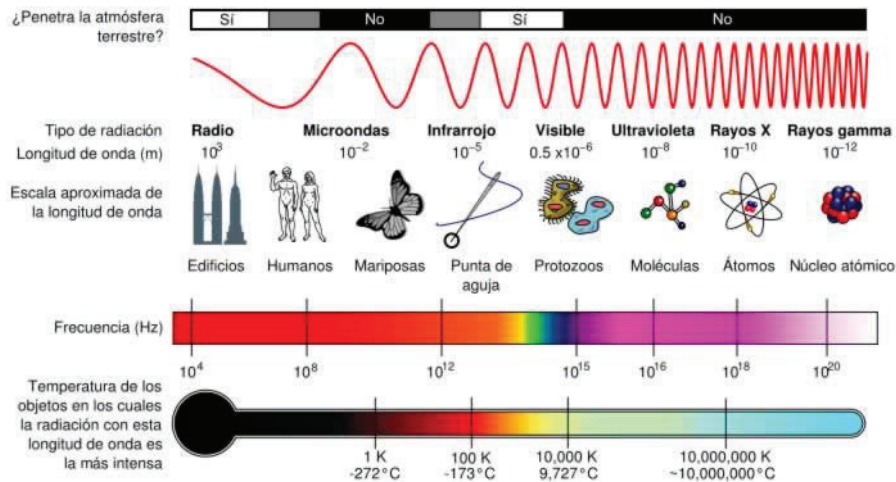


Figura 1.4 Tipos de Radiación Electromagnética [7]

Los principales tipos de radiación electromagnética se clasifican por su longitud de onda y frecuencia en:

- Rayos Cósmicos
- Rayos Gamma
- Rayos X
- Luz Ultravioleta
- Luz Visible
- Luz Infrarroja
- Microondas
- Ondas de radio

En la tabla 1.1 se refleja las distintas bandas del espectro electromagnético con sus respectivos valores de longitud de onda, frecuencia y energía:

Tabla 1.1 Bandas del Espectro Electromagnético [8].

Banda	Longitud de Onda [m]	Frecuencia [Hz]	Energía [J]
Rayos gamma	< 10 pm	> 30,0 EHz	> $20 \cdot 10^{-15}$ J
Rayos X	< 10 nm	> 30,0 PHz	> $20 \cdot 10^{-18}$ J
Ultravioleta extremo	< 200 nm	> 1,5 PHz	> $993 \cdot 10^{-21}$ J
Ultravioleta cercano	< 380 nm	> 789 THz	> $523 \cdot 10^{-21}$ J
Luz Visible	< 780 nm	> 384 THz	> $255 \cdot 10^{-21}$ J
Infrarrojo cercano	< 2,5 μ m	> 120 THz	> $79 \cdot 10^{-21}$ J
Infrarrojo medio	< 50 μ m	> 6,00 THz	> $4 \cdot 10^{-21}$ J
Infrarrojo lejano	< 1 mm	> 300 GHz	> $200 \cdot 10^{-24}$ J
Microondas	< 30 cm	> 1 GHz	> $2 \cdot 10^{-24}$ J
Ultra Alta Frecuencia - Radio	< 1 m	> 300 MHz	> $19,8 \cdot 10^{-26}$ J
Muy Alta Frecuencia - Radio	< 10 m	> 30 MHz	> $19,8 \cdot 10^{-28}$ J
Onda Corta - Radio	< 180 m	> 1,7 MHz	> $11,22 \cdot 10^{-28}$ J
Onda Media - Radio	< 650 m	> 650 KHz	> $42,9 \cdot 10^{-29}$ J
Onda Larga - Radio	< 10 Km	> 30 KHz	> $19,8 \cdot 10^{-30}$ J
Muy Baja Frecuencia - Radio	> 10 Km	< 30 KHz	< $19,8 \cdot 10^{-30}$ J

El equipo conocido como espectroscopio permite detectar una gran región del espectro electromagnético rondando valores de longitud de onda de 2 a 2500 [nm], a diferencia del ojo humano que solo es capaz de percibir valores de luz visible de entre 400 y 750 [nm] aproximadamente [7].

A continuación se tratará los temas sobre radiación infrarroja y luz visible, ya que con este tipo de radiaciones operará el prototipo de cámara térmica.

1.4 RADIACIÓN INFRARROJA

La radiación de tipo infrarroja se encuentra situada en la región del espectro electromagnético con valores de frecuencia de entre 300 [GHz] hasta los 400 [THz], es decir, niveles de longitud de onda desde 1[mm] hasta los 750 [nm] respectivamente [7].

La radiación infrarroja se asocia directamente con el calor que emiten los cuerpos, aunque también este calor puede ser generado por otro tipo de fuentes como los diodos emisores de luz y por algún tipo de láser [9].

Las señales infrarrojas se utilizan en distintos campos como la Astronomía para descubrir estrellas y otros cuerpos lejanos en el espacio. Para visión nocturna al brindar la posibilidad de diferenciar los cuerpos en base a las emisiones de calor que presentan. También se utiliza en los mandos a distancia de los televisores, al emplear el envío de la señal infrarroja codificada desde un transmisor hacia su respectivo receptor [9].

Para conseguir una comunicación óptima de estas señales infrarrojas, es indispensable contar con Línea de Vista.

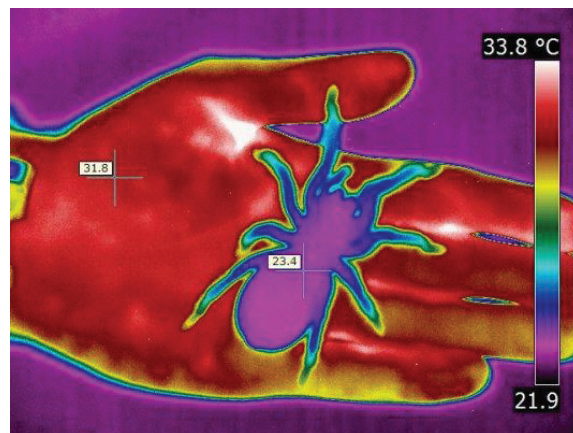


Figura 1.5 Araña en Visión Infrarroja [10]

El rango que comprende la radiación infrarroja está compuesto por tres tipos de ondas:

1. Infrarrojo Lejano: 10 a 1000 [μm].
2. Infrarrojo Medio: 2,5 a 10 [μm].
3. Infrarrojo Cercano: 0,75 a 2,5 [μm].

1.4.1 INFRARROJO LEJANO

Este tipo de radiación trabaja con frecuencias desde los 30 [THz] hasta los 300 [GHz], es decir, con rangos de longitud de onda de entre 10 a 1000 [μm] respectivamente.

El rango de longitud de onda comprendido entre los 200 y 1000 [μm] suele llamarse "radiación submilimétrica" en astronomía, la región más lejana puede considerarse en ciertos casos como microondas. Por lo tanto el rango de infrarrojo lejano comúnmente se trabaja con valores por debajo de los 200 [μm] [7].

1.4.2 INFRARROJO MEDIO

Se encuentra en el rango de frecuencias de entre 120 y 30 [THz], y sus longitudes de onda varían desde 2,5 a 10 [μm] respectivamente.

El rango de infrarrojo medio es conocido como "región de huella digital", ya que su espectro de absorción en cada compuesto es muy específico. La absorción se manifiesta por vibración molecular, que consiste en la vibración de los átomos de una molécula desde su posición de equilibrio. Los cuerpos irradian fuertemente en este rango [7] [11].

1.4.3 INFRARROJO CERCANO

Este tipo de infrarrojo se encuentra en el rango de frecuencias de entre 400 y 120 [THz], y su rango de longitudes de onda va desde 0,75 a 2,5 [μm] respectivamente. Los procesos físicos que se producen en este rango son similares a los de la luz visible, tema que será tratado a continuación.



Figura 1.6 Tipos de Infrarrojos [12]

1.5 LUZ VISIBLE

Este tipo de radiación electromagnética presenta valores de longitud de onda de entre 400 y 750 [nm] aproximadamente, ubicándose por debajo de la radiación infrarroja. La luz visible se caracteriza por lograr ser percibida naturalmente por el ojo humano, y cabe señalar que dicha radiación ocupa una zona muy pequeña dentro del espectro electromagnético [7].

El proceso para lograr percibir el ambiente inicia cuando a un objeto le llega luz con diferentes valores de longitud de onda, por tanto diferentes valores de energía, el objeto según su material absorberá determinados valores de energía y otros no, las señales que no se absorben rebotan y se dirigen hacia el ojo humano con cierto valor de longitud de onda, esta información se transmite por el ojo hacia el cerebro y se traduce en los distintos tipos de colores.

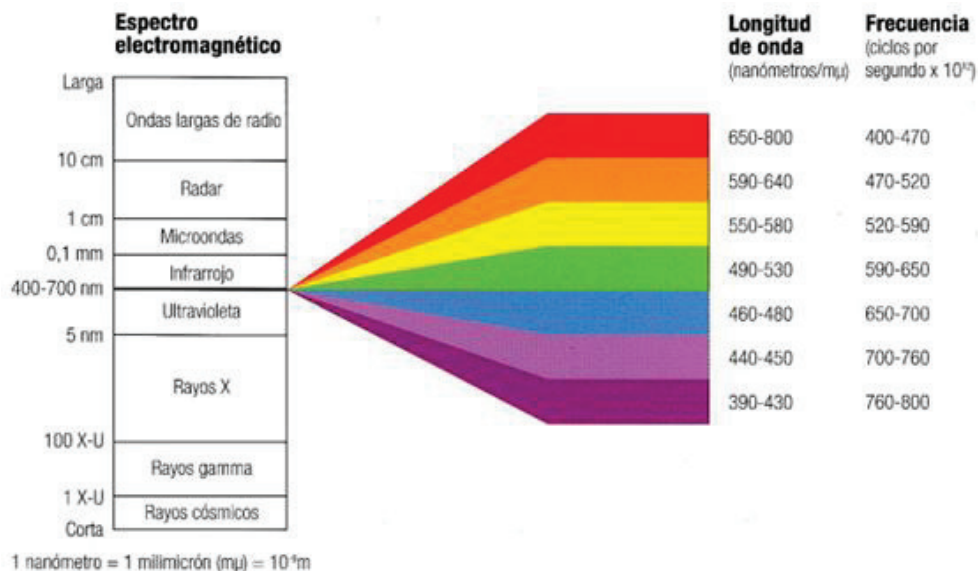


Figura 1.7 Espectro de Luz Visible [13]

1.6 TERMOGRAFÍA

Se conoce como termografía a la técnica que permite medir la temperatura superficial de los objetos, mediante la radiación infrarroja que estos emiten. Se puede realizar estos procesos de medición sin la necesidad del contacto físico entre el instrumento de medida y dichas superficies [15].

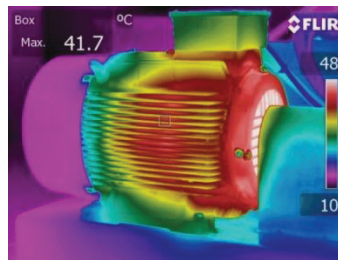


Figura 1.8 Termografía de un motor con sus distintas temperaturas [19]

Entre más alta sea la temperatura de un objeto, más alta será la energía espectral radiada y más pequeña su longitud de onda predominante o pico. Por ejemplo se tiene al Sol con una temperatura de 5500 [°C] y un pico de longitud de onda de 530 [nm], lo que para el ojo humano significa luz visible de color verde, que es el tipo de color con mayor facilidad de ver [14].

La mayoría de radiaciones del espectro infrarrojo no pueden ser utilizadas para sistemas de detección, debido a que el agua o el dióxido de carbono absorben este tipo de radiación [14].

Los valores de longitud de onda con buenas capacidades de transmisión en cada tipo de infrarrojo, se muestran a continuación:

- **LWIR**, (*Long Wavelength Infrared*, Infrarrojo de Longitud de Onda Larga) su rango se extiende desde 8 a 14 [μm], con una efectividad del 100% en longitudes de onda entre 9 y 12 [μm]. Este tipo de ondas ofrecen una excelente visibilidad.
- **MWIR**, (*Mid Wavelength Infrared*, Infrarrojo de Longitud de Onda Media) trabaja con rangos de longitud de onda de entre 3,3 y 5 [μm].
- **SWIR**, (*Short Wavelength Infrared*, Infrarrojo de Longitud de Onda Corta) opera con rangos de longitud de onda de entre 0,35 y 2,5 [μm]. Con este

tipo de onda se tiene imágenes muy pobres del entorno. Se necesita algún tipo de luz adicional, sea esta natural o artificial, para mejorar la imagen

1.6.1 VENTAJAS DE LA TERMOGRAFÍA

Las diferentes ventajas que brinda el uso de la tecnología termográfica son en su gran mayoría dedicadas al campo industrial; el trabajo de los sistemas mecánicos produce calor y cuando es elevado puede originar averías. La termografía permite detectar estos aumentos de temperatura, con el fin de corregir los problemas y así evitar daños graves [22].

Al utilizar la tecnología termográfica las fábricas podrían estar funcionando las 24 horas del día, los 365 días del año sin presentar costosas averías gracias al mantenimiento predictivo con imágenes térmicas. Las razones más relevantes para utilizar cámaras térmicas son:

1. No se tiene contacto con el objeto de análisis.
2. Las imágenes son bidimensionales.
3. Se realizan las mediciones en tiempo real.

Una de las principales ventajas que brinda la termografía es poder prevenir posibles incendios, debido a problemas eléctricos, ya que con el tiempo el rendimiento de los sistemas eléctricos se deteriora, produciendo calor en los equipos y por tanto puede ocasionarse chispas en las instalaciones que generen graves incendios [23].

1.6.2 TERMÓMETRO DE INFRARROJOS

Los termómetros de infrarrojos permiten detectar la temperatura de un solo punto, lo que significa una seria desventaja frente a las cámaras térmicas, que permiten una apreciación completa del ambiente que se monitorea.

Las cámaras térmicas al brindar la posibilidad de analizar componentes completos, permiten detectar todos los riesgos de calentamiento aunque sean

mínimos, en contraparte con los termómetros de infrarrojos se puede ignorar fácilmente los sectores con fallas de un equipo [22].



Figura 1.9 Termómetro de Infrarrojos [24]

La capacidad de detección térmica de un entorno está determinada por la resolución de imagen que se tenga. Por ejemplo al tener una cámara térmica con resolución de 60x60 pixeles, significa que se trabaja con 3600 puntos de medición, lo que significaría utilizar 3600 termómetros de infrarrojos al mismo tiempo. A continuación se muestra en el gráfico la respectiva comparación [22]:

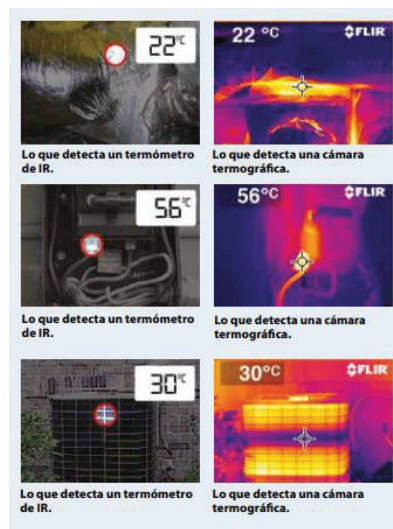


Figura 1.10 Comparación entre Termómetro de IR y Cámara Termográfica [23].

1.6.3 CÁMARAS TÉRMICAS

Las cámaras térmicas son equipos diseñados para detectar la radiación infrarroja que emiten los objetos y poder transformarla en una imagen posible de observar por el ojo humano. Los diferentes tonos de color de dicha imagen representarán las distintas intensidades de calor que poseen los objetos [16].

Se puede presentar el caso en que dos objetos a una misma temperatura irradian energía infrarroja a distintas intensidades, esto se debe al tipo de material. Se puede verificar esta condición en la figura 1.14 en la cual la taza metálica presenta distinta intensidad de radiación infrarroja en referencia a la cinta adhesiva (“taipe negro”) [16].

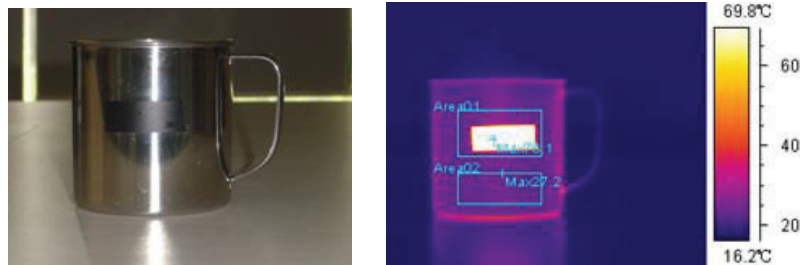


Figura 1.11 Taza metálica con cinta adhesiva (“taipe” negro) [16]

1.6.3.1 Funcionamiento de las Cámaras Térmicas

Normalmente una cámara térmica básica consta de los siguientes componentes:

- Lentes.
- Filtro.
- Detector o Microbolómetro.
- Circuito de procesamiento de la imagen.
- Interfaz de Usuario (pantalla, salida de vídeo, memoria, etc.)

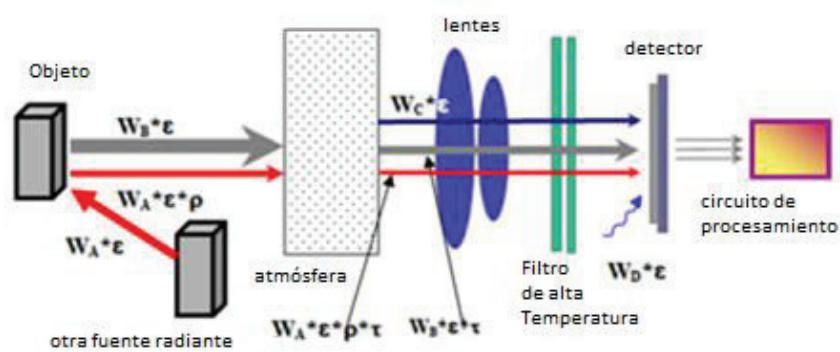


Figura 1.12 Componentes de una Cámara Térmica [16].

Todo tipo de material que presente una temperatura por encima del cero absoluto (0°K , -273°C) emite energía infrarroja. La energía emitida por la

radiación infrarroja se convierte en una señal eléctrica por medio de un detector conocido como microbolómetro, y a su vez según el valor de estas señales eléctricas se puede interpretarlas en un determinado rango de colores [16].

Un bolómetro es un instrumento que mide la cantidad de radiación electromagnética que emite un objeto. En las cámaras térmicas se utiliza un microbolómetro, normalmente optimizado para tener el más alto nivel de sensibilidad en longitudes de onda de entre 8 y 14 $[\mu\text{m}]$. A diferencia de otros equipos de detección de infrarrojos los microbolómetros no requieren refrigeración [17] [18].

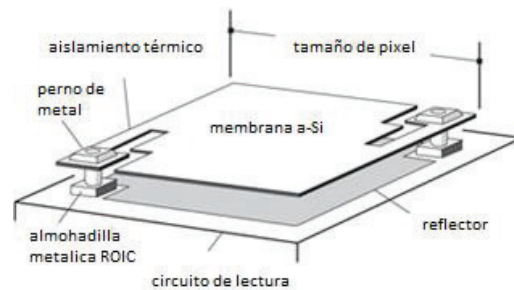


Figura 1.13 Estructura de un Microbolómetro de Silicio-Amorfo (a-Si) [20].

Un sensor LWIR no refrigerado con microbolómetro es un sensor térmico con un resistor en cada píxel. Los microbolómetros frecuentemente se fabrican con silicio amorfo (a-Si) u óxido de vanadio (Vox). El valor de temperatura en el resistor de cada píxel cambiará en función de la cantidad de radiación infrarroja incidente, esta variación mostrará los distintos tipos de color sobre cada píxel y así se generará una imagen visible al ojo humano [14].

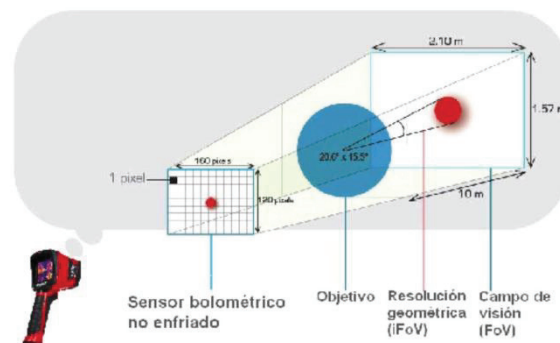


Figura 1.14 Cámara Térmica con microbolómetro no refrigerado [21].

1.6.3.2 Características Principales de las Cámaras Térmicas

El desempeño de una cámara térmica está determinado por ciertos requisitos esenciales de funcionamiento detallados a continuación:

1.6.3.2.1 Resolución de la Cámara

La resolución de la cámara indica el nivel de calidad que tendrá la imagen que se capture. Según el tipo de cámara se tiene resoluciones que van desde las más básicas de 60x60 píxeles teniendo 3600 puntos de medición, hasta modelos más avanzados de 640x480 píxeles que tienen 307200 puntos de medición sobre una imagen. Una alta resolución permite observar, medir e interpretar con mayor precisión [23].

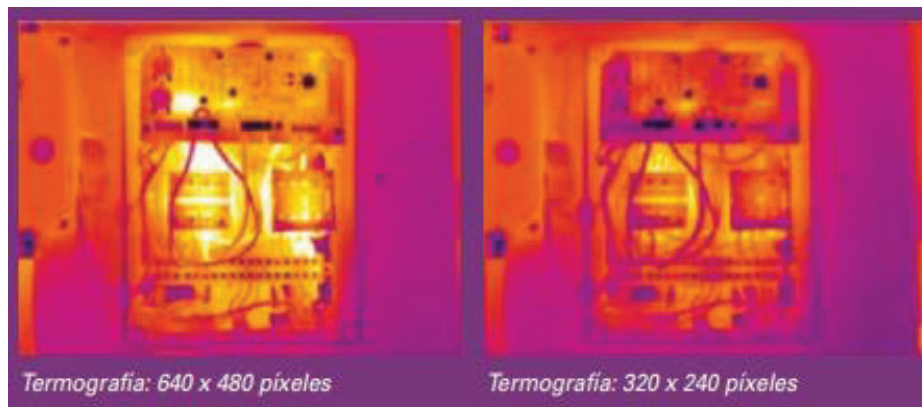


Figura 1.15 Resoluciones de cámara de 640x480 y 320x240 píxeles [23].

1.6.3.2.2 Sensibilidad Térmica

De acuerdo a la sensibilidad térmica se puede conocer la cantidad de diferencia de temperatura que podrá detectar una cámara. Mientras mejor es esta sensibilidad térmica, menor es el rango de diferenciación de temperatura que se podrá captar.

Por lo general, la sensibilidad térmica se mide en grados centígrados [°C] o también en milésimas de grado Kelvin [mK]. Las cámaras térmicas más avanzadas para mantenimiento predictivo poseen una sensibilidad térmica de 0,03 °C (30 mK) [23].

1.6.3.2.3 Precisión

La precisión es el margen de error en el cual opera la cámara térmica, las temperaturas pueden variar entre los valores reales y los valores medidos. La precisión normalmente viene expresada en porcentajes y en grados centígrados. El estándar para la precisión es de $\pm 2\%$ / ± 2 [°C]. Ciertas cámaras más avanzadas poseen valores de precisión de $\pm 1\%$ / ± 1 [°C] [23].

1.6.4 FACTORES DE INTERPRETACIÓN DE LAS IMÁGENES TÉRMICAS

Para interpretar los gráficos obtenidos por las cámaras térmicas correctamente, se necesita conocer los distintos materiales y las circunstancias que influyen en las lecturas de temperatura de la cámara térmica. Algunos de los factores más importantes son:

1.6.4.1 Conductividad Térmica

Se refiere a la velocidad con que se transmite cierta cantidad de calor a través de un material. Una alta conductividad térmica supone una transferencia de calor en mayor proporción que la de materiales con baja conductividad térmica.

Para la disipación térmica se utiliza materiales con alta conductividad térmica, mientras que como aislante térmico se usan materiales con baja conductividad térmica. Por ejemplo metales como el cobre poseen alta conductividad térmica que les permite tener una alta conductividad eléctrica [25].

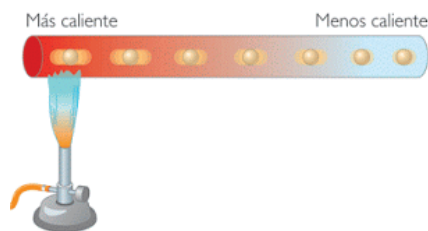


Figura 1.16 Conductividad térmica de un material [26]

1.6.4.2 Emisividad

Es la capacidad que posee un cuerpo para emitir radiación infrarroja dependiendo del tipo de material del cual este constituido. La calibración de

emisividad en una cámara térmica, permite obtener lecturas de temperatura con valores precisos.

Por ejemplo para configurar correctamente la emisividad de una cámara, se coloca un pedazo de cinta adhesiva con emisividad conocida sobre el material que se vaya a medir, para que luego de cierto tiempo la cinta adquiera la temperatura de dicho material y así determinar con precisión la temperatura exacta de la cinta [22].

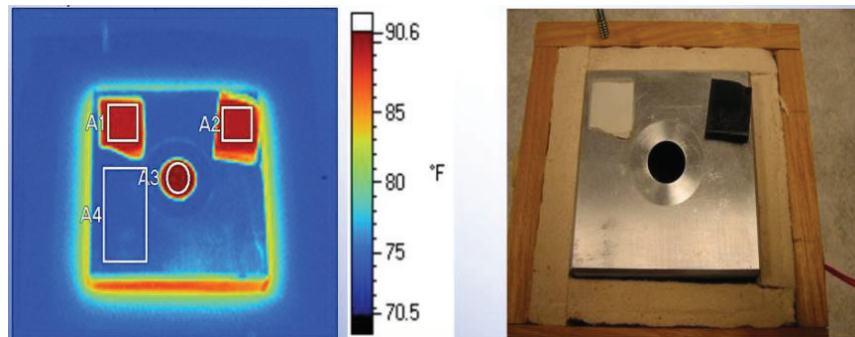


Figura 1.17 Medición de la emisividad de un objeto con ayuda de cinta adhesiva [28].

1.6.4.3 Temperatura reflejada

La temperatura reflejada se basa en las características que posee un material para reflejar la radiación infrarroja del mismo modo que un espejo reflejaría la luz visible. Los materiales que presentan esta condición son por ejemplo los metales no oxidados o también el vidrio. Estas reflexiones provocan confusión al momento de monitorear el ambiente con cámaras térmicas [22].



Figura 1.18 Temperatura reflejada sobre el vidrio de una ventana [22].

1.6.4.4 Temperatura ambiente

Una temperatura ambiente alta puede provocar que ciertos puntos calientes de un equipo no puedan ser apreciados claramente debido a la similitud de temperaturas, provocando confusiones de monitoreo [22].

1.6.5 APLICACIONES DE LA TERMOGRAFÍA

La termografía se puede aplicar en distintas áreas como por ejemplo el monitoreo de procesos eléctricos, inspección de edificios, inspección de cables de alimentación, hornos y calderas, motores recalentados, procesos médicos, etc [22].

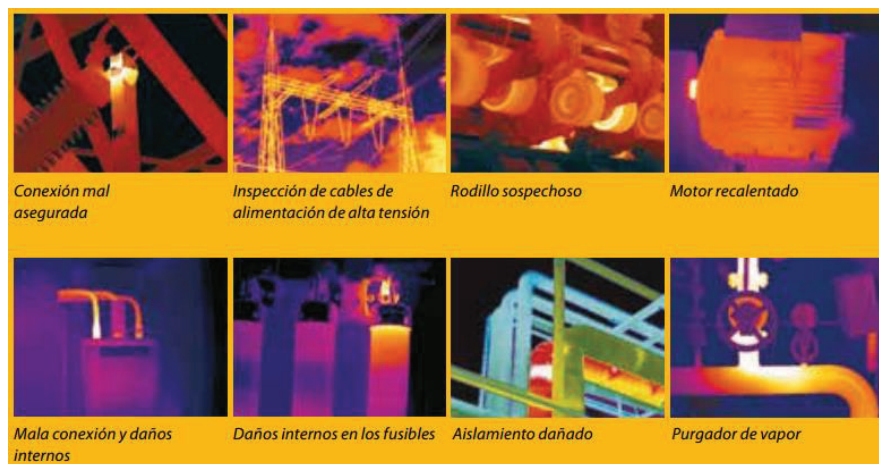


Figura 1.19 Mantenimiento predictivo en base a imágenes térmicas [23]

El detalle de los diferentes ámbitos en que se puede aplicar la termografía se muestra a continuación:

1.6.5.1 Sistemas Mecánicos

Algunas de las averías mecánicas que pueden detectarse en base a imágenes térmicas son:

- Problemas de lubricación.
- Errores de alineación.
- Motores recalentados.
- Rodillos sospechosos.

- Bombas sobrecargadas.
- Ejes de motor recalentados.
- Rodamientos calientes.

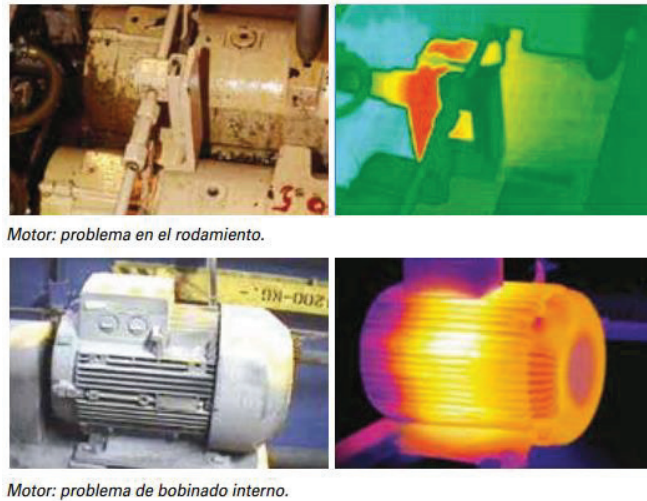


Figura 1.20 Aplicación de la termografía en motores [22].

1.6.5.2 Construcción

Al utilizar cámaras térmicas para el monitoreo de edificaciones, se puede solucionar el siguiente tipo de inconvenientes:

- Visualizar las pérdidas de energía de la edificación.
- Detectar un fallo o defecto en aislamientos.
- Localizar fugas de aire.
- Encontrar rastros de humedad en aislamientos, tejados y muros.
- Detectar moho y áreas mal aisladas.
- Localizar puentes térmicos.
- Localizar filtraciones de agua.
- Detectar roturas en tuberías de agua caliente.
- Detectar fallos de construcción.
- Encontrar averías en la calefacción.
- Detectar fallos eléctricos.

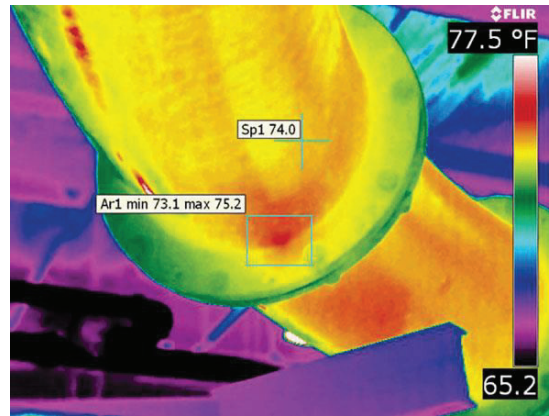


Figura 1.21 Termografía de una tubería [22]

1.6.5.3 Sistemas Eléctricos

La gran variedad de posibles aplicaciones de la termografía en sistemas eléctricos se divide en: instalaciones de alta tensión y de baja tensión.

1.6.5.3.1 Instalaciones de Alta Tensión

Algunos de los problemas que se pueden detectar con imágenes térmicas son:

- Oxidación de interruptores de alta tensión.
- Conexiones recalentadas.
- Conexiones mal aseguradas.
- Defectos de aislamiento.

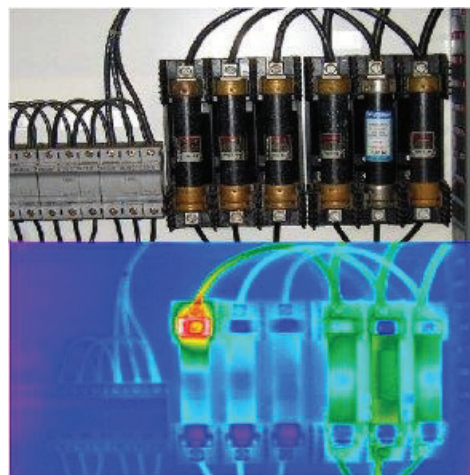
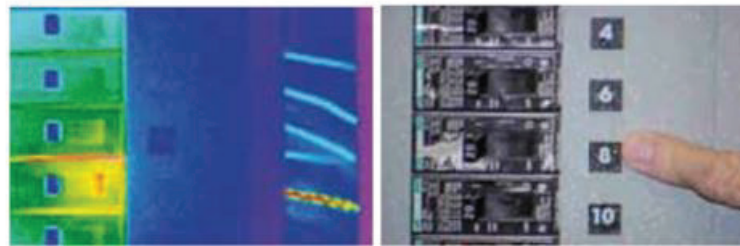


Figura 1.22 Termografía de una instalación eléctrica [15]

1.6.5.3.2 Instalaciones de Baja Tensión

Para este tipo de instalaciones eléctricas las anomalías que pueden solucionarse en base a imágenes térmicas son:

- Conectores recalentados.
- Problemas con fusibles sobrecargados.
- Evitar cortocircuitos.



El punto caliente indica un cortocircuito que podría provocar un incendio.

Figura 1.23 Termografía en instalaciones eléctricas de baja tensión [23].

1.6.5.4 Otras Aplicaciones

Los bomberos utilizan la visión infrarroja para poder ver a través del humo generado por las llamas, y de esta manera rescatar eficientemente a las personas atrapadas en un incendio.



Figura 1.24 Cámara Térmica utilizada por bomberos [29].

Además de esto se tiene varias aplicaciones como las detalladas a continuación:

- Detección de Nivel de Depósitos.
- Localización de puntos calientes en robots de soldadura.
- Inspección de material aeronáutico.
- Inspección de moldes.
- Comprobación de la distribución térmica en pavimentos asfálticos.
- Inspecciones en fábricas de papel.
- En seguridad para la detección de intrusos.
- En medicina para realizar procesos no invasivos.
- Localización de seres vivos.
- Detección de llamas invisibles a simple vista.

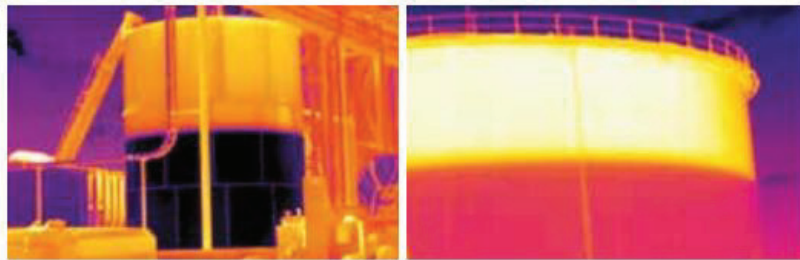


Figura 1.25 Detección de nivel de líquido en depósitos de almacenamiento [23].



Figura 1.26 Detección de llamas [23].

1.7 PLACA RASPBERRY PI

Raspberry Pi es un computador de placa reducida desarrollado el año 2011 en el Reino Unido por la Fundación Raspberry Pi, recibiendo apoyo del laboratorio de informática de la Universidad de Cambridge y del fabricante de circuitos integrados “Broadcom”.

Consumo	300 mA / 1,5W	400 mA / 2w	700 mA / 3,5 w	500 mA / 2,5w	800 mA / 4w	160 mA / 0.8w	2,5 A / 12,5 w	220 mA/ 1.1w
Tamaño	85,6 x 53,9 mm	65 x 56 mm	85,6 x 53,98 mm	85 x 56 mm	85 x 56 mm	65 x 31 mm	85x56 mm	66x30.5 mm

1.7.2 Ventajas de la Placa Raspberry Pi

La gran expansión de este tipo de minicomputadora se debe a las grandes posibilidades que presenta frente a otros dispositivos como un PC normal. A continuación se especifica las ventajas más relevantes [33]:

- Bajo consumo, aproximadamente consume 700mA (sin accesorios ni *overclocking*³).
- Bajo precio.
- Gran comunidad de usuarios y proyectos basados en esta placa.
- Accesorios de fácil disponibilidad.
- Tamaño reducido.
- Capacidad suficiente para observar video en HD debido al aprovechamiento de su GPU (*Graphics Processor Unit*, Unidad de Procesamiento Gráfico).
- Cuenta con varios tipos de programas del sistema Linux como: apache, samba, mysql, transmission, xbmc, etc.

1.7.3 Desventajas de la Placa Raspberry Pi

Como en cualquier dispositivo electrónico, también se presenta cierto tipo de inconvenientes, detallados a continuación:

- No es capaz de alimentar un disco duro por USB.
- Comparte un mismo bus para los puertos USB y la conexión LAN, lo que limita la velocidad de transferencia de datos.
- El procesador es insuficiente para trabajar con particiones de sistema de archivos NTFS (*New Technology File System*, Sistema de Archivos de Nueva Tecnología), típicos de Windows.

³ Overclocking, es la técnica de aumentar la velocidad de reloj de un dispositivo electrónico para que opere más rápido de lo que fue diseñado, lo que genera gastos mayores de energía.

- Se requiere de cierto conocimiento técnico para poder configurar el equipo.

1.8 SISTEMA OPERATIVO RASPBIAN

El sistema operativo Raspbian es la distribución más utilizada para la tarjeta Raspberry Pi. Raspbian se basa en la distribución Debian Wheezy (Debian 7.0) optimizando el código para su uso en Raspberry Pi. Esta distribución es la más completa y optimizada de las existentes, y por eso cuenta con apoyo oficial [34].

El sistema operativo Raspbian cuenta con un entorno de escritorio LXDE (*Lightweight X11 Desktop Environment*, Entorno de Escritorio X11 Ligero) y usa “Midori” como navegador web predeterminado [34].

Además incluye herramientas de desarrollo muy interesantes, como:

- Herramienta de programación Python.
- Editor de texto IDLE (*Integrated Development and Learning Environment*, Entorno de Desarrollo y Aprendizaje Integrado) para Python.
- Scratch para programar videojuegos.
- LibreOffice.
- Navegador Web (“Midori” como predeterminado).
- Calculadora.
- Visor de imágenes, PDF.

Para las últimas actualizaciones del sistema operativo se tiene un nuevo escritorio denominado PIXEL (Pi Improved Xwindows Environment Lightweight, Entorno de Escritorio Pi Mejorado Ligero). La mayoría de los cambios de este nuevo entorno se basan en el aspecto visual del sistema operativo, para hacerlo más amigable a quien lo utilice y se destaca la presencia de un nuevo navegador web denominado “Chromium” [35].

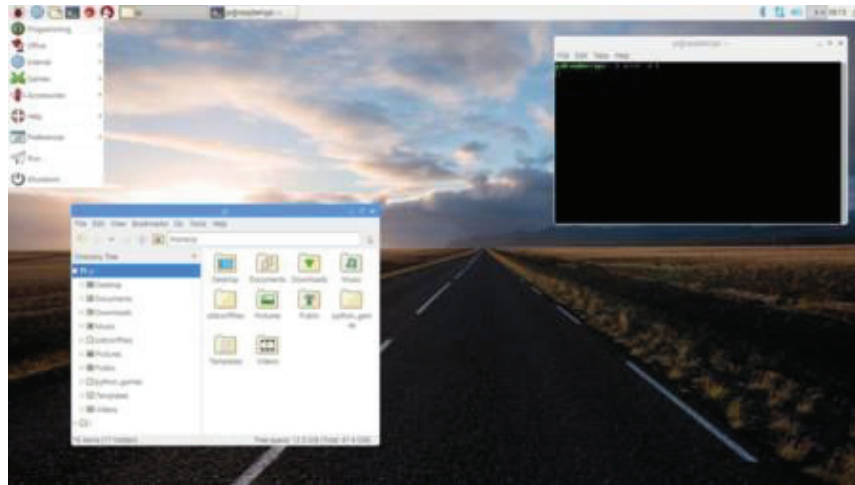


Figura 1.28 Entorno de trabajo de Raspbian [35].

1.9 TECNOLOGÍAS MÓVILES

La tecnología móvil o también conocida como telefonía celular es un tipo de comunicación inalámbrica concebida inicialmente para utilizarse solo en transmisiones de voz, pero con el paso del tiempo y el advenimiento de nuevas innovaciones se consigue transmitir nuevos servicios como datos, audio y video [36].

Los principales elementos de un sistema celular son la central telefónica fija, la central telefónica celular, las estaciones base y los terminales. Las estaciones base se comunican entre ellas y se conectan a la central telefónica celular, la cual también se encuentra conectada con la central telefónica fija. Según la ubicación en que se encuentre un usuario, su terminal se enlazará a la correspondiente estación base y de esta manera se logrará la comunicación.

El funcionamiento de los sistemas de telefonía celular se caracteriza por dividir en celdas o zonas el territorio en donde se brinda el servicio (área de cobertura), cada una de estas celdas es servida por una estación base, de modo que cuando el abonado celular se mueve con su terminal a través de la zona de cobertura y atraviesa varias celdas, no se pierde la comunicación [36].

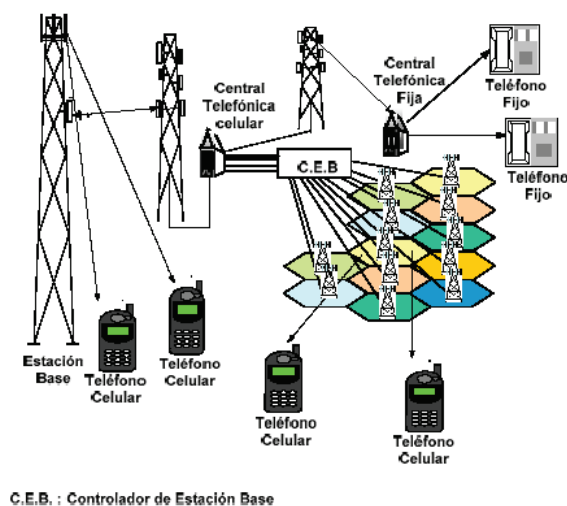


Figura 1.29 Estructura de un Sistema Celular [37]

La tecnología móvil a lo largo de la historia se ha ido actualizando y mejorando a través de varias generaciones de telefonía celular partiendo desde la primera generación (1G), hasta la más reciente conocida como cuarta generación (4G). Cada generación está comprendida por distintos estándares de comunicación celular que se diferencian principalmente por la velocidad de acceso que ofrecen. Se pueden distinguir esta serie de estándares en el gráfico a continuación:

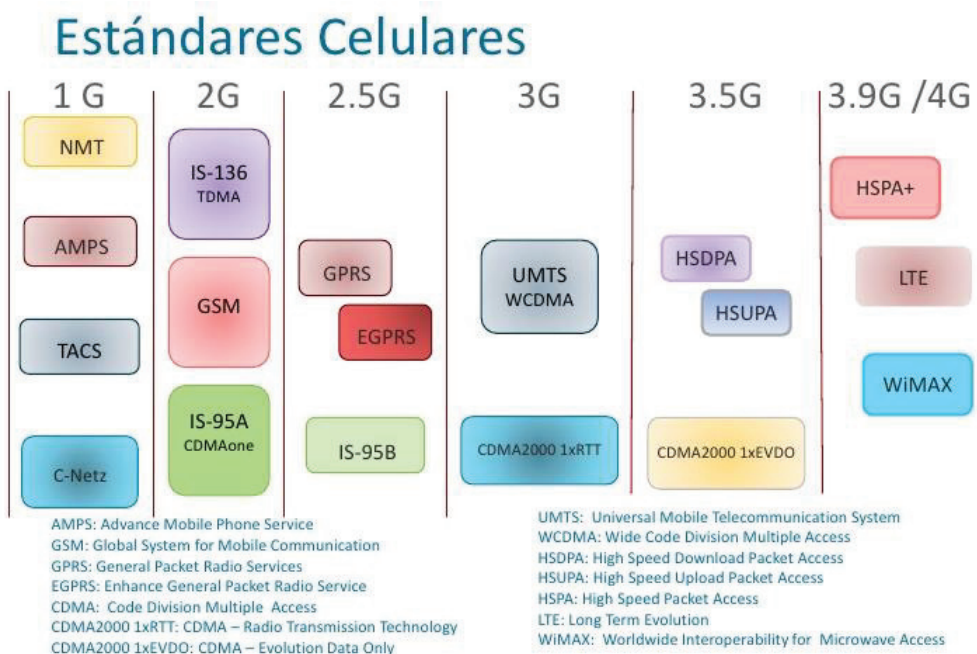


Figura 1.30 Estándares Celulares de Tecnología Móvil [38].

Las tecnologías móviles de interés para el presente proyecto se detallan a continuación:

1.9.1 TECNOLOGÍA MÓVIL 2.5G

Esta generación intermedia surge para establecer varias mejoras a los estándares iniciales de 2G, ya que se tenía cierto tipo de limitaciones para el envío de la información. Se requería que la red fuera más robusta para soportar el constante crecimiento de usuarios, debido a que la utilización de los computadores portátiles y del propio Internet se popularizó [39].

Se detalla a continuación los estándares celulares que surgieron durante esta generación:

1.9.1.1 GPRS (*General Packet Radio Service, Servicio General de Paquetes vía Radio*).

En la segunda generación se contaba con el sistema GSM (*Global System for Mobile communications, Sistema Global para las comunicaciones Móviles*) en el cual todos los circuitos eran dedicados en forma exclusiva a cada usuario, situación que cambió con GPRS al permitir que los usuarios compartan un mismo canal para el envío de los paquetes de información; con esto se aprovecha eficientemente la infraestructura y se abaratan los costos [39].

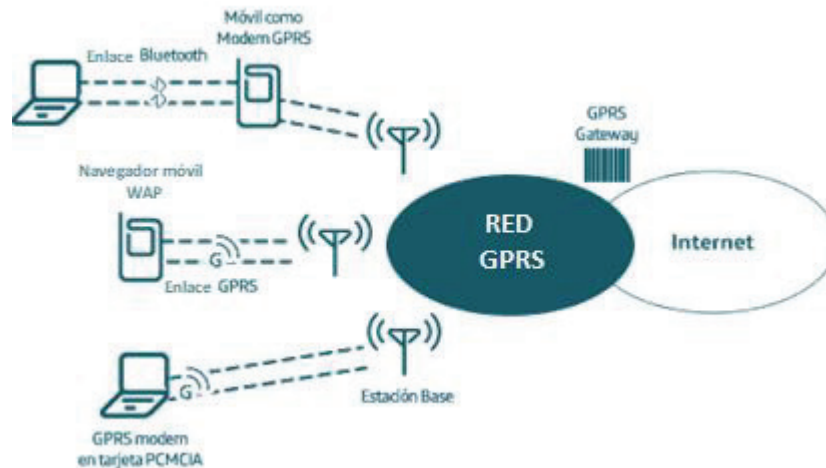


Figura 1.31 Red GPRS [42].

GPRS utiliza un canal o portadora GSM para la transmisión de datos con la técnica TDM (*Time Division Multiplexing*, Multiplexación por División de Tiempo) para dividir el canal en ocho ranuras de tiempo y así de esta forma aprovechar eficientemente el ancho de banda ofrecido por cada canal GSM [40].

El estándar GPRS permite velocidades de transferencia teóricas en un rango de entre 56 y 144 [Kbps], mientras que en la práctica esta velocidad es de 40 [Kbps] [40].

1.9.1.2 EDGE (*Enhanced Data rates for GSM Evolution*, Tasa de Datos Mejorada para la evolución de GSM).

También se conoce como EGPRS (*Enhanced GPRS*, GPRS mejorado), y como su nombre lo indica se caracteriza por mejorar aún más la transferencia de la información al introducir técnicas de modulación en frecuencia [39] [40].

En el estándar EDGE se utiliza la modulación GMSK (*Gaussian Minimum-Shift Keying*, Modulación por Desplazamiento Mínimo Gaussiano) y también la modulación 8-PSK (*8 Phase Shift Keying*, Modulación por Desplazamiento de ocho Fases) para aumentar la eficiencia de la transmisión [40].

EDGE podría considerarse como un servicio de tercera generación en vista que su tasa de transferencia supera los 144 [Kbps], pero varios autores tratan este estándar como un servicio 2.5G o 2.75G, debido a que estos rangos son en realidad varias veces más lentos que los implementados en una red 3G [39].

1.9.2 TECNOLOGÍA MÓVIL 3G

Los requerimientos para considerar que una tecnología sea de tercera generación consisten en que su tasa de transferencia sea de aproximadamente 2 [Mbps] para ambientes cerrados y de 384 [Kbps] para ambientes de tipo abierto [39].

La tecnología móvil 3G cuenta con los siguientes estándares:

1.9.2.1 UMTS (*Universal Mobile Telecommunications System*, Servicio Universal de Telecomunicaciones Móviles).

En el estándar UMTS se utiliza la tecnología de transmisión WCDMA (*Wideband Code Division Multiple Access*, Acceso Múltiple por División de Código de Banda Ancha) que permite obtener tasas de transmisión teóricas de 2 [Mbps], las cuales en la práctica tendrán valores de entre 144 y 512 [Kbps] [40].

UMTS permite acelerar la navegación en Internet, al contar con mejores velocidades de transmisión que anteriores estándares y por tanto surgen los primeros teléfonos que soportan este tipo de tecnología 3G. Además aparecen equipos conocidos como “módems 3G” los cuales por medio de un conector USB permiten el acceso a la red celular a dispositivos diferentes a los teléfonos celulares [40].

1.9.2.2 HSDPA (*High Speed Downlink Packet Access*, Acceso Descendente de Paquetes a Alta Velocidad)

Se conoce también como 3.5G o 3G+ y se caracteriza por haber mejorado la tecnología UMTS/WCDMA, llegando a soportar tasas de transmisión teóricas de hasta 14 [Mbps] para el enlace de descarga [40].

Este estándar permite el acceso simultáneo de una gran cantidad de usuarios al contar con técnicas como la codificación de errores, redundancia incremental y la modulación QAM (*Quadrature Amplitude Modulation*, Modulación de Amplitud en Cuadratura), también se logra disminuir la latencia [40].

1.9.2.3 HSUPA (*High Speed Uplink Packet Access*, Acceso Ascendente de Paquetes a Alta Velocidad).

Se conoce también como 3.75G y al igual que el estándar HSDPA se caracteriza por mejorar la tecnología UMTS/WCDMA, llegando a proporcionar velocidades de transmisión teóricas de hasta 5.76 [Mbps], pero en este caso para el enlace de subida [41].

1.9.2.4 HSPA+ (*Evolved HSPA*, HSPA Evolucionado)

En este estándar se mejora aún más las velocidades de transferencia al utilizar técnicas como la modulación 64-QAM y la transmisión de datos MIMO (*Multiple-Input Multiple-Output*, Múltiples Entradas con Múltiples Salidas) que se caracteriza por emplear varias antenas para transmitir. El rango de velocidades teóricas es de 84 [Mbps] para el enlace de descarga y 22 [Mbps] para el enlace de subida [40].

CAPÍTULO 2: DISEÑO E IMPLEMENTACIÓN DEL SISTEMA

En el presente capítulo se expondrá todas las pautas necesarias para conseguir realizar el diseño del prototipo de cámara térmica y de esta manera lograr desarrollar la respectiva implementación de todas las partes que componen el sistema. Primeramente se detallarán las características y especificaciones técnicas que poseen los componentes del prototipo planteado.

Para un mejor entendimiento del diseño del prototipo se divide en dos sistemas principales:

1. Sistema de Captura y Visualización de Imágenes Térmicas.
2. Sistema de Transmisión de Datos hacia Internet.

2.1 SISTEMA DE CAPTURA Y VISUALIZACIÓN DE IMÁGENES TÉRMICAS

Este sistema está destinado a lograr obtener las imágenes térmicas con ayuda del sensor de cámara infrarroja para poder desplegarlas sobre una interfaz gráfica en la pantalla táctil y en una determinada escala de colores.

Los elementos que conforman este primer sistema son los detallados a continuación:

- Placa Raspberry Pi 2B.
- Sensor de Cámara Térmica.
- Pantalla.
- Batería.
- Software.
- Carcasa.

2.1.1 PLACA RASPBERRY PI 2B

La placa Raspberry Pi 2B es una mini computadora o computador de placa reducida que pertenece a la segunda generación de los modelos de Raspberry Pi. Esta placa surge en Febrero de 2015 como un modelo mejorado de la inicial Raspberry Pi 2 [43].

La placa Raspberry Pi 2B es una parte importante dentro del sistema, la cual se encargará de procesar los datos y controlar la gran mayoría de los elementos que conforman el prototipo de cámara térmica.

El hecho de escoger el modelo 2B de Raspberry Pi, se debe a la compatibilidad que dicha placa posee con la pantalla táctil utilizada para la manipulación de la interfaz gráfica del prototipo.

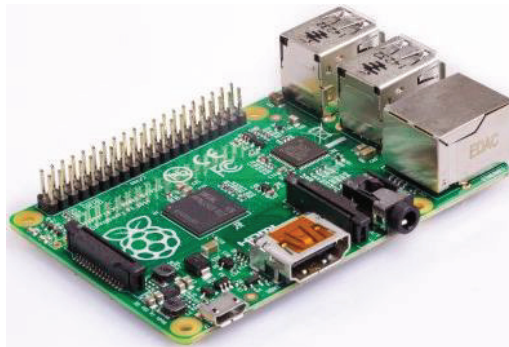


Figura 2.1 Placa Raspberry Pi 2B [44].

Las características con las que cuenta el modelo de Placa Raspberry Pi 2B son las siguientes:

- Procesador de cuatro núcleos Broadcom BCM2836 ARM Cortex-A7 con 900 MHz de velocidad.
- Memoria RAM de 1GB.
- Unidad de Procesamiento Gráfico (GPU) Videocore IV de doble núcleo a 250 MHz.
- GPIO extendido de 40 pines.
- 4 puertos USB 2.0.
- Salida de audio estéreo de 4 polos y puerto de vídeo compuesto.

- Puerto Full HDMI.
- Puerto de cámara CSI para conectar una cámara Raspberry Pi.
- Puerto de pantalla DSI para conectar una pantalla táctil Raspberry Pi.
- Puerto Micro SD para cargar el sistema operativo y almacenar datos.
- Fuente de alimentación conmutada Micro USB 1.8 A.
- Puerto Ethernet 10/100.
- Dimensiones: 85 x 56 x 17 mm

2.1.1.1 Pines GPIO

Los pines GPIO (*General Purpose Input/Output*, Entrada/Salida de Propósito General), son una interfaz física entre la Raspberry Pi y el mundo exterior. El funcionamiento de cada uno de estos pines se asemeja al de un interruptor que se puede activar o desactivar. La distribución de los 40 pines es la siguiente:

- 26 pines para GPIO.
- 8 para conexión de tierra (ground).
- 2 para conexiones eléctricas de 3.3 v.
- 2 para conexiones eléctrica de 5 v.
- 2 para conexiones ID EEPROM (solo para uso avanzado).

En el grafico a continuación se detalla los pines GPIO:

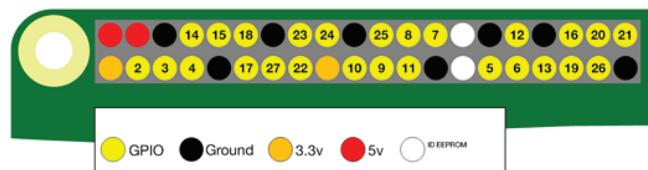


Figura 2.2 Distribución de pines GPIO [43]

A continuación se presenta la distribución de los pines GPIO indicando su numeración física ("Pin#"), su numeración de chip Broadcom (modo BCM), se detalla la función que desempeña cada pin, y se muestra su disposición tanto para los primeros modelos de 26 pines, como para los recientes modelos de 40 pines:

Pin#	NOMBRE		NOMBRE	Pin#
01	3.3v DC Power	●	DC Power 5v	02
03	GPIO02 (SDA1 , PC)	●	DC Power 5v	04
05	GPIO03 (SCL1 , PC)	●	Ground	06
07	GPIO04 (GPIO_GCLK)	●	(TXD0) GPIO14	08
09	Ground	●	(RXD0) GPIO15	10
11	GPIO17 (GPIO_GEN0)	●	(GPIO_GEN1) GPIO18	12
13	GPIO27 (GPIO_GEN2)	●	Ground	14
15	GPIO22 (GPIO_GEN3)	●	(GPIO_GEN4) GPIO23	16
17	3.3v DC Power	●	(GPIO_GEN5) GPIO24	18
19	GPIO10 (SPI_MOSI)	●	Ground	20
21	GPIO09 (SPI_MISO)	●	(GPIO_GEN6) GPIO25	22
23	GPIO11 (SPI_CLK)	●	(SPI_CE0_N) GPIO08	24
25	Ground	●	(SPI_CE1_N) GPIO07	26
27	ID_SD (PC ID EEPROM)	●	(PC ID EEPROM) ID_SC	28
29	GPIO05	●	Ground	30
31	GPIO06	●	GPIO12	32
33	GPIO13	●	Ground	34
35	GPIO19	●	GPIO16	36
37	GPIO26	●	GPIO20	38
39	Ground	●	GPIO21	40

Primeros Modelos

Últimos Modelos

Rev. 1
26/01/2014

<http://www.element14.com>

Figura 2.3 Descripción de Pines GPIO [47]

2.1.1.1.1 Interfaz I2C (*Inter-Integrated Circuit, Circuito Interintegrado*)

Los pines I2C permiten a la Raspberry Pi comunicarse y controlar a distintos tipos de periféricos externos. I2C es un protocolo de comunicación serial y síncrono, es decir, la información es enviada bit a bit coordinadamente y de manera sincronizada [48].

Las señales que componen esta interfaz son las siguientes:

- SCL (“*System Clock*”): Es la línea por donde se envía la señal de reloj para sincronizar todo el sistema.
- SDA (“*System Data*”): Es la línea para transferir datos entre los dispositivos.

2.1.1.1.2 Interfaz SPI (*Serial Peripheral Interface, Interfaz Periférica Serial*)

Es conocido como el bus serie de cuatro hilos. SPI permite la transferencia de información de control entre circuitos integrados por medio de un flujo de bits serie regulado por una señal de reloj. A diferencia de I2C se utiliza más pines pero con una velocidad de transferencia superior [48].

Las 4 señales que conforman esta interfaz son las siguientes:

- MOSI (*“Master Output Slave Input”*): Salida de datos para el maestro, entrada de datos para el esclavo.
- MISO (*“Master Input Slave Output”*): Entrada de datos para el maestro, salida de datos para el esclavo.
- SCLK (*“Serial Clock”*): Señal de reloj para sincronizar el resto de señales.
- CE (*“Chip Enable”*): También se conoce como *“chip select”* y sirve para elegir una o varias señales de selección de destino (*CE0* y *CE1*).

2.1.1.1.3 Interfaz UART (*Universal Asynchronous Receiver/Transmisor, Transmisor-Receptor Asíncrono Universal*)

UART es una interfaz de comunicación serie, utilizada para la transmisión de dato por dato entre dispositivos electrónicos sin la necesidad de una señal de reloj [48].

Las señales que presenta esta interfaz son las siguientes:

- TXD (*“Transmission Data”*): Para la transmisión de datos.
- RXD (*“Reception Data”*): Para la recepción de datos.

2.1.2 SENSOR DE CÁMARA TÉRMICA

La cámara térmica utilizada para el presente proyecto es desarrollada por la marca *“FLIR”*, su modelo es el *“Flir Lepton LWIR”*. Se dispone de un módulo de

circuito electrónico que contiene al sensor de cámara térmica y es compatible con la Placa Raspberry Pi.

La cámara térmica Flir Lepton LWIR permite captar las ondas de radiación infrarroja del ambiente con longitud de onda de entre 8 y 14 micras, para poder transformarlas en ondas de luz visible, expuestas en una determinada gama de colores. La resolución de la cámara térmica es de 80×60 píxeles. La información que brinda el sensor Flir Lepton LWIR se capta por medio de los puertos SPI e I2C de la interfaz GPIO de la Raspberry Pi [47].

2.1.2.1 Aplicaciones

Los principales usos que se pueden dar a este tipo de cámara térmica son los siguientes:

- Para captura de imágenes térmicas.
- Sirve como un sensor de movimiento.
- Para visión nocturna.
- Reconocimiento de gestos.

2.1.2.2 Distribución de Pines

En el gráfico a continuación se muestra la distribución de pines del módulo de cámara térmica, el cual se conecta con la Raspberry Pi 2B a través de la GPIO.

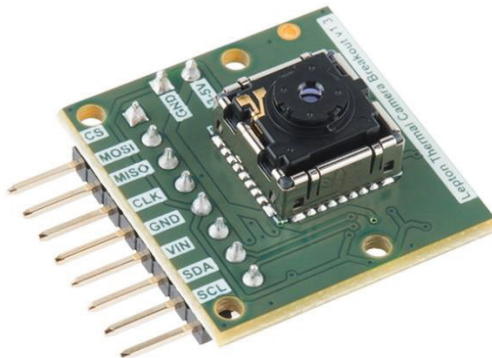


Figura 2.4 Cámara Térmica Flir Lepton LWIR [47]

La interfaz GPIO y la función que desempeña cada pin se muestra a continuación:

1. CS, *Chip Select* (Interfaz SPI).
2. MOSI, envío de datos (Interfaz SPI).
3. MISO, envío de datos (Interfaz SPI).
4. CLK, señal de reloj (Interfaz SPI).
5. GND, tierra.
6. VIN, entrada de alimentación.
7. SDA, datos de interfaz de control de cámara. (Interfaz I2C).
8. SCL, reloj de interfaz de control de cámara. (Interfaz I2C).

2.1.2.3 Especificaciones Técnicas

Las principales especificaciones técnicas que posee el módulo de cámara térmica Flir Lepton LWIR son las siguientes:

- Sensor LWIR (Infrarrojo de Onda Larga), longitud de onda de 8 a 14 μm .
- Campo de Visión Horizontal (HFOV, *Horizontal Field of View*) de 51 grados; diagonal de 63,5 grados (f/1.1 doblete de silicio).
- Resolución de 80 (h) \times 60 (v) píxeles activos.
- Sensibilidad térmica <50 mK.
- Interfaces de vídeo MIPI y SPI.
- Interfaz serial de control I2C de dos hilos.
- Tiempo rápido de imagen (<0,5 segundos).
- Potencia de funcionamiento 150 mW (<160 mW en toda la gama de temperaturas).
- Voltaje de entrada: 3 V a 5.5 V.
- Dimensiones: 8.5 x 11.7 x 5.6 mm (sin zócalo); 10.6 x 11.7 x 5.9 mm (con el zócalo).
- Velocidad de fotograma (< 9 Hz).
- Utiliza fuentes de alimentación compatibles con teléfonos celulares: 2.8 V al sensor, 1.2 V a la base digital e IO flexible de 2.5 V a 3.1 V.

2.1.3 PANTALLA

La pantalla que se utilizará para el prototipo de cámara térmica es de tipo táctil. La función táctil de la pantalla es importante ya que se necesita facilidad para maniobrar la interfaz gráfica de la cámara térmica; esta función es útil para poder realizar la captura de imágenes térmicas eficientemente y también para que se consiga enviar dichas imágenes por medio de esta interfaz hacia un servidor web.

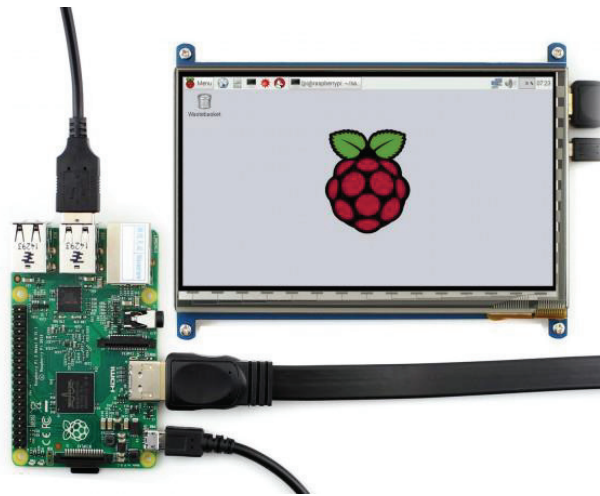


Figura 2.5 Pantalla Táctil conectada con la Raspberry Pi [51].

La pantalla que se utiliza para el prototipo es de cristal líquido, su tamaño es de 5 pulgadas. El modelo en específico es el “5inch HDMI LCD (B)”, la letra “B” indica la resolución de 800x480 pixeles que posee la pantalla. Soporta varios sistemas operativos, y en especial está diseñada para operar con el sistema operativo de la Raspberry Pi (“Raspbian”) [51].

Para la comunicación entre la Raspberry Pi 2B y la Pantalla se trabaja con dos canales de información. El primer canal sirve para el envío de la información de imagen y se realiza a través de un cable HDMI. El segundo canal se utiliza para las funciones de control táctil y de alimentación eléctrica, para esto se utiliza un cable con conector micro USB en la pantalla y conector USB en la Raspberry Pi 2B [51].

2.1.3.1 Especificaciones Técnicas

Las características técnicas que ofrece la pantalla táctil de cristal líquido son las siguientes:

- Resolución de 800×480 pixeles.
- Control táctil resistivo.
- Dimensiones: 121x89 mm.
- Interfaz HDMI para mostrar imagen.
- Interfaz USB para control táctil.
- Interruptor on/off para reducir el consumo de energía.
- Soporta la plataforma Raspberry Pi (funciona directamente con Raspbian).
- Soporta la plataforma Beaglebone Black, funcionando con Angstrom, una distribución basada en Debian.
- Soporta Banana Pi / Banana Pro, funcionando con distribuciones como: Lubuntu, Raspbian.
- Puede trabajar como un monitor de una computadora con Windows, funcionando como cualquier otra pantalla general de HDMI.
- Funciona con los sistemas operativos Windows 10 / 8.1 / 8 / 7.

2.1.3.2 Función IPS

La pantalla que se utiliza para el proyecto posee la característica IPS (*In-Plane Switching*, Cambio en el Plano), es un tipo de panel LCD con disposición interna de los cristales líquidos que evita las fugas y pérdidas de luz, y así no se degrada la calidad de imagen. La función IPS permite observar las imágenes en pantalla hasta en ángulos de 180° [53] [54].

Las ventajas que ofrece IPS son:

- Mejor reproducción del color.
- Mayor nitidez.
- Mejor ángulo de visión.
- Color blanco más intenso.

- Mejor tiempo de respuesta.

Y algunas desventajas que se presentan con IPS son las siguientes:

- Necesita una retroiluminación fuerte.
- Consume más energía.
- Provoca terminales más gruesos.

2.1.4 BATERÍA

La fuente de alimentación que se utilizará para el prototipo de cámara térmica es una batería externa o también conocida como batería portátil.

La razón para elegir este tipo de batería se debe a su tamaño reducido y por tanto representa facilidad para ser ensamblada junto con el resto de componentes del sistema.



Figura 2.6 Batería Externa [55]

Se necesita que la batería cumpla con los requerimientos energéticos que se recomiendan para la placa Raspberry Pi 2B, lo cual supone un voltaje de 5 [V] y un amperaje de 1.8 [A]. También se debe considerar el resto de periféricos que estarán conectados a la placa principal, es decir, la cámara térmica, la pantalla táctil y el módulo para la conexión móvil celular.

2.1.5 SOFTWARE

Para el funcionamiento de la placa principal, que en este caso es la Raspberry Pi 2B, el sistema operativo que se utiliza es “*Raspbian*”; dicho sistema está compuesto de aproximadamente 35 mil paquetes precompilados, para ser utilizados durante la elaboración de un proyecto [56] [57].

Sobre el sistema operativo Raspbian se necesitará instalar varios tipos de programas que servirán para el funcionamiento del prototipo de cámara térmica, entre ellos se destacan programas como “Qt Creator” que servirá para el diseño de la Interfaz Gráfica de Usuario que controlará el sistema.

También se deberá realizar cierto tipo de configuraciones para el buen funcionamiento y control del sensor de cámara térmica, de la pantalla táctil y del módem USB. Todo este tipo de parámetros serán detallados más adelante en cada sección correspondiente del documento.

2.1.5.1 Sistema Operativo Raspbian

Se debe destacar que Raspbian no está afiliado de ninguna manera con la Fundación Raspberry Pi, fue creado por un pequeño grupo de desarrolladores fanáticos de la Raspberry Pi, interesados en las metas educativas que puede ofrecer esta mini computadora y con conocimientos altos de la distribución “*Debian*” [57].

El paquete del sistema operativo se puede descargar de la página oficial de la organización Raspberry Pi, el enlace directo a esta descarga es el siguiente: “<https://www.raspberrypi.org/downloads/>”. En este sitio se encuentran dos formas de descarga del sistema operativo Raspbian.

La primera forma permite descargar el paquete Raspbian preinstalado con software para la educación, la programación y el uso general. Posee los programas Python, Scratch, Sonic Pi, Java, Mathematica y muchos más.



Figura 2.7 Logo del sistema operativo Raspbian [56]

La segunda forma se denomina “NOOBS” (*New Out Of the Box Software*), es un sencillo instalador del sistema operativo Raspbian para principiantes. Luego de descargar NOOBS, la propia página oficial de Raspberry posee una guía detallada sobre su instalación.

NOOBS además de contener Raspbian, también proporciona una selección de sistemas operativos alternativos que pueden ser descargados e instalados desde Internet.

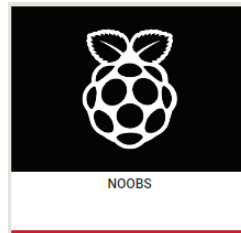


Figura 2.8 NOOBS, la alternativa de instalación de Raspbian [56].

Además del sistema operativo oficial de la Raspberry Pi (Raspbian), se tiene la posibilidad de instalar otros sistemas operativos como:

- UBUNTU MATE - Escritorio de Ubuntu.
- SNAPPY UBUNTU CORE - Para desarrolladores.
- WINDOWS 10 IoT CORE - Para el Internet de las Cosas.
- OSMC - Open Source Media Centre.
- LIBREELEC - Un sistema operativo para Kodi.
- PINET - Solución para gestiones de salas de clase.
- RISC OS - Una distribución que no pertenece a Linux.
- WEATHER STATION - Una función de estación meteorológica de Oracle para Raspberry Pi.

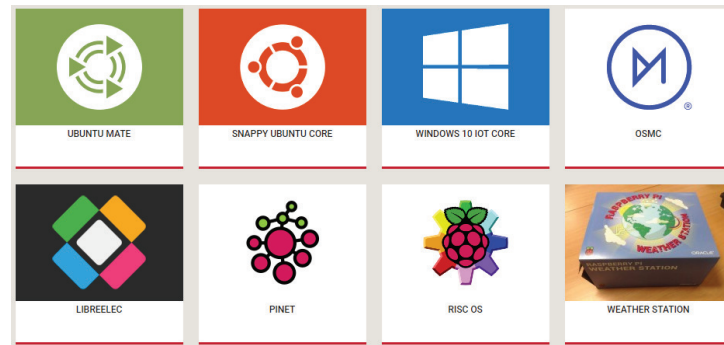


Figura 2.9 Sistemas Operativos alternativos para Raspberry Pi [56]

2.1.5.2 Qt Creator

Qt Creator es un IDE (*Integrated Development Environment*, Entorno de Desarrollo Integrado) multiplataforma que sirve para el desarrollo de aplicaciones con Interfaces Gráficas de Usuario. Qt Creator es parte del “Qt Project”, un proyecto para coordinar el desarrollo de software de Qt [69].

El proyecto Qt fue fundado inicialmente por NOKIA, para luego ser adquirido por la empresa DIGIA creando “The Qt Company”, y así quedándose con los negocios y tecnologías de software de Qt desde el año 2012 [70].

Qt Creator utiliza de forma nativa el lenguaje de programación C++, pero también puede ser utilizado con otros lenguajes de programación como Python. Qt Creator se centra en proporcionar características que ayudan a los nuevos usuarios de Qt a aprender y comenzar a desarrollar rápidamente aplicaciones.

Entre las características de Qt Creator se destacan las siguientes:

- Editor de código con soporte para C+, *QML* y *ECMAScript*.
- Herramientas para la rápida navegación del código.
- Resaltado de sintaxis y auto-completado de código.
- Control estático de código y estilo a medida que se escribe.
- Soporte para refactorización del código.
- Ayuda sensitiva al contexto.
- Plegado de código.
- Paréntesis coincidentes y modos de selección.

2.1.6 CARCASA

Las partes que componen el prototipo de cámara térmica se planifica incorporarlas dentro de una carcasa hecha a base de acrílico. Este material es fuerte y resistente a los impactos.

El tamaño de la carcasa dependerá directamente de las dimensiones que presentan las partes del prototipo de cámara térmica y además de la manera en cómo se conectan cada una de estas. La carcasa también debe permitir facilidad para ser ensamblada, o en caso contrario poder desensamblarse rápidamente para cualquier tipo de mantenimiento.

El material acrílico se asemeja al plástico, posee de diez a veinte veces mejor resistencia que el vidrio. Es resistente a la intemperie y a los rayos ultravioleta, es decir, su estructura perdura con el pasar del tiempo. Es un excelente aislante térmico. Este material presenta la cualidad de ser liviano, lo que es una gran ventaja para la maniobrabilidad del prototipo.

2.2 SISTEMA DE TRANSMISIÓN DE DATOS HACIA INTERNET

Una vez que ya se obtenga la imagen térmica y pueda visualizarse en la pantalla, se procede con la ejecución del siguiente sistema que conforma el proyecto de cámara térmica. Este sistema se basa en adquirir la información de la imagen térmica para poder enviarla hacia un servidor web utilizando la red celular, para esto se utiliza como dispositivo inalámbrico un módem de conexión USB.

En este punto se debe detallar la forma en cómo serán enviados los datos y que protocolo de transferencia de archivos se utilizará para su respectivo envío hacia la nube de internet. Posteriormente se debe establecer la aplicación web en donde se receptorá este tipo de datos, y en donde se alojarán las imágenes térmicas que se capturen con el prototipo de cámara térmica.

Este sistema está compuesto por los siguientes elementos:

- Módem USB.
- Protocolo de Transferencia de Archivos.
- Computador.
- Servidor Alojado en la Nube.
- Aplicación Web.

2.2.1 MÓDEM USB

El equipo utilizado para enviar los datos de las imágenes térmicas de manera inalámbrica desde el prototipo hacia el servidor web se conoce como módem de acceso a Internet móvil. El módem está fabricado por la empresa “*Huawei*”, y el modelo a utilizarse en el presente proyecto es el E303.

La razón de elegir este tipo de módem se debe a que ya se contaba previamente con el equipo, el cual se encontraba en perfecto estado de funcionamiento, por tanto se aprovechó para ahorrar en gastos de elaboración del prototipo.

El tamaño del módem también influye para optar por este tipo de dispositivo y además su forma de conexión con la placa Raspberry Pi 2B es por medio de un puerto USB, lo que significa que se deja libre la interfaz GPIO para poder ser utilizada en la conexión del sensor de cámara térmica.

Otra de las razones por las cuales se utiliza este tipo de modelo (Huawei E303), se debe a que permite operar con el sistema operativo Raspbian, después de realizar cierto tipo de configuraciones.



Figura 2.10 Módem Huawei E303 [58].

2.2.1.1 Especificaciones Técnicas

Las características que presenta el módem Huawei E303 son detalladas a continuación [59].

- Opera bajo los estándares celulares:
 - HSUPA/HSDPA/UMTS trabajando con bandas de frecuencia de 2100 MHz o 2100/900MHz o también 2100/1900/850 MHz.
 - EDGE/GPRS/GSM trabajando con bandas de frecuencia de 1900/1800/900/850 MHz.
- Interfaz USB estándar (Tipo A).
- Ranura para tarjeta micro SD.
- Capacidad de memoria micro SD de hasta 32 GB.
- Ranura para tarjeta SIM.
- Ecualizador HSPA.
- Receptor de diversidad (opcional).
- Servicio de Downlink HSDPA de hasta 7.2 Mbps.
- Servicio de Uplink HSUPA de hasta 5.76 Mbps.
- Servicio de Downlink UMTS PS (*Packet Switched* o Conmutación de paquetes) de hasta 384 Kbps.
- Servicio de Downlink EDGE de hasta 236.8 Kbps.
- Servicio de Uplink EDGE de hasta 236.8 Kbps (opcional).
- Servicio de Downlink GPRS de hasta 85.6 Kbps.
- Tecnología Plug and Play.
- Servicio de SMS basado en CS/PS (*Circuit Switched / Packet Switched* o Conmutación de Circuitos / Conmutación de Paquetes) dominio de GSM y WCDMA.
- Servicio de USSD (*Unstructured Supplementary Service Data* o Servicio Suplementario de Datos no Estructurados. Es un servicio semejante a SMS).
- Driver del módem NDIS (*Network Driver Interface Specification* o Especificación de Interfaz de Controlador de Red).
- Disponible para trabajar con sistemas operativos:

- Windows XP SP2/SP3.
- Windows Vista SP1/SP2.
- Windows 7.
- Mac OS 10.5, 10.6 y 10.7 con las últimas actualizaciones.
- Linux.
- Tamaño:
 - Largo: 84.5 mm
 - Ancho: 27.0 mm
 - Grosor: 12.3 mm
- Peso menor a 30 gramos.

2.2.2 PROTOCOLO DE TRANSFERENCIA DE ARCHIVOS

El protocolo que se utilizará para la transferencia de archivos desde el prototipo hacia el servidor web es SCP (*Secure Copy Protocol*, Protocolo de Copia Segura), el cual se caracteriza por permitir la transferencia segura de archivos entre un *host* local y un *host* remoto, o también entre dos *host* remotos, por medio de la utilización del protocolo SSH⁴ (*Secure Shell*, Intérprete de Órdenes Seguro).

Los datos son cifrados durante la transferencia, pero cabe indicar que SCP por sí solo no posee autenticación ni seguridad, estas funciones son proporcionadas por SSH [64].

La estructura del comando para poder enviar un archivo desde un *host* local hacia un *host* remoto es la siguiente: ["scp"] + ["directorio local del archivo"] + ["usuario del servidor web" @ "directorio remoto del archivo"].

2.2.3 COMPUTADOR

Cuando las imágenes térmicas se envían al servidor web, se encuentran alojadas en un directorio remoto, al cual es posible acceder por medio de la correspondiente dirección IP del servidor, es decir, cualquier dispositivo con

⁴ SSH, es un protocolo que facilita las comunicaciones seguras entre dos sistemas usando una arquitectura cliente/servidor.

conexión a Internet tendrá la posibilidad de visualizar dichos archivos de imagen térmica.

En el presente proyecto se utiliza un computador portátil para visualizar las imágenes térmicas en la aplicación web. El fabricante del equipo es “ASUSTeK Computer Inc.” y su modelo en específico es el X455LB.



Figura 2.11 Computador Portátil ASUS X455LB [60].

2.2.3.1 Especificaciones Técnicas

Algunas de las características principales que presenta el computador portátil son detalladas a continuación:

- Procesador Intel® Core™ i7-5500U CPU 2.40 GHz.
- Memoria RAM de 8GB.
- Sistema Operativo Windows 10 Pro de 64 bits.
- Disco Duro de 1 TB de capacidad.
- Pantalla de 14.0" (35.56 cm), resolución 16:9 HD (1366x768).
- Bluetooth™ Integrado V4.0.
- Conectividad Wi-Fi 802.11n.

2.2.4 SERVIDOR ALOJADO EN LA NUBE

Las imágenes térmicas capturadas por el prototipo se alojarán en un servidor web, por lo tanto, es necesario contratar dicho servicio y realizar las respectivas configuraciones para su montaje.

En el mercado existe una gran variedad de empresas que ofrecen este tipo de servicios, cada una con sus respectivas ventajas y desventajas. Para el presente proyecto se opta por contratar los servicios de la plataforma de computación en la nube de la empresa “Amazon”, conocida específicamente como AWS (*Amazon Web Services*, Servicios Web de Amazon).

La principal razón de elegir esta plataforma se debe a las facilidades de pago que permite, ya que solo se necesita una tarjeta de crédito para acceder a su capa gratuita durante 12 meses. Con este tipo de cuenta se tiene acceso a los servicios básicos de la plataforma, los cuales son suficientes para el presente proyecto de cámara térmica.

2.2.5 APLICACIÓN WEB

Para poder observar las imágenes térmicas en Internet se necesita el establecimiento de una aplicación web que permita su respectiva visualización y procesamiento.

La aplicación se configura en el servidor web contratado previamente, para poder mostrar los archivos de imagen enviados por el prototipo de cámara térmica, dicha aplicación se podrá actualizar cada vez que se realice nuevas capturas de imágenes térmicas.

Para la elaboración del diseño y configuración de la aplicación web se empleará lenguajes de programación como HTML, CSS, PHP, JavaScript, etc.

2.3 IMPLEMENTACIÓN DEL PROTOTIPO DE CÁMARA TÉRMICA

Se procede con la implementación de todo el sistema después de haber tratado en detalle las especificaciones técnicas de cada componente del prototipo, así como el distinto software que se utilizará para su respectiva programación.

A continuación se especifica los procedimientos de configuración de las partes que componen el prototipo de cámara térmica y además la estructura de los

respectivos scripts de programación que permitirán el funcionamiento de todo el sistema.

2.3.1 CONFIGURACIÓN DE PLACA RASPBERRY PI 2B

Para iniciar con la configuración de la placa Raspberry Pi 2B, son necesarios ciertos periféricos básicos que permiten interactuar con dicha placa. Los periféricos necesarios son: un monitor o televisión, cable HDMI, fuente de alimentación o también puede utilizarse un cargador de celular que cumpla con las condiciones eléctricas del equipo, un teclado, un mouse y la tarjeta micro SD que contiene la información del sistema operativo.



Figura 2.12 Periféricos básicos para la configuración de la Raspberry Pi 2B [61].

2.3.1.1 Preparación de la tarjeta Micro SD

El sistema operativo que se utiliza sobre la Placa Raspberry Pi 2B es “*Raspbian*”, el cual ocupa un tamaño en memoria de aproximadamente 4 [GB]. Es necesario que la tarjeta micro SD sea capaz de contener esta información y además soportar la carga de otro tipo de aplicaciones, por lo tanto la capacidad recomendada debe ser mayor o igual a 8 [GB]; en el presente proyecto se utiliza una tarjeta de 16 [GB] de capacidad para prevenir cualquier sobrecarga de tamaño en disco.

La tarjeta micro SD debe estar completamente vacía, es decir, formateada. El programa recomendado por la fundación Raspberry Pi para formatear dicho dispositivo se denomina “*SD Formatter 4.0*” y se puede descargar ingresando al sitio web: https://www.sdcard.org/downloads/formatter_4/index.html. Una vez

instalado dicho software, se inserta la tarjeta micro SD en el computador, tomando en cuenta la letra de la unidad asignada (F) y se procede a formatear según como se muestra en la imagen a continuación:

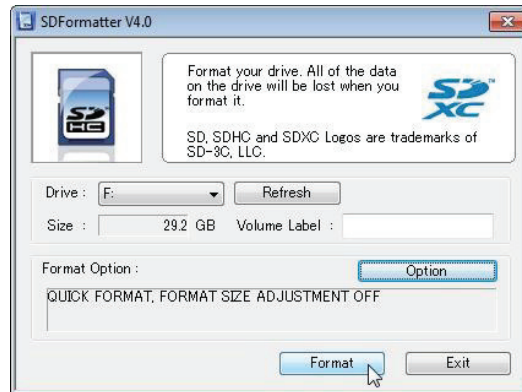


Figura 2.13 Interfaz del programa SD Formatter 4.0 [61].

2.3.1.2 Instalación del Sistema Operativo Raspbian

El proceso de instalación del sistema operativo Raspbian sobre la placa Raspberry Pi 2B se muestra en el siguiente diagrama de flujo:

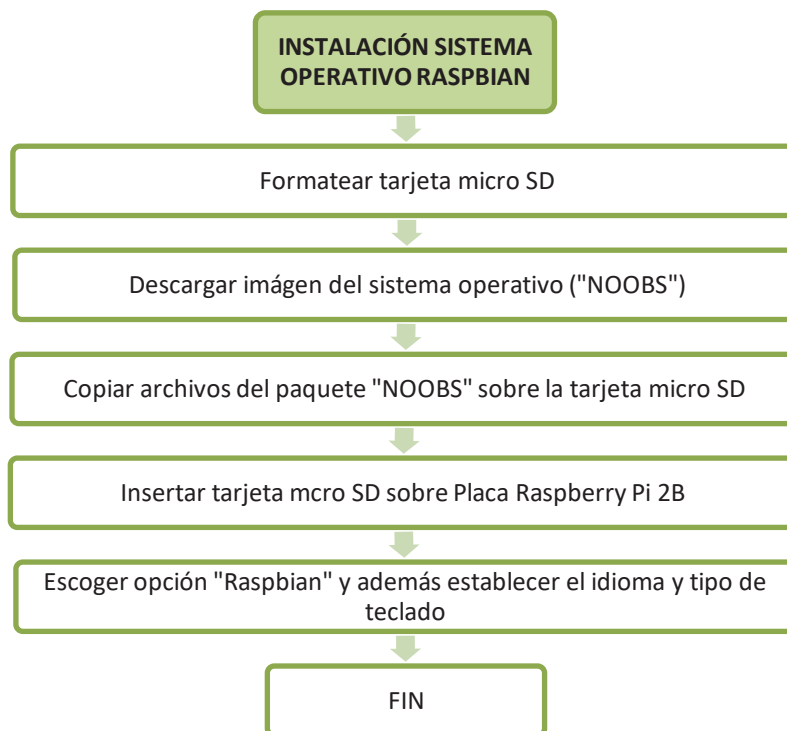


Figura 2.14 Diagrama de Flujo de Instalación del Sistema Operativo Raspbian.

Los detalles del proceso de instalación se especifican con mayor profundidad en el anexo A.

2.3.2 CONFIGURACIÓN DE PANTALLA TÁCTIL

Cuando ya se dispone del sistema operativo instalado sobre la Raspberry Pi 2B y totalmente funcional, se procede a configurar y ensamblar cada una de las partes que componen el prototipo.

El primer elemento en ser configurado es la pantalla, por lo tanto, se procede a conectar su dos vías de información, el cable HDMI y el cable USB, para mostrar la imagen y operar por medio de la función táctil respectivamente.

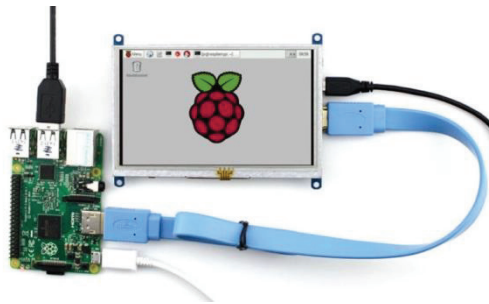


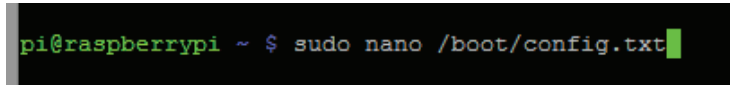
Figura 2.15 Conexión Pantalla - Raspberry Pi 2B.

Al momento de encender la pantalla se presenta un inconveniente con la imagen, se tiene del lado derecho de la pantalla una franja que recorta la imagen. Esto se debe a que no está configurada la resolución adecuada para el tipo de pantalla. Se puede observar el error en el gráfico a continuación:



Figura 2.16 Pantalla con errores en la imagen.

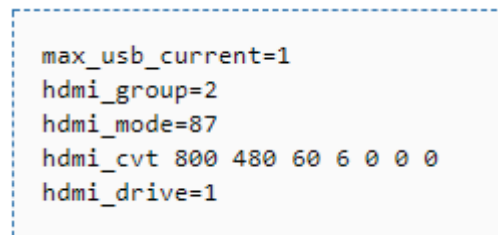
Para solucionar este inconveniente se debe establecer los rangos adecuados de resolución de la pantalla táctil, para esto se edita las características del archivo “*config.txt*” ubicado en el directorio raíz del sistema (*/boot/config.txt*).



```
pi@raspberrypi ~ $ sudo nano /boot/config.txt
```

Figura 2.17 Configuración del archivo *config.txt*

Se abre una ventana de terminal en el sistema operativo Raspbian y por medio del editor de texto “nano” se procede a editar el archivo “*config.txt*”. La siguiente línea de comandos establece los rangos de configuración óptima de la pantalla HDMI de 5 pulgadas LCD (B) que se utiliza en el proyecto, trabajando con una resolución de 800x480 pixeles:



```
max_usb_current=1
hdmi_group=2
hdmi_mode=87
hdmi_cvt 800 480 60 6 0 0 0
hdmi_drive=1
```

Figura 2.18 Rangos de configuración de la pantalla.

Después de establecer estos rangos se guarda el archivo “*config.txt*” y se procede a reiniciar el sistema, con lo cual se tiene la resolución adecuada de imagen para la pantalla utilizada en el prototipo.

La función que desempeña cada uno de los comandos se detalla a continuación:

2.3.2.1 **max_usb_current:**

Este comando sirve para aumentar la corriente en los puertos USB de la Raspberry Pi en los casos en que se utiliza dispositivos que requieren de mayores prestaciones de corriente. Por defecto los 4 puertos USB trabajan con 600 mA. Al agregar la línea de comandos en valor 1, se duplica la corriente de los puertos USB a 1200 mA [62].

2.3.2.2 `hdmi_group`:

Este comando define el grupo de salida HDMI especificando dos posibles opciones:

- CEA (*Consumer Electronics Association*, Asociación de Electrónicos de Consumo), es el estándar utilizado normalmente por los televisores.
- DMT (*Display Monitor Timings*, Tiempos de Visualización del Monitor), es el estándar utilizado típicamente por los monitores.

Este ajuste se debe utilizar junto con `hdmi_mode`. Las opciones de configuración se presentan en la tabla a continuación:

Tabla 2.1 Valores posibles del comando `hdmi_group` [63].

<code>hdmi_group</code>	Resultado
0	Detección automática desde EDID
1	CEA
2	DMT

2.3.2.3 `hdmi_mode`

Este comando se utiliza junto con `hdmi_group`, para definir el formato de salida HDMI, es decir se especifica la resolución de pantalla, junto con su frecuencia de trabajo y su relación de aspecto.

Existe un gran listado de diferentes opciones de resolución tanto para pantallas del grupo CEA como para el grupo DMT. Estos listados se pueden verificar en: <https://www.raspberrypi.org/documentation/configuration/config-txt/video.md>.

2.3.2.4 `hdmi_cvt`

Con este comando se puede personalizar un modo de pantalla que no se encuentre dentro de los listados de `hdmi_mode`. El modo `hdmi_cvt` permite establecer características especiales como:

- `<width>` - Ancho de la pantalla en pixeles.
- `<height>` - Altura de la pantalla en pixeles.
- `<framerate>` - Cuadros por segundo en Hz.
- `<aspect>` - Relación de aspecto con 6 tipos de elección posible.
- `<margins>` - Para habilitar o deshabilitar los márgenes de la pantalla.
- `<interlace>` - Para elegir un barrido progresivo o entrelazado de imagen.
- `<rb>` - Para elegir reducción de borrado o trabajar en modo normal.

Los tres últimos campos se pueden omitir para utilizar las opciones predeterminadas con valor cero. El hecho de optar por el comando `hdmi_cvt`, significa que se está creando el “`hdmi_mode=87`” del “`hdmi_group=2`”, ya que estos valores no aparecen en los listados. A continuación se muestran los valores que ofrece el comando `hdmi_cvt` [63].

Tabla 2.2 Valores posibles del comando `hdmi_cvt` [63].

Valor	Por Defecto	Descripción
ancho	(necesario)	Ancho en pixeles
alto	(necesario)	Alto en pixeles
cuadros por segundo	(necesario)	Cuadros por segundo en Hz
aspecto	3	Relación de aspecto: 1=4:3, 2=14:9, 3=16:9, 4=5:4, 5=16:10, 6=15:9
márgenes	0	0=márgenes deshabilitados, 1=márgenes habilitados
entrelazado	0	0=progresivo, 1=entrelazado
Rb	0	0=normal, 1=reducción de blanking

El sensor de cámara térmica Flir Lepton LWIR que se utiliza para el proyecto se conecta directamente con la interfaz GPIO de la Raspberry Pi 2B, a través de 8 cables conocidos como “jumpers” o cables de puente, que permiten la transmisión de las 8 señales de información entre estos dos dispositivos.



Figura 2.20 Conexión de cámara térmica y pantalla con la Raspberry Pi 2B.

En las figuras 2.19 y 2.20 se muestra la manera en cómo se encuentra conectada la cámara térmica.

Para proceder con la respectiva configuración de este dispositivo se necesita habilitar las interfaces SPI e I2C en el sistema operativo de la placa Raspberry Pi 2B. Estas interfaces permiten el intercambio de datos entre la Raspberry Pi y los periféricos que dependan de este tipo de datos. Este par de interfaces no están habilitadas de forma predeterminada y se necesita realizar el siguiente procedimiento para su configuración en el sistema operativo Raspbian.

2.3.3.1 Interfaz SPI

Para configurar la interfaz SPI es necesario abrir una ventana de terminal, y escribir el comando “*sudo raspi-config*”, con esto se habilita la herramienta *raspi_config*, que sirve para la configuración de ciertas características en la placa Raspberry Pi 2B, entre ellas la interfaz SPI.

Una vez iniciada esta aplicación se despliegan varios campos de configuración, se escoge el número 9 (*Advanced Options*) y se habilita la interfaz SPI [65].

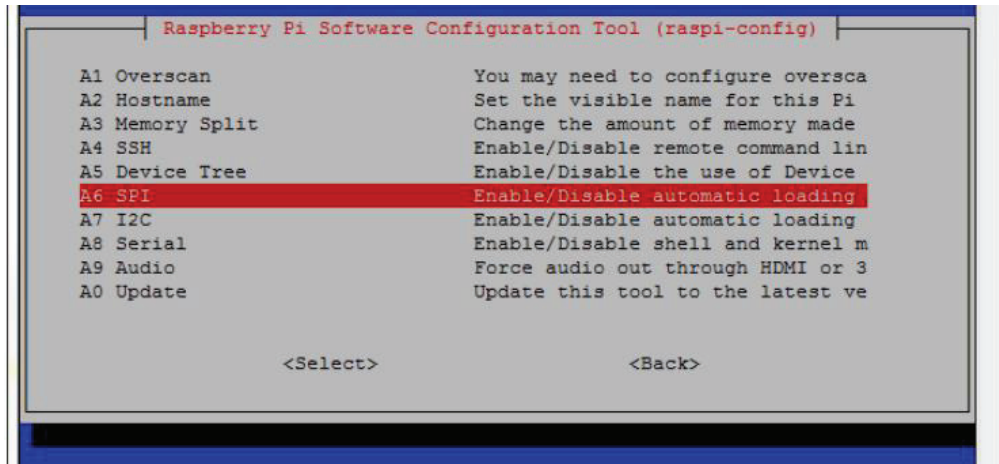


Figura 2.21 Activación de la Interfaz SPI [65].

2.3.3.2 Interfaz I2C

Para activar la interfaz I2C se realiza el mismo procedimiento detallado en la interfaz SPI, con la diferencia que en el campo 9 de “*Advanced Options*” se elige la opción I2C. Luego de haber activado las dos interfaces se sale de la herramienta de configuración y se reinicia el sistema para que los cambios surtan efecto.

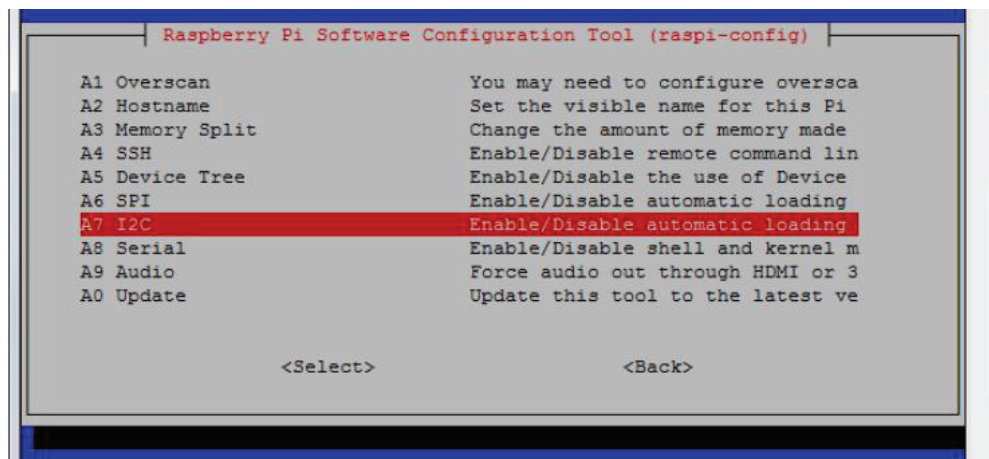


Figura 2.22 Activación de la Interfaz I2C [65].

2.3.3.3 Prueba de Funcionamiento

Para verificar que la cámara térmica se encuentra operando y desplegando información de imágenes térmicas sobre la placa Raspberry Pi 2B se puede utilizar la aplicación denominada “Qt” [65].

Por lo tanto se abre la ventana de terminal y se ejecuta el comando `“sudo apt-get install qt4-dev-tools”`, que permite instalar la herramienta `“Qt”` en el sistema operativo Raspbian [65].

Una vez que finaliza el proceso de instalación, se debe descomprimir la carpeta `“LeptonModule-master”`. Se realiza este proceso ingresando la sentencia `“unzip LeptonModule-master.zip”` en la ventana de terminal, específicamente en el directorio donde se encuentre el archivo descargado [65].

Luego en el directorio `“/software/raspberrypi_libs/leptonSDKEmb32PUB/”` se ejecuta el comando `“make”` para compilar el programa.

Una vez que el programa se encuentra compilado se retorna al directorio `“/software/raspberrypi_video/”`, y se ejecuta el comando `“qmake && make”` que permite crear el software portable del programa, gracias al comando `“qmake”`.

Y como último procedimiento para ejecutar la herramienta `“Qt”` se escribe en la línea de comandos la sentencia `“sudo ./raspberrypi_video”`, y el programa empezará a funcionar.

Luego de todo el proceso se obtuvo un error, el cual mostraba en la interfaz del programa solamente un recuadro rojo en la esquina superior izquierda y no se lograba observar la imagen térmica.

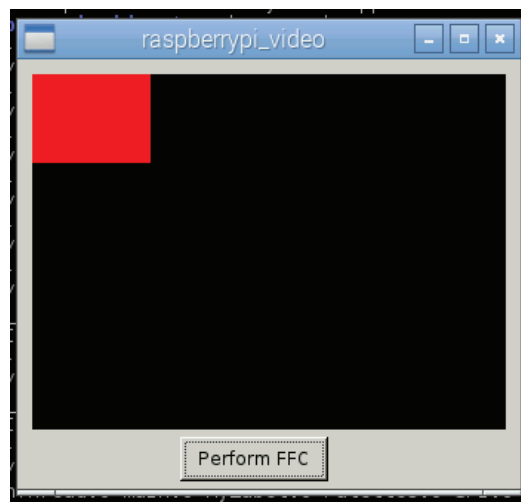


Figura 2.23 Error del programa de pruebas Qt [65].

Como solución se procede a desconectar cuidadosamente los cables jumpers que conectan la cámara térmica con la Raspberry Pi 2B, y se vuelve a conectarlos manteniendo el equipo encendido. Después de un par de intentos se consigue mostrar la imagen térmica sobre la pantalla, como se muestra en el siguiente gráfico:

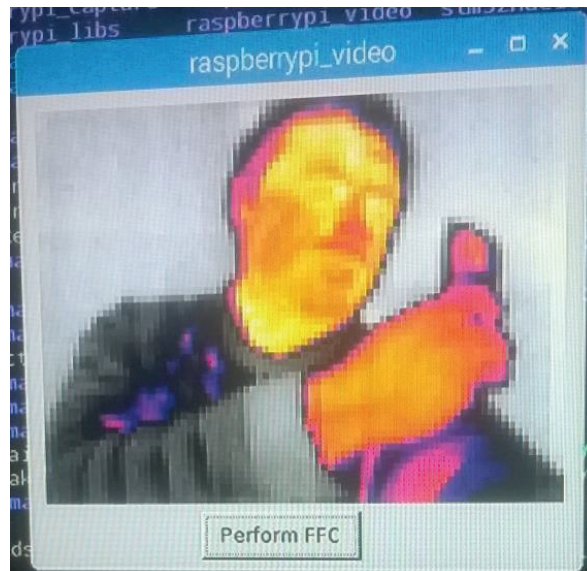


Figura 2.24 Imagen Térmica de prueba.

2.3.4 CONFIGURACIÓN DE MÓDEM USB

El equipo que permitirá la conexión del prototipo a la red móvil celular es el módem USB “*Huawei E303*”.



Figura 2.25 Módem USB “*Huawei E303*”.

El módem posee una ranura para insertar la tarjeta SIM (*Subscriber Identity Module*, Módulo de Identificación del Suscriptor) de la empresa que se elija como

proveedor del servicio de Internet. En Ecuador las empresas principales que ofrecen este servicio son: “Claro”, “CNT EP” y “Movistar”. De las tres empresas se elige “Claro”, considerando la compatibilidad con el módem USB utilizado en el proyecto.



Figura 2.26 Estructura interna del módem USB con ranura para Tarjeta SIM.



Figura 2.27 Tarjeta SIM de la empresa “Claro”.

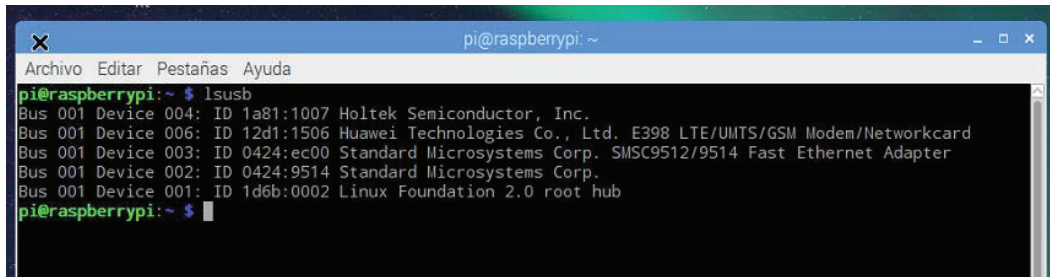
Para configurar el módem sobre la placa Raspberry Pi 2B es necesario realizar ciertos pasos detallados a continuación:

Se conecta el módem USB a la Raspberry Pi 2B y por medio del comando “*lsusb*” en la ventana de terminal se verifica que es reconocido como una unidad de almacenamiento y no como una tarjeta de red. Esto se debe a que los fabricantes diseñan estos dispositivos para que al conectarlos a equipos que trabajan con sistemas operativos como Windows o Mac, se instalen automáticamente los drivers desde el almacenamiento.

El comando “*lsusb*” muestra que el módem USB está conectado al “*Bus 001 Device 006*” con la “*ID 12d1:14fe*”. La numeración hexadecimal indica la identificación del proveedor (*idVendor=12d1*) y la identificación del producto (*idProduct=14fe*) [66].

Para solucionar este inconveniente se retira el módem de la Raspberry pi 2B y se instala el programa “*usb-modeswitch*” por medio de la sentencia “*sudo apt-get install usb-modeswitch*”. Luego se vuelve a conectar el dispositivo y se

comprueba que en esta ocasión es reconocido como una tarjeta de red utilizando el comando “*lsusb*”, ya que la identificación del producto cambia (*idProduct=1506*) [66].



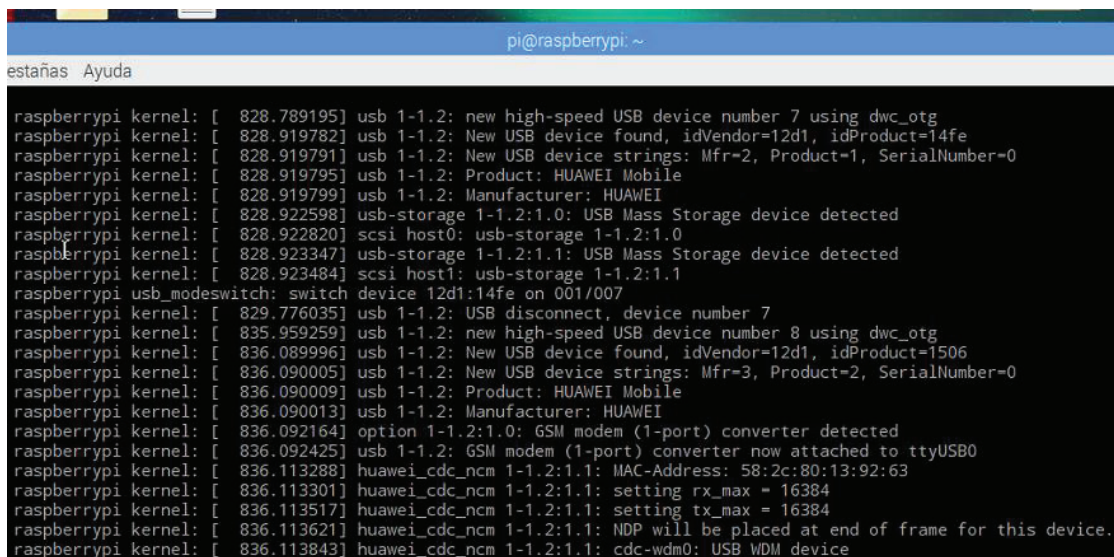
```

pi@raspberrypi: ~ $ lsusb
Bus 001 Device 004: ID 1a81:1007 Holtek Semiconductor, Inc.
Bus 001 Device 006: ID 12d1:1506 Huawei Technologies Co., Ltd. E398 LTE/UMTS/GSM Modem/Networkcard
Bus 001 Device 003: ID 0424:ec00 Standard Microsystems Corp. SMSC9512/9514 Fast Ethernet Adapter
Bus 001 Device 002: ID 0424:9514 Standard Microsystems Corp.
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
pi@raspberrypi: ~ $

```

Figura 2.28 Módem funcionando como Tarjeta de Red.

Para verificar el cambio de modo de operación del módem USB desde Unidad de Almacenamiento hacia Tarjeta de Red, se utiliza el comando “*tail -f /var/log/messages*”, el cual permite observar los logs del sistema, como se indica en la imagen a continuación:



```

raspberrypi kernel: [ 828.789195] usb 1-1.2: new high-speed USB device number 7 using dwc_otg
raspberrypi kernel: [ 828.919782] usb 1-1.2: New USB device found, idVendor=12d1, idProduct=14fe
raspberrypi kernel: [ 828.919791] usb 1-1.2: New USB device strings: Mfr=2, Product=1, SerialNumber=0
raspberrypi kernel: [ 828.919795] usb 1-1.2: Product: HUAWEI Mobile
raspberrypi kernel: [ 828.919799] usb 1-1.2: Manufacturer: HUAWEI
raspberrypi kernel: [ 828.922598] usb-storage 1-1.2:1.0: USB Mass Storage device detected
raspberrypi kernel: [ 828.922820] scsi host0: usb-storage 1-1.2:1.0
raspberrypi kernel: [ 828.923347] usb-storage 1-1.2:1.1: USB Mass Storage device detected
raspberrypi kernel: [ 828.923484] scsi host1: usb-storage 1-1.2:1.1
raspberrypi usb_modeswitch: switch device 12d1:14fe on 001/007
raspberrypi kernel: [ 829.776035] usb 1-1.2: USB disconnect, device number 7
raspberrypi kernel: [ 835.959259] usb 1-1.2: new high-speed USB device number 8 using dwc_otg
raspberrypi kernel: [ 836.089996] usb 1-1.2: New USB device found, idVendor=12d1, idProduct=1506
raspberrypi kernel: [ 836.090005] usb 1-1.2: New USB device strings: Mfr=3, Product=2, SerialNumber=0
raspberrypi kernel: [ 836.090009] usb 1-1.2: Product: HUAWEI Mobile
raspberrypi kernel: [ 836.090013] usb 1-1.2: Manufacturer: HUAWEI
raspberrypi kernel: [ 836.092164] option 1-1.2:1.0: GSM modem (1-port) converter detected
raspberrypi kernel: [ 836.092425] usb 1-1.2: GSM modem (1-port) converter now attached to ttyUSB0
raspberrypi kernel: [ 836.113288] huawei_cdc_ncm 1-1.2:1.1: MAC-Address: 58:2c:80:13:92:63
raspberrypi kernel: [ 836.113301] huawei_cdc_ncm 1-1.2:1.1: setting rx_max = 16384
raspberrypi kernel: [ 836.113517] huawei_cdc_ncm 1-1.2:1.1: setting tx_max = 16384
raspberrypi kernel: [ 836.113621] huawei_cdc_ncm 1-1.2:1.1: NDP will be placed at end of frame for this device.
raspberrypi kernel: [ 836.113843] huawei_cdc_ncm 1-1.2:1.1: cdc-wdm0: USB WDM device

```

Figura 2.29 Cambio de modo de operación del módem USB.

Luego de que ya se tiene habilitada esta nueva tarjeta de red, se procede a instalar el servicio *pppd*, que permite la utilización de *ppp* (*Point to Point Protocol*, Protocolo Punto a Punto) para la conexión a Internet por medio del módem USB; para esto se ejecuta el comando “*sudo apt-get install ppp*” [66].

Mediante el comando “*ifconfig*” se puede verificar la conexión a internet por medio del protocolo ppp, como se muestra en la imagen a continuación:

```
ppp0 Link encap:Point-to-Point Protocol
      inet addr:10.212.213.138 P-t-P:10.64.64.64 Mask:255.255.255.255
      UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 Metric:1
      RX packets:207 errors:1 dropped:0 overruns:0 frame:0
      TX packets:215 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:3
      RX bytes:20694 (20.2 KiB) TX bytes:37587 (36.7 KiB)
```

Figura 2.30 Conexión a Internet mediante el protocolo ppp.

Por último para configurar el módem USB mediante interfaz gráfica se necesita la instalación de los programas “*UMTSKeeper*” y “*Sakis3g*”, por lo tanto se crea un nuevo directorio sobre el sistema Raspbian denominado “*umtskeeper*” y se descarga dichos programas al ejecutar la sentencia “*sudo wget http://mintakaconciencia.net/squares/umtskeeper/src/umtskeeper.tar.gz*”, luego se descomprime el archivo descargado con el comando “*tar -xzvf umtskeeper.tar.gz*”, y finalmente se asigna permisos de ejecución a los dos programas con “*chmod +x sakis3g umtskeeper*” [66].

2.3.4.1 Pruebas de Conectividad

Para realizar las pruebas de conectividad del módem USB se utiliza el programa “*Sakis3g*” descargado anteriormente. Para lo cual desde la ventana de terminal se ejecuta el comando: “*sudo ./sakis3g --interactive*” y se tendrá la ventana que se muestra en la imagen a continuación:

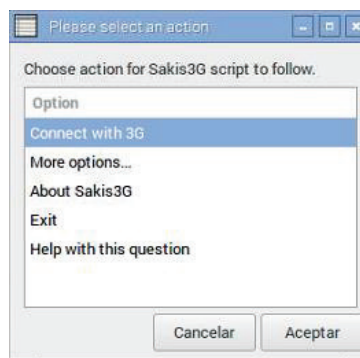


Figura 2.31 Configuración manual del módem USB.

Para configurar manualmente el módem USB se elige las siguientes opciones del programa “*sakis3g*”:

1. Se elige la opción “*Connect with 3G*”.
2. Posteriormente se selecciona el tipo de conexión “*USB Device*”.
3. Luego se elige el tipo de dispositivo “*HUAWEI Mobile*”.
4. Se escoge la interfaz “*Interface #0*”.
5. Se selecciona el APN (*Access Point Name*, Nombre de Punto de Acceso) del proveedor que se vaya a utilizar, en este caso para el presente proyecto se elige el APN “*ba.amx*” que corresponde a la empresa “Claro”.
6. Se describe el usuario y contraseña del APN, en este caso tanto el usuario y contraseña es la palabra “claro”.

Y finalmente si toda la información ingresada es correcta, el módem se conecta a la red celular correspondiente, y se muestra la siguiente ventana:

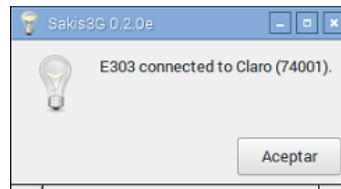


Figura 2.32 Módem USB conectado a la red celular de “Claro”.

Para obtener información detallada sobre la conexión del módem USB a la red celular, se elige la opción “*Connection information*” en el programa “*sakis3g*”, y se obtiene lo siguiente:

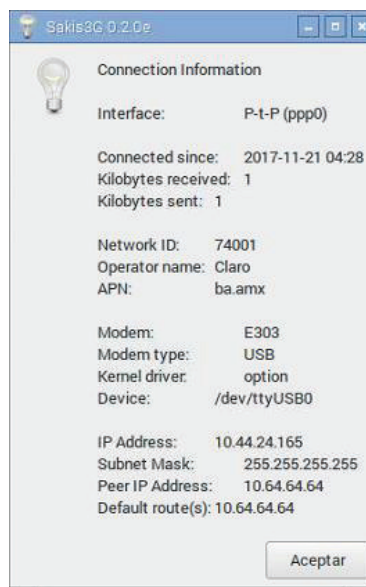


Figura 2.33 Información de la conexión a la red celular de “Claro”.

Físicamente se puede verificar la conexión del módem USB a Internet observando el color del led que tiene el equipo. Los detalles del significado de cada color se muestran en la tabla a continuación:

Tabla 2.4 Significado del color de led del módem USB [67].

COLOR DE LED	SIGNIFICADO
VERDE (parpadeando dos veces cada 3 segundos)	Módem USB está encendido.
VERDE (parpadeando una vez cada 3 segundos)	Módem USB se está registrando con una red 2G.
AZUL (parpadeando una vez cada 3 segundos)	Módem USB se está registrando con una red 3G.
VERDE (fijo)	Módem USB conectado a una red 2G.
AZUL (fijo)	Módem USB conectado a una red 3G.
CELESTE (fijo)	Módem USB conectado a una red 3G+.

2.3.4.2 Configuración de Conexión Automática

Para el presente proyecto de Cámara Térmica es necesario que el módem USB se pueda conectar a la red celular automáticamente, sin la necesidad de ninguna configuración manual previa.

Por tanto se utiliza el programa “UMTSKeeper” que permite configurar sobre el sistema Raspbian la opción para que el módem USB se ejecute automáticamente cada vez que se arranque el sistema operativo, o también cuando se produce una caída de la red celular.

El proceso de configuración se realiza editando el fichero “*rc.local*”, mediante la ejecución del comando “*sudo nano /etc/rc.local*”, y posteriormente se ingresa la siguiente sentencia: “*sudo /home/pi/umtskeeper/umtskeeper -sakisoperators*”

```
"USBINTERFACE='0'    OTHER='USBMODEM'    USBMODEM='12d1:1506'
APN='ba.amx'    APN_USER='0'    APN_PASS='0'" --sakisswitches "--sudo --
console" --devicename 'Huawei' --log --silent --monthstart 8 --nat 'no' --httpserver
&>> /home/pi/umtskeeper/error.log &";
```

tomando en cuenta que dicha sentencia debe ir colocada antes de la línea “exit 0” [68].

La información que contiene la sentencia anterior es la siguiente: interfaz serial, tipo de dispositivo, identificador de modo de operación, nombre de punto de acceso, usuario y contraseña.

Como información adicional para el caso en que se requiera desconectar el módem USB mediante línea de comandos se utiliza lo siguiente: “*sudo /home/pi/umtskeeper/umtskeeper --sakisoperators “disconnect”*” [68].

Para verificar los logs del sistema sobre el proceso de conexión automático del módem USB se consulta el fichero “*/var/log/umtskeeper.log*”, y sirve como una ayuda para poder arreglar cualquier anomalía que se presente durante el proceso. El tiempo promedio que se demora el módem USB en conectarse de manera automática a la red celular es de aproximadamente medio minuto [66].

2.3.5 CONFIGURACIÓN DE INTERFAZ GRÁFICA

Para el desarrollo de la Interfaz Gráfica de Usuario de la Cámara Térmica se utiliza el programa “*Qt Creator*”, el cual es un entorno de desarrollo integrado multiplataforma y de código abierto. El lenguaje de programación que se utiliza para el desarrollo de la aplicación es C++.

Primeramente se explicará la forma de instalación del programa “*Qt Creator*” sobre el sistema operativo “*Raspbian*”, y luego se expondrá el desarrollo de la Interfaz Gráfica de Usuario detallando la estructura de su programación con sus respectivos scripts principales; además la forma en cómo se controla el prototipo y cómo se realiza el envío de los archivos de imagen térmica hacia el servidor web.

2.3.5.1 Instalación de Qt Creator

Qt es un conjunto de librerías caracterizadas por ser independientes de la plataforma donde se ejecuten y que permiten generar GUIs (*Graphical User Interfaces*, Interfaces Gráficas de Usuario) de forma rápida y sencilla para muchos sistemas operativos, incluyendo sistemas embebidos como la Raspberry Pi [71].

Qt dispone de un IDE (*Integrated Development Environment*, Entorno de Desarrollo Integrado) denominado “Qt Creator” que permite crear interfaces gráficas por el método de arrastrar y soltar los componentes como botones, labels, textbox, etc. Además permite la depuración visual de los proyectos por medio de la simulación del programa paso a paso. Qt Creator trabaja principalmente con el lenguaje de programación C++, pero con el paso del tiempo el programa ha ido adaptando otro tipo de lenguajes como Python, Java, Perl, Ruby, PHP, etc [71].

La ventaja que posee Qt es ofrecer una versión gratuita y completamente funcional con licencia GPL (*General Public License*, Licencia Pública General). Esta versión es la que se instalará para el presente proyecto. Los pasos necesarios para la instalación del “IDE Qt Creator” se detallan en el siguiente diagrama de flujo:

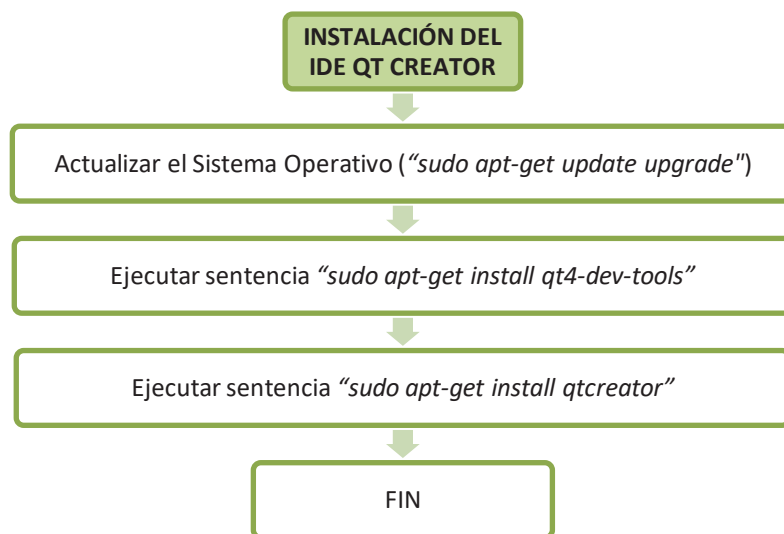


Figura 2.34 Diagrama de Flujo de Instalación del IDE Qt Creator.

El procedimiento para configurar el “*IDE Qt Creator*” se detalla en el siguiente diagrama de flujo:

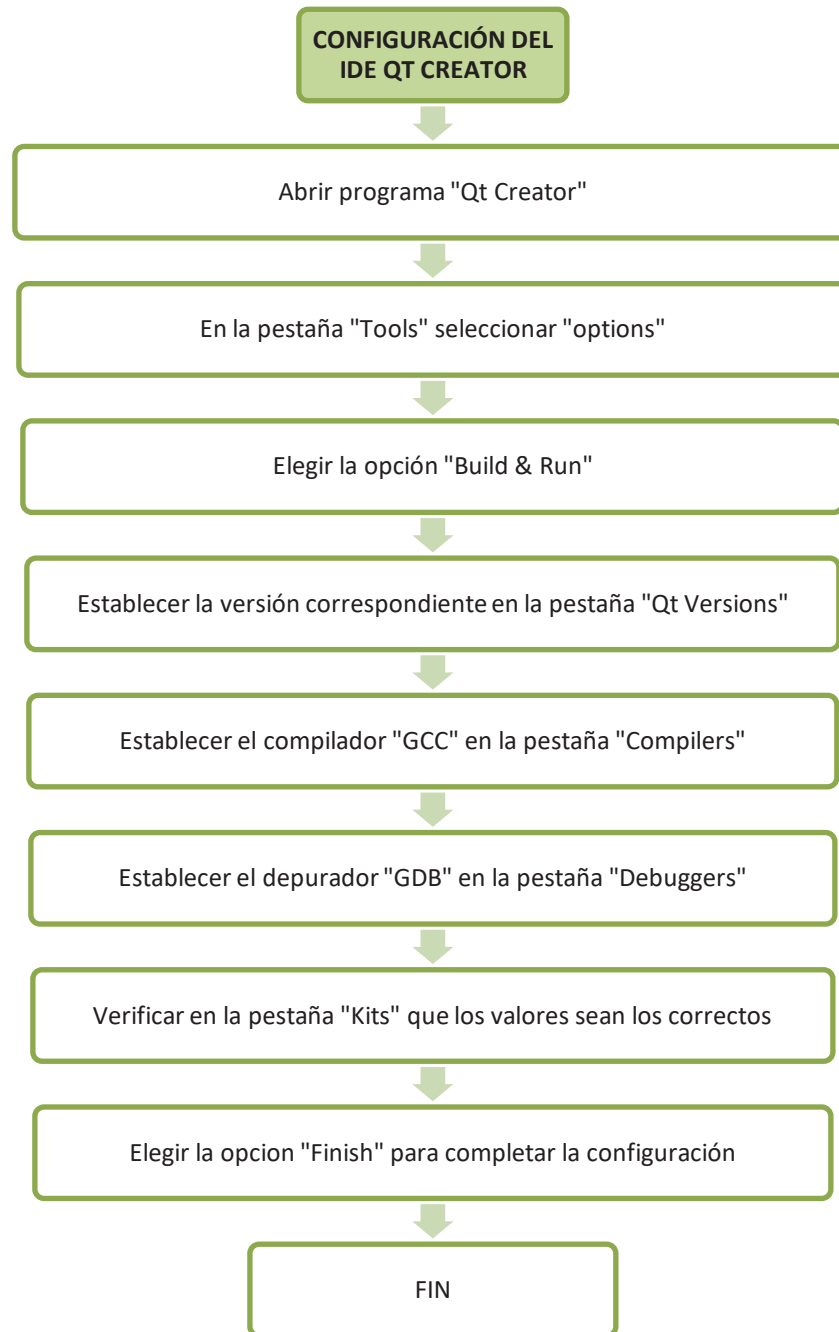


Figura 2.35 Diagrama de Flujo de Configuración del IDE Qt Creator.

Los detalles del proceso de instalación y configuración del entorno de desarrollo integrado “Qt Creator” se especificarán con mayor énfasis en el anexo B.

2.3.5.2 Desarrollo de la Interfaz Gráfica de Usuario

La aplicación que funcionará como Interfaz Gráfica de Usuario está compuesta por una ventana central en donde se despliega la imagen térmica que es detectada por el sensor LWIR Flir Lepton en tiempo real, así como una serie de botones que permiten realizar funciones específicas para el prototipo de cámara térmica.

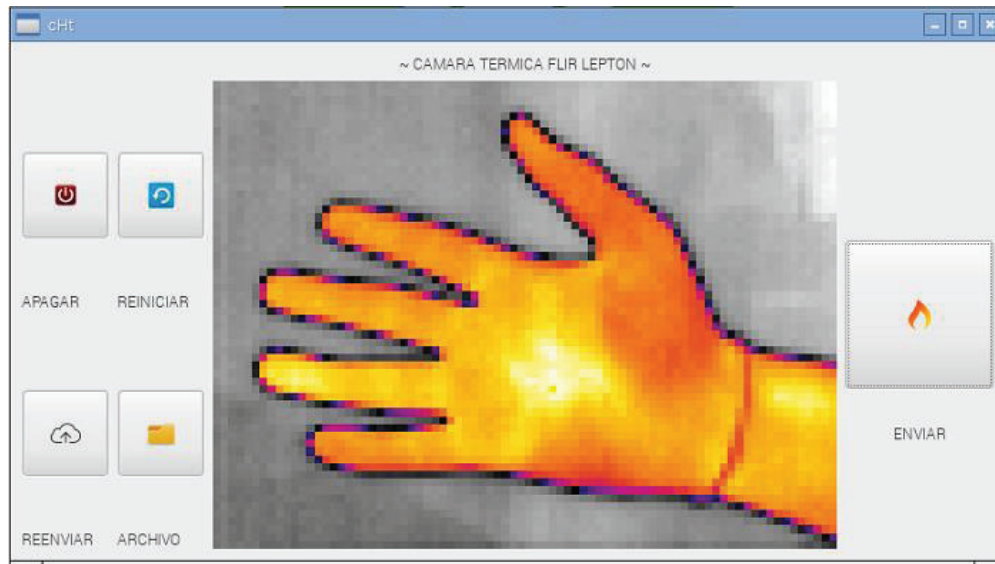


Figura 2.36 Interfaz Gráfica de Usuario de la Cámara Térmica.

Uno de los botones brinda la posibilidad de capturar una instantánea de la imagen térmica, para guardar localmente el archivo generado y también enviar este archivo automáticamente al servidor web.

Se tiene otro botón que permite verificar las imágenes guardadas localmente, es decir, se puede visualizar las imágenes térmicas que han sido capturadas y almacenadas en la memoria de la placa Raspberry Pi 2B.

Un problema que puede suscitarse al momento de capturar las imágenes térmicas es que los archivos no se envíen automáticamente al servidor web, debido a problemas de cobertura de la red celular. Como solución a este inconveniente se tiene un botón que sirve para enviar nuevamente las imágenes térmicas faltantes, y así actualizar el directorio de imágenes que se muestra en la aplicación web.

El momento oportuno para utilizar este botón será cuando el prototipo de cámara térmica se encuentre en una zona con mejor cobertura; o también para evitar el consumo de saldo del módem USB sería óptimo utilizar este botón al momento en que la placa Raspberry Pi 2B se encuentre conectada a Internet por medio de una red cableada.

Además se cuenta con un par de botones adicionales, que permitirán reiniciar el sistema operativo, así como apagar dicho sistema definitivamente, según la ocasión lo amerite.

Para la creación de la Interfaz Gráfica se procede a configurar un nuevo proyecto en el programa Qt Creator. Para esto en la barra de herramientas se elige el menú “*File*” y se selecciona la opción “*New File or Project*”, luego en la sección de proyectos se escoge “*Applications*”, se elige “*Qt Widgets Application*” y por último en la parte inferior la opción “*choose*”.

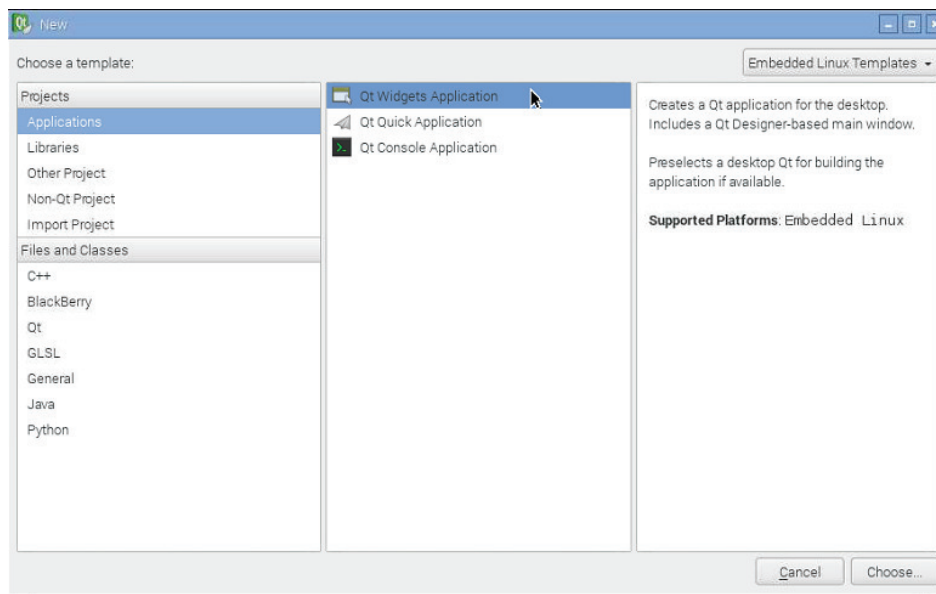


Figura 2.37 Creación de un nuevo Proyecto en Qt Creator.

Posterior a esto se debe elegir un nombre para el proyecto y la ubicación en donde será generado; el nombre del proyecto para la interfaz gráfica de la cámara térmica se define como “*CHt*” y por ende el fichero de proyecto (“*project*”) adquiere este mismo nombre con su respectiva extensión “*CHt.pro*”. Luego se

especifica la opción de “*Kit*”, lo cual determina los estilos que tendrá el entorno de la aplicación, que por defecto se tiene configurado como “*Desktop*” [74].

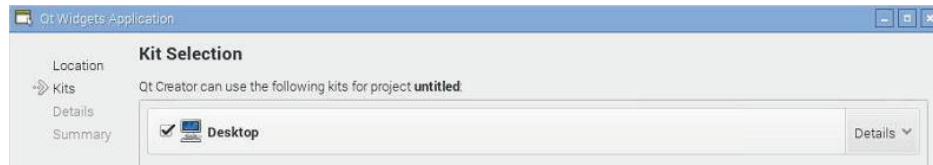


Figura 2.38 Selección del Kit en Qt Creator.

Después se especifica el nombre de la “*clase*” con la que se trabajará en el proyecto, que en este caso es “*MainWindow*”, se elige la “*clase base*” como “*QMainWindow*”, y por tanto se genera automáticamente los ficheros de cabecera (“*Headers*”) y ficheros de fuente (“*Sources*”) con el mismo nombre de la clase los cuales serán “*mainwindow.h*” y “*mainwindow.cpp*” respectivamente, además del fichero “*main.cpp*” que representa la función global para el arranque del proyecto [74].

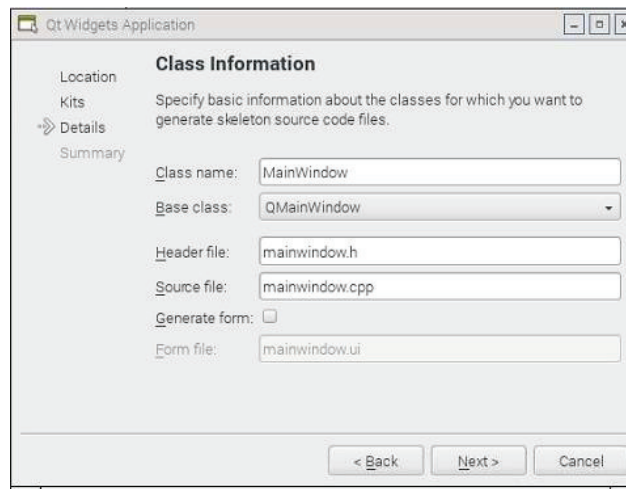


Figura 2.39 Configuración de la Información de Clase del proyecto.

En la ventana que aparece posteriormente se verifica toda la información acerca del proyecto y si todo está bien se procede a seleccionar el botón “*finish*” para crear el nuevo proyecto.

Además se adiciona a la estructura del proyecto los ficheros “*LeptonThread.h*” y “*LeptonThread.cpp*” que trabajan con la clase “*QThread*”. Estos ficheros sirven como hilos de control para extraer datos y señales del sensor de cámara térmica

LWIR Flir Lepton. Para obtener la información de la imagen térmica se utiliza las librerías del sensor Lepton, ubicadas en el directorio raíz “/camara_termica/raspberrypi_libs/leptonSDKEmb32PUB/”.

La estructura del proyecto realizado en “Qt Creator” se establece como se muestra en la imagen a continuación:

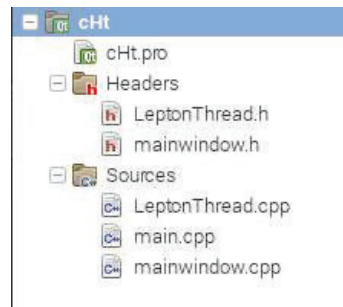


Figura 2.40 Estructura del Proyecto de Interfaz Gráfica.

2.3.5.2.1 Fichero de Cabecera “mainwindow.h”

Dentro del fichero de cabecera como punto inicial se declara las condicionales del preprocesador utilizando la directiva “*#ifndef*” [74].

```

1  #ifndef MAINWINDOW_H
2  #define MAINWINDOW_H
3
4
5
6
7  #endif // MAINWINDOW_H
8

```

Código 2.1 Condicionales del Preprocesador.

Dentro del condicional anteriormente mencionado se procede a incluir las librerías de Qt utilizadas en la implementación de la cabecera “mainwindow.h”, es decir *QMainWindow*, *QImage*, *QPushButton*, *QDesktopServices*, *QProcess*, *QStringList* y *QTimer*. Así como también se declara las clases que se utilizarán para la configuración del diseño de la aplicación de cámara térmica, siendo estas *QLabel*, *LeptonThread* y *QGridLayout*.

```

1  #ifndef MAINWINDOW_H
2  #define MAINWINDOW_H
3
4  #include <QMainWindow>
5  #include <QImage>
6  #include <QPushButton>
7  #include <QDesktopServices>
8  #include <QProcess>
9  #include <QStringList>
10 #include <QTimer>
11
12 class QLabel;
13 class LeptonThread;
14 class QGridLayout;
15

```

Código 2.2 Declaración de librerías y clases en “mainwindow.h”.

- **QMainWindow:** Proporciona la ventana de aplicación principal.
- **QImage:** Proporciona una representación de imagen de un dispositivo externo, en este caso el sensor LWIR Flir Lepton, permitiendo el acceso directo a los datos de píxeles.
- **QPushButton:** Permite la utilización de botones en la interfaz gráfica.
- **QDesktopServices:** Brinda la posibilidad de acceder a servicios de escritorio, en el presente proyecto se utiliza para acceder a los archivos locales de imagen.
- **QProcess:** Se usa para iniciar programas externos y comunicarse con ellos; en este caso se utiliza para la ejecución de scripts.
- **QStringList:** Proporciona una lista de cadenas, es decir en el presente proyecto se utiliza para formar un arreglo de la línea de comando que permite enviar los archivos de imagen térmica al repositorio en la nube.
- **QTimer:** Permite la aplicación de temporizadores para los diferentes procesos que se tiene en una aplicación. En este proyecto se utiliza esta clase para desplegar una etiqueta de texto durante cierto tiempo y luego hacer que desaparezca.

Además se cuenta con cierto tipo de clases dentro de este fichero, las cuales son:

- **QLabel:** Proporciona una visualización de texto o imagen, siendo utilizada para desplegar los títulos de la aplicación, así como la ventana de imagen térmica.

- **LeptonThread:** Esta es una clase propia del sensor LWIR Flir Lepton, que utiliza la clase QThread para poder administrar los hilos de los ficheros que proporcionan los datos de imagen y control del sensor de cámara térmica.
- **QGridLayout:** Permite dividir la ventana principal de la aplicación en filas y columnas, dando la posibilidad de ubicar a cada elemento de dicha aplicación en una celda específica.

En la sección posterior, dentro de la clase principal “*MainWindow*” se especifica la resolución de la imagen térmica que se mostrará en la interfaz gráfica, especificando los valores de ancho y de largo de la imagen, siendo estos “*ImageWidth = 496 pixeles*” e “*ImageHeight = 372 pixeles*” respectivamente. Los valores son determinados de acuerdo al tamaño de la pantalla táctil que se utiliza para el prototipo.

También en esta sección se declara los punteros, utilizados para referenciar y manipular las estructuras de datos de las diferentes funciones que tiene la aplicación de cámara térmica. Cada puntero está asociado a una clase específica de Qt.

```

16 class MainWindow : public QMainWindow {
17     Q_OBJECT
18
19     enum { ImageWidth = 496, ImageHeight = 372 };
20
21     QLabel *tituloLabel;
22     QLabel *imageLabel;
23     QLabel *notifiLabel;
24     LeptonThread *thread;
25     QGridLayout *layout;
26     QPushButton *boton2;
27     QPushButton *boton3;
28     QPushButton *boton4;
29     QPushButton *boton5;
30     QDesktopServices *abrir;
31     QProcess *webter;
32     QProcess *apaga;
33     QProcess *reinicio;
34     QProcess *nube;
35     QStringList *arreglo;
36     QTimer *lapso;
37     QLabel *etique1;
38     QLabel *etique2;
39     QLabel *etique3;
40     QLabel *etique4;
41     QLabel *etique5;
42
43     unsigned short rawMin, rawMax;
44     QVector<unsigned short> rawData;
45     QImage rgbImage;
46

```

Código 2.3 Declaración de Punteros.

Los punteros utilizados en el proyecto son: **tituloLabel*, **imageLabel*, **notifiLabel*, **thread*, **layout*, **boton2*, **boton3*, **boton4*, **boton5*, **abrir*, **webter*, **apaga*, **reinicio*, **nube*, **arreglo*, **lapso*, **etique1*, **etique2*, **etique3*, **etique4* y **etique5*. Dichos punteros cumplen con una función específica que será detallada más adelante.

Además se tiene los valores “*rawMin*”, “*rawMax*”, “*rawData*” y “*rgbImage*”, utilizados para el manejo de los datos que emite el sensor de cámara térmica Flir Lepton. Toda esta configuración se muestra en el fragmento de código 2.3:

Posteriormente se configura los especificadores de acceso para permitir controlar a una clase el acceso a las variables de datos de esa clase. Los especificadores de acceso permiten acceder a algunos miembros de la clase y restringir el acceso a otros.

En el presente proyecto se utiliza especificadores de acceso de tipo “*public*”, lo cual permite el acceso desde dentro y fuera de la clase, es decir se puede utilizar funciones que no sean propias de una clase específica.

En base a estos especificadores se declaran “*slots*”, que sirven para emitir señales conectadas a un proceso elaborado en dicho “*slot*”. Estas señales pueden ser invocadas por cualquier componente, independientemente de la clase a la cual pertenezca [75].

Los “*slots*” declarados son: *guardarCaptura*, *actualizarImagen*, *archivo*, *reenviarCapturas*, *apagaSistema* y *reinicioSistema*. Sus funciones serán detalladas más adelante.

```

42 public:
43     explicit MainWindow(QWidget *parent = 0);
44
45 public slots:
46     void guardarCaptura();
47     void actualizarImagen(unsigned short *, int, int);
48     void archivo();
49     void reenviarCapturas();
50     void apagaSistema();
51     void reinicioSistema();
52
53 };
54
55 #endif // MAINWINDOW_H
56

```

Código 2.4 Declaración de Slots.

2.3.5.2.2 Fichero de Fuente Principal “main.cpp”

Primeramente se incluye la librería de Qt que indica el tipo de interfaz que se propone realizar, en este caso será “*QApplication*”. También se incluye la clase que permite la creación de la ventana principal, en este caso será “*mainwindow.h*”.

Luego se crea una instancia de “*QApplication*” que permite recibir argumentos de la línea de comandos. Se procede a instanciar la clase “*MainWindow*” como “**mainwindow*” para que se genere la interfaz de usuario, se muestra dicha interfaz, y finalmente se ejecuta para ingresar al bucle de eventos que posibilitan interactuar con el interfaz mostrado.

```

1  #include <QApplication>
2
3  #include "mainwindow.h"
4
5  int main( int argc, char **argv )
6  {
7      QApplication a(argc, argv);
8
9      MainWindow *mainwindow = new MainWindow();
10     mainwindow->show();
11
12     return a.exec();
13 }
14
15

```

Código 2.5 Fichero de Fuente Principal.

2.3.5.2.3 Fichero de Fuente “mainwindow.cpp”

Dentro de este fichero se comienza por incluir el fichero de cabecera “*mainwindow.h*”, seguido por los distintos tipos de librerías Qt, que se necesitarán para la configuración de la interfaz gráfica de la cámara térmica. Además se incluye el fichero de cabecera “*LeptonThread.h*”, que conlleva la información de control y señales del sensor de cámara térmica LWIR Flir Lepton.

Las librerías de Qt que se incluyen dentro de este fichero son: *QLabel*, *QPushButton*, *QGridLayout*, *QImage*, *QPixmap*, *QPainter*, *QTime*, *QString*, *QDesktopServices*, *QUrl* y *QProcess*.

```

1  #include "mainwindow.h"
2
3  #include <QLabel>
4  #include <QPushButton>
5  #include <QGridLayout>
6  #include <QImage>
7  #include <QPixmap>
8  #include <QPainter>
9  #include <QTime>
10 #include <QString>
11 #include <QDesktopServices>
12 #include <QUrl>
13 #include <QProcess>
14
15 #include "LeptonThread.h"
16

```

Código 2.6 Declaración de librerías en “mainwindow.cpp”.

Ciertas librerías ya fueron detalladas anteriormente, por lo tanto las características de las librerías restantes se especifican a continuación:

- **QPixmap:** Proporciona funciones para mostrar imágenes en pantalla, este tipo de librería se puede mostrar fácilmente en pantalla utilizando las etiquetas QLabel. En el proyecto se utiliza para mostrar la imagen térmica en base a los datos proporcionados por el sensor.
- **QPainter:** Permite reflejar un dibujo de bajo nivel en widgets y otros dispositivos de pintura. Proporciona funciones para recrear la mayor parte de dibujos que requieren los programas de Interfaz Gráfica de Usuario. En el proyecto se utiliza para almacenar y mostrar los colores del mapa de pixeles que detecta el sensor de cámara térmica.
- **QTime:** Proporciona funciones de tiempo de reloj. En el proyecto se utiliza para extraer la fecha y hora de un determinado momento e incluir dicho dato en el nombre de archivo de imagen térmica capturada.
- **QString:** Proporciona una cadena de caracteres Unicode, utilizada para obtener y mostrar los datos de fecha y hora entregados por QTime.
- **QUrl:** Proporciona una interfaz para trabajar con URL (*Uniform Resource Locator*, Localizador Uniforme de Recursos). En el proyecto se utiliza para ubicar el directorio local en donde se guardan los archivos de imágenes térmicas capturadas.

Como siguiente paso se procede a configurar el modelo de color que tendrá la imagen térmica, se elige el formato de color “RGB888”, que trabaja con 24 bits

en 3 canales de color, dividiendo 8 bits por cada canal. Los colores que se utiliza son rojo (R, *red*) verde (G, *green*) y azul (B, *blue*), en sus diferentes tonalidades.

```

17 // Establecimiento del modelo de color de la imagen receptada
18 MainWindow::MainWindow(QWidget *parent)
19     : QMainWindow(parent)
20     , rawData(LeptonThread::FrameWords)
21     , rgbImage(LeptonThread::FrameWidth, LeptonThread::FrameHeight, QImage::Format_RGB888) {
22

```

Código 2.7 Modelo de color RGB888.

Los valores “*FrameWidth*” y “*FrameHeight*” indican la resolución del sensor de cámara térmica de 80x60 pixeles respectivamente. La imagen está compuesta por el valor “*FrameWords = FrameWidth*FrameHeight*”, y se genera la variable “*rawData*” (fragmento de código 2.7).

La clase “*QWidget*” es la clase base de todos los objetos de la interfaz de usuario. Se declara la variable “*layout*” que servirá para indicar la posición de cada objeto sobre la interfaz. “*QGridLayout*” permite dividir la ventana principal en filas y columnas para formar celdas, en las cuales se dispone cada objeto.

```

23 // Establecimiento del diseño de la aplicacion
24 QWidget *mainWidget = new QWidget();
25 setCentralWidget(mainWidget);
26
27 layout = new QGridLayout();
28 mainWidget->setLayout(layout);
29

```

Código 2.8 Modelo de color RGB888.

Para establecer el título de la aplicación “~CAMARA TERMICA FLIR LEPTON~”, se declara la variable “*tituloLabel*”, funcionando como una etiqueta de texto, y su posición se determina en base a la variable “*layout*”, en este caso se ubica el título en la fila 0, columna 2 de la ventana principal.

```

30 // Establecimiento del titulo de la aplicacion
31 tituloLabel= new QLabel("~ CAMARA TERMICA FLIR LEPTON ~");
32 layout->addWidget(tituloLabel, 0, 2, Qt::AlignCenter);
33

```

Código 2.9 Título de la aplicación.

Para observar la imagen térmica en la ventana principal se determina la variable “*imageLabel*”, funcionando como una etiqueta de imagen. Su posición inicia en la fila 1, columna 2; y además se especifica que este objeto ocupe 2 espacios de

celda en fila y 1 espacio de celda en columna, debido a cuestiones de diseño de la aplicación.

```

34 // Posicion de la imagen termica en la aplicacion
35 imageLabel = new QLabel();
36 layout->addWidget(imageLabel, 1, 2, 2, 1, Qt::AlignCenter);
37

```

Código 2.10 Posición de la Imagen Térmica.

El hecho de que en ciertos casos no se pueda mostrar la imagen térmica, se debe a inconvenientes de comunicación entre el sensor de cámara térmica y la interfaz GPIO. Para esto se configura desplegar un color de relleno (rojo) en vez de la imagen térmica, indicando el error.

```

38 // Error de comunicacion del sensor con la interfaz GPIO
39 QPixmap filler(ImageWidth, ImageHeight);
40 filler.fill(Qt::red);
41 imageLabel->setPixmap(filler);
42

```

Código 2.11 Color de relleno (rojo) para indicar error de comunicación.

El proceso para mostrar la imagen térmica inicia por la configuración del slot que contiene la señal “*actualizarImagen*”. En primer lugar se registra los valores mínimo y máximo de la variable “*rawData*”. Luego se transforma el archivo “*rawData*” a valores RGB, generando la variable “*rgbImage*” por medio de los datos de color “*colormap*”.

Se toma un valor de entrada del intervalo {0, 65536} y se asigna el valor al intervalo {0, 256} para establecer el pixel en su valor de color anterior. La función “*setPixel*” permite establecer el color de píxel en una posición especificada.

Y finalmente para actualizar la imagen se utiliza las clases “*QPixmap*” y “*QPainter*”, accediendo a la variable “*rgbImage*” y desplegando esta imagen con la variable “*imageLabel*”. La dimensión de la imagen en la aplicación se define con los valores “*ImageWidth = 496*” e “*ImageHeight = 372*” declarados en el fichero de cabecera “*mainwindow.h*”.


```

const int colormap[] = {255, 255, 255, 253, 253, 253, 251, 251, 251, 249, 249, 249, 247, 247, 247, 245, 245, 245, 243, 243,
// Funcion de actualizacion de imagen termica
void MainWindow::actualizarImagen(unsigned short *data, int minValue, int maxValue){
    // Registro de raw data y sus valores min/max
    memcpy(&rawData[0], data, 2*LeptonThread::FrameWords);
    rawMin = minValue; rawMax = maxValue;

    // Transformacion de "rawData" a valores rgb en "rgbImage", a traves de "colormap"
    int diff = maxValue - minValue + 1;
    for (int y = 0; y < LeptonThread::FrameHeight; ++y) {
        for (int x = 0; x < LeptonThread::FrameWidth; ++x) {
            int baseValue = rawData[LeptonThread::FrameWidth*y + x]; // se toma el valor de entrada en [0, 65536)
            int scaledValue = 256*(baseValue - minValue)/diff; // se asigna el valor al intervalo [0, 256), y se establece
            rgbImage.setPixel(x, y, qRgb(colormap[3*scaledValue], colormap[3*scaledValue+1], colormap[3*scaledValue+2]));
        }
    }

    // Actualizacion de la imagen en pantalla
    QPixmap pixmap = QPixmap::fromImage(rgbImage).scaled(ImageWidth, ImageHeight, Qt::KeepAspectRatio);
    QPainter painter(&pixmap);
    imageLabel->setPixmap(pixmap);
}

```

Código 2.12 Actualización de Imagen Térmica.

Una vez finalizado el proceso del “slot”, se tiene la señal “*actualizarImagen*”, la cual se conecta con el hilo de datos “*thread*”, que contiene la información de imagen emitida desde el sensor de cámara térmica Flir Lepton, en base a la librería “*LeptonThread*”.

```

// Recepcion de los datos de imagen termica
thread = new LeptonThread();
connect(thread, SIGNAL(actualizarImagen(unsigned short *,int,int)), this, SLOT(actualizarImagen(unsigned short *, int,int)));
thread->start();

```

Código 2.13 Recepción de datos desde el sensor Flir Lepton.

La siguiente señal en ser configurada sobre un slot es “*guardarCaptura*”, que sirve para capturar la imagen térmica, guardarla localmente, y enviarla hacia un servidor en la nube.

El proceso inicia con la declaración de la variable “*fechat tiempo*” que permite obtener datos de fecha y hora en que se realiza la captura de imagen y posteriormente adicionar este dato al nombre de archivo de imagen térmica.

Utilizando la variable “*rgbImage*” se guarda un archivo de captura de imagen sobre un directorio específico de la Raspberry Pi 2B, con el nombre “*termica_yyyyMMdd_HHmmss*”. Los campos *yyyy=año*, *MM=mes*, *dd=día*, *HH=hora*, *mm=minuto*, *ss=segundo*, indican el tiempo exacto en que se realiza una captura. El archivo de imagen se guarda en formato “*.jpg*” con 100% de calidad.

Para enviar el archivo de imagen hacia el servidor en la nube se declara la variable “*arreglo*”, permitiendo ejecutar la línea de comandos “*sudo scp -i /home/pi/servidorawstermica2017.pem /home/pi/Desktop/camara_termica/build-cHt-Desktop-Debug/repositorio/térmica%1.jpg ubuntu@18.221.244.121:/var/www/html/repositorio/*”.

La línea de comandos está compuesta por: *protocolo de transferencia de archivos “scp” + llave de acceso al servidor web + directorio local + directorio remoto*.

Además se configura el mensaje de aviso “*Enviando Captura...*”, que indica la captura de una imagen. Para esto se declara la variable “*notifiLabel*”, se indica la posición en que se desea desplegar el aviso sobre la aplicación, y con la variable “*lapso*” se especifica un periodo de tiempo para que el aviso desaparezca luego de que la imagen haya sido capturada.

```
// Funcion para capturar, guardar y enviar imagen termica
void MainWindow::guardarCaptura() {
    // Se crea variable de fecha y hora, para asignar al nombre de archivo
    QString fechatiempo = QDateTime::currentDateTime().toString("_'yyyMMdd_HHmss");
    // Se guarda imagen localmente en formato jpg con calidad 100%
    QImage.save(QString("/home/pi/Desktop/camara_termica/build-cHt-Desktop-Debug/repositorio/termica%1.jpg").arg(fechatiempo), "JPG", 100);
    // Se envia imagen capturada al servidor en la nube
    arreglo = new QStringList();
    arreglo->append("scp");
    arreglo->append("-i");
    arreglo->append("/home/pi/servidorawstermica2017.pem");
    arreglo->append(QString("/home/pi/Desktop/camara_termica/build-cHt-Desktop-Debug/repositorio/termica%1.jpg").arg(fechatiempo));
    arreglo->append("ubuntu@18.221.244.121:/var/www/html/repositorio/");
    nube->start("sudo",*arreglo);
    // Se muestra aviso, informando que la imagen fue capturada
    notifiLabel= new QLabel("Enviando\nCaptura...");
    layout->addWidget(notifiLabel, 1, 3, Qt::AlignCenter);
    lapso = new QTimer();
    lapso->singleShot(7000, notifiLabel, SLOT(hide()));
}
}
```

Código 2.14 Captura y envío de archivo de imagen térmica.

La señal “*guardarCaptura*” se conecta al botón “*snapshotButton*”, y las funciones que cumple dicho botón al ser presionado son: capturar imagen térmica, guardar archivo capturado localmente y enviar archivo al servidor web.

```
47 // Boton para capturar imagenes
48 QPushButton *snapshotButton = new QPushButton();
49 layout->addWidget(snapshotButton, 1, 3, 2, 1, Qt::AlignCenter);
50 connect(snapshotButton, SIGNAL(clicked()), this, SLOT(guardarCaptura()));
51 snapshotButton->setIcon(QIcon("/home/pi/Desktop/camara_termica/build-cHt-Desktop-Debug/iconos/llama.png"));
52 snapshotButton->setMinimumSize(127,127);
53
```

Código 2.15 Botón conectado a la señal “*guardarCaptura*”.

Posteriormente en otro “slot” se configura la señal “archivo”, declarando la variable “abrir”, para abrir el directorio local de la Raspberry Pi 2B en donde se guardan los archivos de imágenes capturadas. Esta señal se conecta a la variable “boton4”.

```
// Funcion para abrir archivo local
void MainWindow::archivo() {

    // Se abre el directorio local que contiene las imagenes termicas
    abrir->openUrl(QUrl("/home/pi/Desktop/camara_termica/build-cHt-Desktop-Debug/repositorio/", QUrl::TolerantMode));

}
```

Código 2.16 Señal “archivo”.

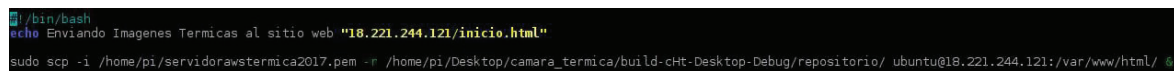
En el siguiente “slot” se configura la señal “reenviarCapturas”, se declara la variable “webter” que posibilita la ejecución del script “enviater”, creado previamente y alojado en el directorio “/usr/local/bin/enviater” de la Raspberry Pi 2B. La señal “reenviarCapturas” se conecta a la variable “boton2”.

```
158 // Funcion para reenviar imagenes capturadas
159 void MainWindow::reenviarCapturas() {
160
161     // Opcion emergente para reenviar imagenes capturadas
162     webter->start("/usr/local/bin/enviater");
163
164 }
165
```

Código 2.17 Señal “reenviarCapturas”.

El script “enviater” establece la orden para enviar nuevamente las imágenes térmicas faltantes, que tuvieron problemas al ser enviadas inicialmente al servidor web, debido a fallos de cobertura de la red celular.

Se ejecuta el comando “`sudo scp -i /home/pi/servidorawstermica2017.pem -r /home/pi/Desktop/camara_termica/build-cHt-Desktop-Debug/repositorio/ ubuntu@18.221.244.121:/var/www/html/ &`”.



```
~/bin/bash
echo Enviando Imagenes Termicas al sitio web "18.221.244.121/inicio.html"
sudo scp -i /home/pi/servidorawstermica2017.pem -r /home/pi/Desktop/camara_termica/build-cHt-Desktop-Debug/repositorio/ ubuntu@18.221.244.121:/var/www/html/ &
```

Código 2.18 Script “enviater”.

Los posteriores “slots” contienen las señales “apagaSistema” y “reinicioSistema”, se configuran para apagar y reiniciar el sistema operativo de la Raspberry Pi 2B

respectivamente. Se declara la variable “*apaga*” que permite ejecutar el script “*apagater*” y la variable “*reinicio*” que permite ejecutar el script “*reiniter*”.

Las señales “*apagaSistema*” y “*reinicioSistema*” se conectan a las variables “*boton3*” y “*boton5*” respectivamente.

```

166 // Funcion para apagar el sistema
167 void MainWindow::apagaSistema() {
168
169     // Se ejecuta script para apagar el sistema
170     apaga->start("/usr/local/bin/apagater");
171
172 }
173
174 // Funcion para reiniciar el sistema
175 void MainWindow::reinicioSistema() {
176
177     // Se ejecuta script para reiniciar el sistema
178     reinicio->start("/usr/local/bin/reiniter");
179
180 }
181

```

Código 2.19 Señales “*apagaSistema*” y “*reinicioSistema*”.

Los scripts “*apagater*” y “*reiniter*” contienen líneas de comando con sentencias específicas para apagar y reiniciar el sistema operativo de la Raspberry Pi 2B respectivamente.

```

#!/bin/bash
echo Apagando el Sistema.

sudo shutdown -h now &

```

Código 2.20 Script “*apagater*”.

```

#!/bin/bash
echo Reiniciando el Sistema.

sudo reboot &

```

Código 2.21 Script “*reiniter*”.

A continuación se especifica la configuración de los botones que se conectan a las señales establecidas anteriormente. Los botones “*boton2*”, “*boton3*”, “*boton4*” y “*boton5*” se conectan a las señales “*reenviarCapturas*”, “*apagaSistema*”, “*archivo*” y “*reinicioSistema*” respectivamente.

La configuración para cada botón es similar, se especifica la posición que tendrá sobre la aplicación, se establece un icono con una imagen específica y se define el tamaño en pixeles que tendrá cada botón.

```
// Boton de reenvio de capturas
boton2 = new QPushButton();
layout->addWidget(boton2, 2, 0, Qt::AlignCenter);
connect(boton2, SIGNAL(clicked()), this, SLOT(reenviarCapturas()));
boton2->setIcon(QIcon("/home/pi/Desktop/camara_termica/build-cHT-Desktop-Debug/iconos/cloud.png"));
boton2->setMinimumSize(70,70);

// Boton para apagar el sistema
boton3 = new QPushButton();
layout->addWidget(boton3, 1, 0, Qt::AlignCenter);
connect(boton3, SIGNAL(clicked()), this, SLOT(apagaSistema()));
boton3->setIcon(QIcon("/home/pi/Desktop/camara_termica/build-cHT-Desktop-Debug/iconos/off.png"));
boton3->setMinimumSize(70,70);

// Boton para abrir directorio de imagenes locales
boton4 = new QPushButton();
layout->addWidget(boton4, 2, 1, Qt::AlignCenter);
connect(boton4, SIGNAL(clicked()), this, SLOT(archivo()));
boton4->setIcon(QIcon("/home/pi/Desktop/camara_termica/build-cHT-Desktop-Debug/iconos/carpeta.png"));
boton4->setMinimumSize(70,70);

// Boton para reiniciar el sistema
boton5 = new QPushButton();
layout->addWidget(boton5, 1, 1, Qt::AlignCenter);
connect(boton5, SIGNAL(clicked()), this, SLOT(reinicioSistema()));
boton5->setIcon(QIcon("/home/pi/Desktop/camara_termica/build-cHT-Desktop-Debug/iconos/reboot.jpg"));
boton5->setMinimumSize(70,70);
```

Código 2.22 Configuración de botones.

Finalmente se compila y ejecuta el código generado sobre el programa “Qt Creator”, y si el archivo no presenta errores de compilación, se desplegará la Interfaz Gráfica de Usuario creada para el Prototipo de Cámara Térmica.

2.3.6 CONFIGURACIÓN DE APLICACIÓN WEB

Las imágenes térmicas capturadas con el prototipo de cámara térmica, se envían a un servidor en la nube para poder desplegar los archivos de imagen sobre una aplicación web.

La implementación de la aplicación web comprende inicialmente el establecimiento de un servidor web, que permitirá el acceso de los usuarios mediante Internet al sitio web que se genere. Para el desarrollo del sitio web o aplicación web, se utiliza lenguajes de programación como HTML, CSS, PHP, JavaScript, etc.

2.3.6.1 Servidor Web

El montaje del servidor web se realiza sobre el servicio de computación en la nube “*Amazon Web Services*”.

Por tal razón es necesario crear una cuenta de usuario sobre este servicio, la cual será gratuita durante los primeros 12 meses. Para la creación de la cuenta se accede a la dirección web “*https://aws.amazon.com/es/free/*” y se elige la opción “Cree una cuenta gratuita”.



uro | <https://portal.aws.amazon.com/billing/signup?n>

aws

Crear una cuenta de AWS

Nombre de la cuenta de AWS

Dirección de correo electrónico

Contraseña

Confirmar contraseña

[Continuar](#)

[Iniciar sesión en una cuenta de AWS existente](#)

Figura 2.41 Creación de cuenta de Amazon Web Services (AWS).

Se llena toda la información solicitada por la empresa, tomando en cuenta que es necesario contar con una tarjeta de crédito, que sirve como garantía para la utilización del servicio.

Una vez finalizado todo el proceso de registro se accede al entorno de trabajo de Amazon Web Services, en donde se puede elegir diferentes tipos de servicios. Para el presente proyecto se opta por el servicio “*Amazon EC2*” (*Amazon Elastic Compute Cloud*), el cual permite acceder a ordenadores virtuales con cierto tipo de características, dependiendo de la instancia que se elija.

“Amazon EC2” proporciona una amplia selección de tipos de instancias para adaptarse a diferentes casos de uso. Los tipos de instancia abarcan varias combinaciones de capacidad de CPU, memoria, almacenamiento y redes [76].

Pero el hecho de solamente contratar una capa gratuita limita la elección del tipo de instancias; por lo tanto se elige la instancia de tipo “t2.micro” que se encuentra dentro del rango de la capa gratuita de Amazon y la cual proporciona un nivel base de desempeño de la CPU.

El sistema operativo que se elige para el servidor web y que se encuentra dentro de las opciones de la instancia “t2.micro” es “Ubuntu Server 16.04 LTS” (Long Term Support, Soporte a Largo Plazo).



Figura 2.42 Sistema Operativo del servidor web.

Antes de finalizar el proceso de creación de la nueva instancia, se debe generar un par de claves de seguridad, que consisten en una clave pública almacenada por AWS y un archivo de llave de seguridad almacenado por el usuario, para conectarse a la instancia de forma segura. El archivo de llave de seguridad generado se establece como “servidorawstermica2017.pem”.

Instance: **i-0ad10aa3ec5654219** (servidor camara termica) Elastic IP: 18.221.244.121

Description	Status Checks	Monitoring	Tags
Instance ID	i-0ad10aa3ec5654219		
Instance state	running		
Instance type	t2.micro		
Elastic IPs	18.221.244.121*		
Availability zone	us-east-2c		
Security groups	launch-wizard-1. view inbound rules		
Scheduled events	No scheduled events		
AMI ID	ubuntu/images/hvm-ssd/ubuntu-xenial-16.04-amd64-server-20170721 (ami-10547475)		
Public DNS (IPv4)	ec2-18-221-244-121.us-east-2.compute.amazonaws.com		
IPv4 Public IP	18.221.244.121		
IPv6 IPs	-		
Private DNS	ip-172-31-44-108.us-east-2.compute.internal		
Private IPs	172.31.44.108		
Secondary private IPs			
VPC ID	vpc-50102839		
Subnet ID	subnet-6a6aff27		

Figura 2.43 Instancia t2.micro para el servidor web.

Una vez creada la instancia, se genera automáticamente la IP Pública “18.221.244.121” que servirá para ingresar a la aplicación web. También se cuenta con una IP Elástica “18.221.244.121”, DNS Público, DNS Privado, IP Privada “172.31.44.108”, y otro tipo de características que se muestran en la Figura 2.43.

El siguiente paso es asociar la IP Elástica con la instancia de servidor creada. Para esto se selecciona la opción “*Elastic IPs*”, se elige “*Allocate new address*” seguido de “*EC2*” que es el tipo de instancia, y se establece las opciones respectivas. Con esto se facilita el uso de transferencia de archivos y servicios web, ya que se apunta hacia una dirección IP fija.

Associate address

Select the instance OR network interface to which you want to associate this Elastic IP address (18.221.244.121)

Resource type Instance **i**
 Network interface

Instance i-0ad10aa3ec5654219 **↕**

Private IP 172.31.44.108 **↕** **i**

Reassociation Allow Elastic IP to be reassociated if already attached **i**

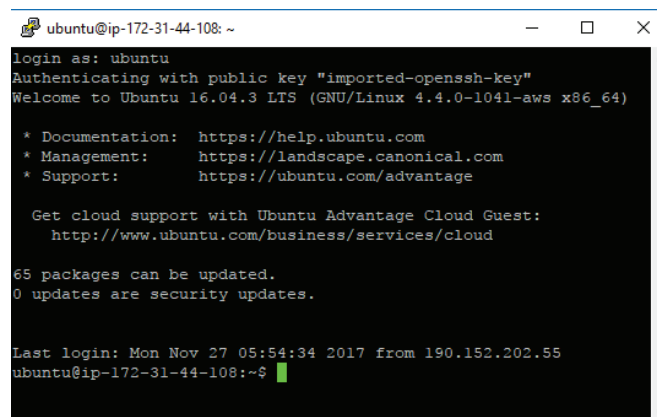
Figura 2.44 Asociación de IP Elástica con la Instancia.

La dirección IP elástica es una dirección IP estática y pública, que se utiliza en nubes dinámicas, para asociarla con cualquier tipo de instancia. Con esta IP elástica se puede ocultar los errores de las instancias mediante la reasignación de dicha dirección a una nueva instancia. No es necesario que personal técnico realice tal reasignación, la IP elástica posibilita realizar este cambio en un solo paso. Normalmente solo se admiten IPs elásticas para IPv4.

Finalmente cuando la instancia ya es creada se procede a ingresar al servidor mediante el protocolo SSH. Para esto se necesita la utilización del programa “*PuTTY*”. Antes se debe configurar la llave de seguridad sobre este programa, pero se debe tomar en cuenta que para poder conectarse a Amazon Web Services desde “*PuTTY*”, es necesario que la clave “.pem” esté en formato “.ppk”, lo cual reconoce “*PuTTY*”. Para convertir al nuevo formato se utiliza la herramienta “*PuTTY Key Generator*”.

Luego de generar el nuevo formato, se vuelve al programa “PuTTY” y sobre la sección de categorías se elige la ruta “Connection>SSH>Auth” y se escoge el archivo “servidorawsternica2017.ppk” generado previamente.

Una vez configurada la llave de seguridad sobre “PuTTY”, se accede a la instancia del servidor utilizando la dirección IP “18.221.244.121” y puerto “22” mediante SSH. Para acceder al servidor solamente se solicita el login, que en este caso es “ubuntu”.



```

ubuntu@ip-172-31-44-108: ~
login as: ubuntu
Authenticating with public key "imported-openssh-key"
Welcome to Ubuntu 16.04.3 LTS (GNU/Linux 4.4.0-1041-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

Get cloud support with Ubuntu Advantage Cloud Guest:
http://www.ubuntu.com/business/services/cloud

65 packages can be updated.
0 updates are security updates.

Last login: Mon Nov 27 05:54:34 2017 from 190.152.202.55
ubuntu@ip-172-31-44-108:~$

```

Figura 2.45 Conexión al servidor mediante SSH.

Para instalar el servidor web se utiliza el comando “*sudo apt-get install apache2*”, con esto se logra instalar el “servidor HTTP Apache”, que es un servidor web HTTP de código abierto multiplataforma que brinda la posibilidad de crear sitios web y servicios web. Al finalizar la instalación se verifica que el servidor funciona correctamente ingresando en cualquier navegador web la dirección IP Pública “18.221.244.121” como se muestra en la figura 2.46.

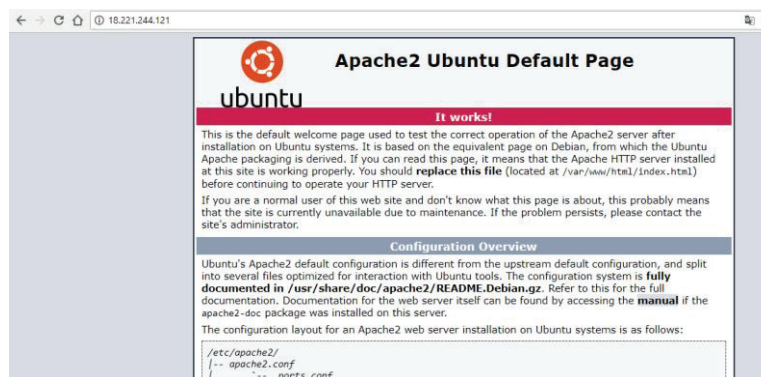


Figura 2.46 Servidor Web Apache2.

2.3.6.2 Desarrollo de la Aplicación Web

Para la generación de la aplicación web que alojará los archivos de imagen térmica capturados por el prototipo de cámara térmica, se utiliza varios lenguajes de programación como HTML, CSS, PHP y JavaScript.

En la instancia de servidor es necesario instalar el lenguaje de programación PHP, este lenguaje permite el desarrollo web de contenido dinámico. Por tanto para su instalación y el de otros componentes que facilitan su procesamiento se ejecuta el comando *“sudo apt-get install php libapache2-mod-php php-mcrypt php-mysql”*.

Por lo tanto para generar el diseño de la aplicación web se utiliza los lenguajes de programación, junto con el servidor HTTP Apache. Dicha aplicación web presenta una interfaz simple compuesta por 2 elementos.

El primer elemento se denomina “inicio”, el cual muestra el título del proyecto de cámara térmica junto con las imágenes de portada, y también se cuenta con el panel de navegación para dirigirse al repositorio que contiene las imágenes térmicas. Para acceder al sitio web que contiene este primer elemento se ingresa la dirección web <http://18.221.244.121/inicio.html>.



Figura 2.47 Página de Inicio de la Aplicación Web.

El segundo elemento se denomina “termicas”, y se utiliza para mostrar los archivos de imagen térmica enviados desde el prototipo de cámara térmica hacia el servidor web. Dichas imágenes se alojan en el directorio

“/var/www/html/repositorio/”. Se utiliza una galería dinámica para desplegar las imágenes térmicas sobre la aplicación web.

Para acceder al sitio web que contiene este segundo elemento, existe dos posibilidades: La primera ingresando a la dirección web <http://18.221.244.121/termicas.php>, y la segunda ingresando desde el panel de navegación del primer elemento.

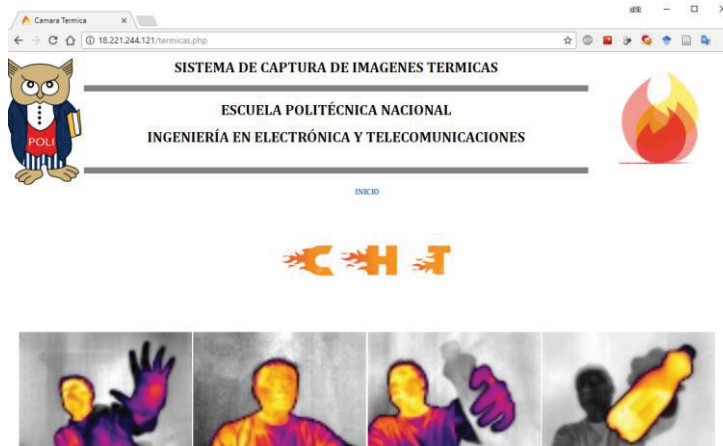


Figura 2.48 Página de almacenamiento de Imágenes Térmicas.

2.3.7 IMPLEMENTACIÓN DE FUENTE DE ALIMENTACIÓN

En el presente proyecto se utiliza la batería externa fabricada por la empresa “mophie” que proporciona un voltaje de salida de 5 [V] con amperajes de 2.1 [A] y 1 [A] según se requiera. Cuenta con una capacidad de carga de 5200 [mAh].



Figura 2.49 Dimensiones de la Batería [55].

Las dimensiones de la batería son primordiales, ya que el prototipo deberá ser lo más pequeño posible. Por lo tanto las dimensiones que presenta la batería son de 47,5 x 85,1 x 21,3 [mm]. Además se cuenta con 4 LEDs que informan el nivel de carga que posee la batería, con valores de 25%, 50%, 75% y 100% de nivel de carga, según el número de LEDs que se encuentren encendidos.

Para proveer de energía eléctrica al prototipo se utiliza un cable USB, con conector USB en la batería y conector micro USB en la placa Raspberry Pi 2B. Además se realiza un corte en el medio de este cable para instalar un interruptor que servirá para encender y apagar el prototipo de cámara térmica.



Figura 2.50 Cable de conexión Batería - Raspberry Pi 2B.

Para cargar la batería del prototipo se utiliza de igual manera un cable USB pero con conector micro USB en la batería y conector USB tipo hembra en la pared de la carcasa, de esta manera se podrá cargar el equipo con cualquier tipo de cargador USB.



Figura 2.51 Cable de conexión Batería – Cargador.

2.3.8 IMPLEMENTACIÓN DE CARCASA Y ENSAMBLAJE

Se procede a ensamblar todos los componentes del prototipo dentro de una carcasa de acrílico, luego de comprobar su buen funcionamiento y desempeño. El tamaño de la plancha de acrílico que se adquiere es de 60x60 [cm] con un espesor de 4 [mm]. El espesor permite que las paredes de la carcasa sean fuertes y resistentes.



Figura 2.52 Componentes del prototipo sobre la Plancha de Acrílico antes de cortar.

El diseño de las paredes que conforman la carcasa se realiza utilizando el programa “AutoCAD”, tomando en cuenta las respectivas medidas de cada elemento del prototipo. Se realiza los cortes de la carcasa en un servicio de cortadora láser. El diseño de la carcasa se asemeja al de una caja, compuesta por 6 paredes externas. Se muestra el diseño de las paredes en la imagen a continuación:

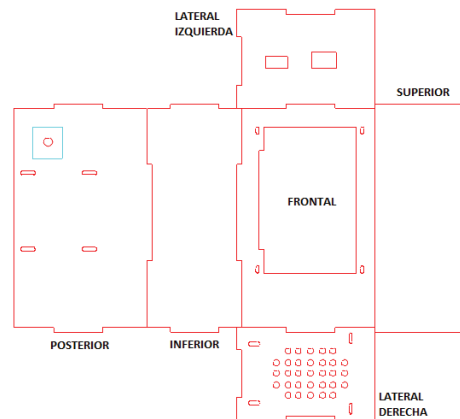


Figura 2.53 Diseño de las paredes de la carcasa.

Todas las paredes estarán fijas a excepción de dos; la pared superior que funciona como tapa y la pared lateral derecha que puede ser removida al retirar los 4 tornillos que la sujetan. Estas dos paredes de la carcasa servirán para ensamblar y desensamblar el prototipo.

El ensamblaje de los elementos que conforman el prototipo se distribuye de la siguiente manera:

- En la pared frontal se realiza un corte rectangular para posicionar la pantalla táctil y 4 cortes adicionales para fijar los tornillos que sujetarán dicha pantalla.



Figura 2.54 Pared Frontal.

- Sobre la pared posterior se ubica el sensor de cámara térmica LWIR Flir Lepton junto con la placa Raspberry Pi 2B. Para esto se realiza el corte de un orificio por donde el lente del sensor detectará las señales infrarrojas. El sensor se sujetará internamente sobre una sección cuadrada de acrílico acorde a su tamaño y la placa Raspberry Pi 2b se sujetará por medio de tornillos.



Figura 2.55 Pared Posterior.

- En la pared lateral izquierda se realiza 2 cortes que servirán para colocar el interruptor y el conector USB tipo hembra.



Figura 2.56 Pared Lateral Izquierda.

- En la pared lateral derecha se efectúa el corte de varios orificios sobre la zona central, que permitirán la ventilación del prototipo. También se realiza 4 cortes adicionales en las esquinas para posicionar los tornillos que brindarán la posibilidad de remover dicha pared.



Figura 2.57 Pared Lateral Derecha.

- La pared superior funcionará como la tapa del prototipo. Su movilidad dependerá de una bisagra sujeta entre ésta tapa y la pared lateral izquierda; además para permitir cerrar con seguridad se coloca un broche de presión en su extremo derecho.



Figura 2.58 Pared Superior.

- En la pared inferior se coloca el conector para el soporte del prototipo de cámara térmica. El soporte que se utilizará para el presente proyecto es el trípode conocido como “*Gorillapod*”, permitiendo maniobrar de forma segura el prototipo. Además sobre esta pared se ubicará internamente la batería y el módem USB.



Figura 2.59 Pared Inferior.

También se presenta la disposición interna de los equipos sobre la carcasa de acrílico en el gráfico a continuación:



Figura 2.60 Disposición Interna de los equipos.

Adicionalmente se muestra la forma en como el prototipo recarga su fuente de alimentación. Cabe mencionar que cuando se recarga la batería no es posible utilizar la cámara térmica.



Figura 2.61 Modo de recarga de batería.

Finalmente se tiene ensamblado todo el equipo y listo para realizar las respectivas pruebas de funcionamiento, detalladas en el siguiente capítulo de este documento. En la imagen a continuación se presenta el prototipo de cámara térmica totalmente ensamblado y funcional.



Figura 2.62 Prototipo de Cámara Térmica.

CAPÍTULO 3: PRUEBAS DE FUNCIONAMIENTO Y ANÁLISIS DE RESULTADOS

Para la realización de pruebas del prototipo se presentan varios escenarios en los cuales la visión infrarroja es de gran ayuda, destacando entre sus principales funciones la posibilidad de ofrecer una mejor visión a bomberos que ingresan en ambientes llenos de humo y de esta manera facilitar el rescate de personas atrapadas entre el fuego.

La cámara térmica permite distinguir objetos que presentan diferentes temperaturas, y dependiendo del tipo de material de cada objeto se obtendrá una mejor referencia de la imagen térmica, esta característica se conoce como emisividad del material. Por tanto se somete la cámara a distintos ambientes de medición con diferente tipo de materiales.

Otra característica presente al tratar con visión infrarroja es la temperatura reflejada, que produce la reflexión de radiación térmica sobre la superficie de un material, funcionando a modo de espejo, de esta manera también se somete el prototipo a esta clase de prueba.

En definitiva las pruebas de funcionamiento realizadas con el prototipo sobre diferentes ambientes son expuestas a continuación. Pero antes se informa la manera correcta de utilización del prototipo de cámara térmica, para evitar posibles daños sobre el equipo, detallando de igual forma el proceso para capturar una imagen y enviarla a la respectiva aplicación web.

3.1 SUGERENCIAS DE USO DEL PROTOTIPO DE CÁMARA TÉRMICA

La forma adecuada de utilizar el prototipo de cámara térmica se menciona en los pasos a continuación:

1. Para encender el equipo se debe presionar el interruptor. Los símbolos impresos sobre el interruptor significan encendido (I) y apagado (O).

2. Luego de unos segundos de haber accionado el interruptor se muestra en pantalla la aplicación para visualizar y capturar imágenes térmicas, pero aún no es posible enviar las capturas de imagen térmica a Internet.
3. El módem USB tarda en engancharse a la red celular alrededor de un minuto luego de haber encendido el equipo.
4. El botón táctil “ENVIAR” sirve para capturar la imagen térmica y enviarla al sitio web.
5. Para verificar que la imagen térmica fue enviada correctamente, se ingresa al sitio web <http://18.221.244.121/inicio.html>.
6. En el caso que no se encuentre la imagen deseada en la aplicación web se utiliza el botón táctil “REENVIAR”, para volver a enviar el(los) archivo(s) de imagen capturado(s) previamente.
7. Para observar las imágenes térmicas localmente se presiona el botón táctil “ARCHIVO”. De esta manera se ingresa al repositorio local de los archivos de imagen.
8. Para reiniciar el equipo se presiona el botón táctil “REINICIAR”, que cuenta con un ícono de color azul para mejor referencia.
9. Para apagar el equipo se presiona el botón táctil “APAGAR”, que cuenta con un ícono de color rojo para mejor referencia.
10. Luego de apagar el prototipo mediante su respectivo botón táctil, se debe esperar unos segundos hasta que el sistema se cierre por completo y después se presiona el interruptor dejándolo en la opción (O).
11. Cuando los LEDs de la batería se apagan totalmente significa que el equipo está apagado correctamente, se puede verificar esto abriendo la tapa del equipo.

NOTA: Se debe tomar en cuenta que para cargar la batería del equipo, solamente es posible cuando el prototipo se encuentra apagado.

3.2 PRUEBA 1: SIMULACIÓN DE INCENDIO

Esta prueba se realiza en la ciudad de Pimampiro - Imbabura, en un ambiente controlado, ideal para la realización de un pequeño incendio, sin generar peligro

alguno. El objetivo de esta prueba es comprobar que la cámara térmica permite observar personas a través del humo generado por las llamas.

La prueba se fundamentó en crear una fogata y luego agregar material que produzca bastante humo para que de esta manera se simule lo que ocurre durante un incendio. Se ubica a una persona por detrás de la columna de humo generada por la fogata y se procede a utilizar el prototipo de cámara térmica.

Al inicio de la prueba no se cuenta con el suficiente humo para cubrir a la persona como se puede observar en la figura 3.1.



Figura 3.1 Inicio de la prueba de incendio.

Después de transcurrir cierto tiempo la columna de humo se hace más espesa y el ojo humano es incapaz de percibir a la persona que se encuentra por detrás del humo. Al utilizar la cámara térmica se logra obtener una apreciación de la silueta de calor que emite el cuerpo de la persona, dichos escenarios se muestran en la figura 3.2.



Figura 3.2 Imagen Normal y Térmica de la prueba de incendio.

Al momento de realizar esta prueba se verificó que cuando se enfoca la cámara térmica directamente a las llamas no se logra observar con facilidad a la persona que se ubica detrás, por lo tanto para una mejor apreciación del ambiente se retira un poco del cuadro de imagen a las llamas.

Se debe tomar en cuenta que la prueba de incendio se realizó en un ambiente abierto, pero normalmente los incendios se producen en ambientes cerrados en los cuales el calor corporal de las personas aumenta, y por tanto se podrá distinguir la silueta del cuerpo con mejor claridad.

3.3 PRUEBA 2: DIFERENCIA DE TEMPERATURA

El prototipo brinda la posibilidad de distinguir objetos con distintos grados de temperatura. Para interpretar estas diferencias de temperatura se presenta las imágenes térmicas en cierta gama de colores.

Los tonos en amarillo muestran los objetos con mayor temperatura, seguidos por tonos en violeta y azul que informan un menor grado de temperatura, hasta llegar al color negro. Los objetos fríos o con menor temperatura frente a otros objetos se presentan en tonos grises, partiendo desde tonos grises opacos hasta los grises más claros que mostrarán los objetos más fríos de cada imagen térmica.

Por lo tanto la gama de colores para las imágenes termicas se rige a un patrón de caliente a frío en el orden de los colores amarillo, violeta, azul, negro y gris respectivamente. La escala de temperaturas según el color se puede apreciar en la imagen a continuación:

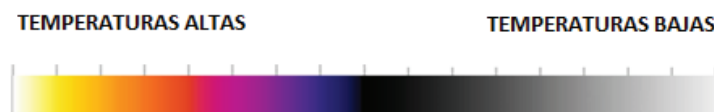


Figura 3.3 Escala de temperaturas según el color.

Para la prueba de diferencia de temperatura se utiliza en el primer caso dos botellas compuestas de agua tanto fría como caliente. En la imagen normal no

se nota mayor diferencia entre ambas botellas, pero en la imagen térmica se puede visualizar con facilidad su diferencia de temperatura. Se muestra los resultados de esta prueba en la figura 3.4



Figura 3.4 Caso #1 de diferencia de temperatura.

En el segundo caso se utiliza 3 vasos de plástico llenos de agua fría, tibia y caliente. Como resultado de esta prueba se puede ver en la figura 3.5 que el vaso del medio que contiene el agua tibia es difícil de distinguir.



Figura 3.5 Caso #2 de diferencia de temperatura.

Esta prueba se utiliza para apreciar el grado de sensibilidad que posee el sensor de cámara térmica, y sirve como precaución de que no se puede distinguir con facilidad objetos con temperaturas intermedias.

3.4 PRUEBA 3: REFLEXIÓN DE MATERIALES

Cierto tipo de materiales poseen la característica de reflejar la radiación térmica del mismo modo que un espejo refleja la luz visible. Estas reflexiones pueden ocasionar una interpretación incorrecta de la imagen térmica.

Para esta prueba se verifica la característica de reflexión enfocando el prototipo hacia una ventana de vidrio, la cual normalmente se observa transparente, pero al someterse a la visión infrarroja actúa a manera de espejo como se puede observar en la figura 3.6.

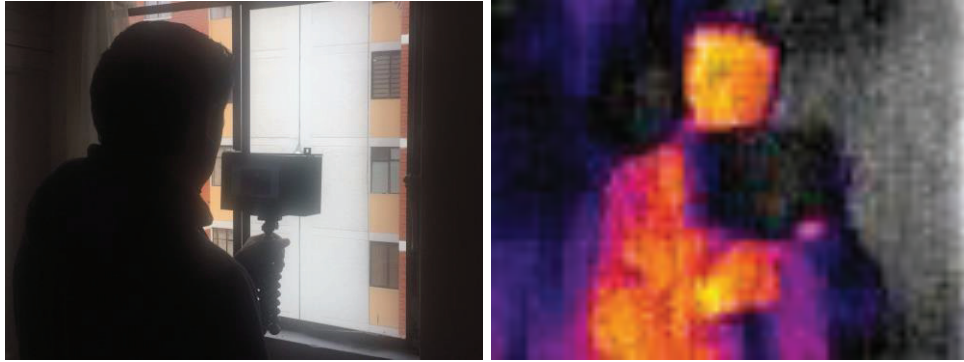


Figura 3.6 Imagen Normal y Térmica, reflexión del vidrio.

3.5 PRUEBA 4: INSPECCIÓN DE ESTRUCTURAS

La temperatura en las superficies puede proporcionar gran cantidad de información sobre los elementos estructurales. Los problemas que podría tener una construcción resultan invisibles a simple vista, pero al utilizar la visión infrarroja se podrían notar evidentes. Las características estructurales que se ocultan tras los muros podrían ser detectadas gracias a la visión infrarroja.

Durante esta prueba se inspeccionó el techo de una habitación ubicada bajo la terraza de una vivienda. La prueba se realizó en horas de la tarde durante un día soleado, para comprobar el estado de la estructura de los cimientos.

Por medio de la visión infrarroja se logró apreciar el recalentamiento de las vigas de hierro que conforman el techo, comprobando que la estructura se encuentra en estado normal, sin mostrar ninguna anomalía. Se muestra el resultado de esta prueba en la figura 3.7.

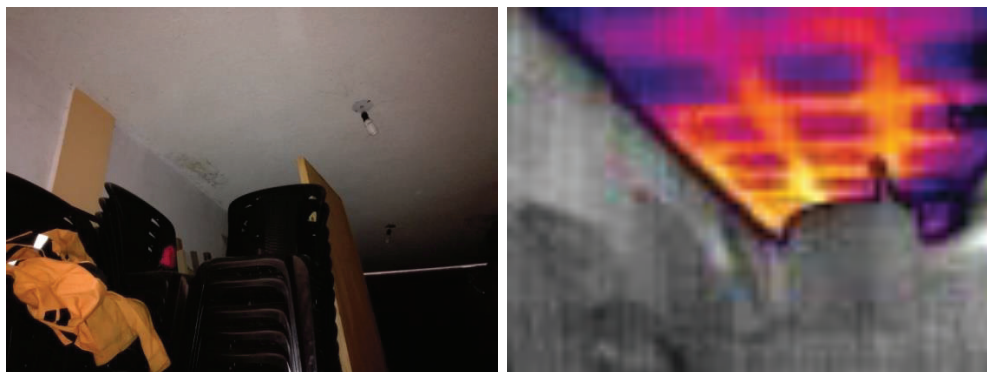


Figura 3.7 Imagen Normal y Térmica, inspección estructural.

3.6 PRUEBA 5: INSPECCIÓN DE HUMEDAD

La humedad puede penetrar por las uniones y grietas de tejados, techos y muros, ocasionando a largo plazo la generación de moho estructural y un olor desagradable.

Para comprobar el rastro de humedad, se inspecciona el piso de una casa. En dicha zona se encuentra ubicado un lavabo, el cual provocará cotidianamente salpicaduras de agua, lo que resulta en acumulación de humedad en el piso, que a simple vista no puede ser apreciada. Se procede a inspeccionar la zona del piso con la cámara térmica y se logra observar el rastro de humedad, como se muestra en la figura 3.8.

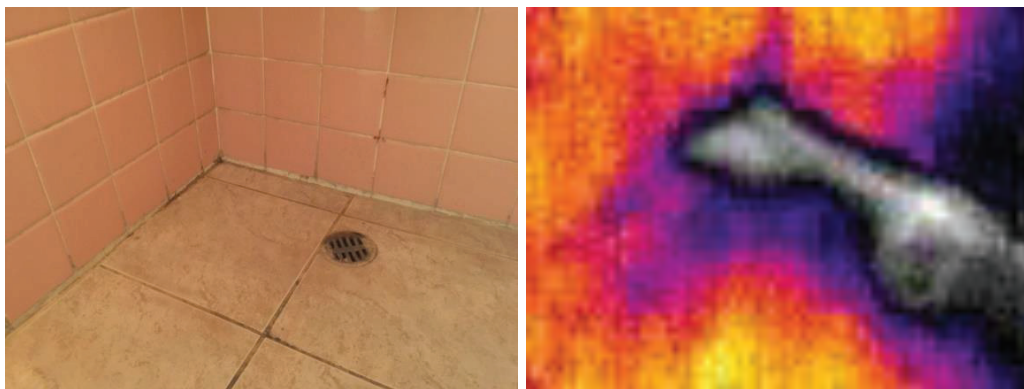


Figura 3.8 Imagen Normal y Térmica, inspección de humedad.

3.7 PRUEBA 6: INSPECCIÓN ELÉCTRICA

Los cables eléctricos en mal estado podrían provocar sobrecalentamiento en las instalaciones eléctricas, y por lo tanto ocasionar cortes de energía, daños en los equipos e inclusive riesgos para la seguridad al desencadenar incendios. El mantenimiento preventivo de estas instalaciones servirá para prevenir estos sucesos. La visión infrarroja sirve para la inspección de estas zonas y de esta manera evitar daños futuros.

En esta prueba se verifica la conexión de varios cables eléctricos que mediante la inspección con la cámara térmica se logra apreciar el recalentamiento que presentan (figura 3.9). El hecho de que la conexión se encuentre sobre el piso, supone un escenario propicio para la generación de un incendio, ya que dichos cables podrían estar ubicados junto a objetos inflamables.

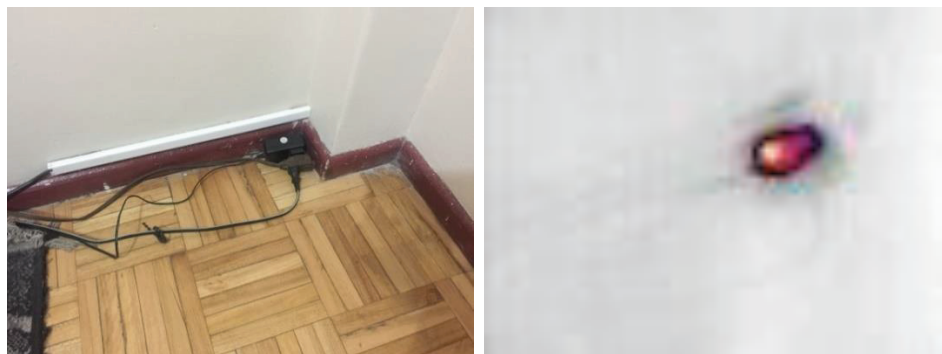


Figura 3.9 Imagen Normal y Térmica, inspección eléctrica.

3.8 PRUEBA 7: INSPECCIÓN DE EQUIPOS ELECTRÓNICOS

Los equipos tecnológicos están compuestos de varios elementos, destacando entre ellos los componentes electrónicos, los cuales se recalientan al encontrarse en funcionamiento, al cabo de cierto tiempo estas partes se deterioran y ya no funcionan de igual manera, lo que provoca un aumento de temperatura de todo el equipo e inclusive en ocasiones el exceso de temperatura es tan alto, que ocasiona el daño total del equipo.

Para esta prueba se monitorea con la cámara térmica un computador portátil, para constatar el comportamiento de temperatura de su estructura. Se verifica

que la parte inferior de la pantalla, así como el lado izquierdo del computador presentan un mayor grado de temperatura, dando a suponer que sobre este lugar se encuentra ubicado el procesador del equipo, el cual emite gran cantidad de calor. Se puede observar esta situación en la figura 3.10.

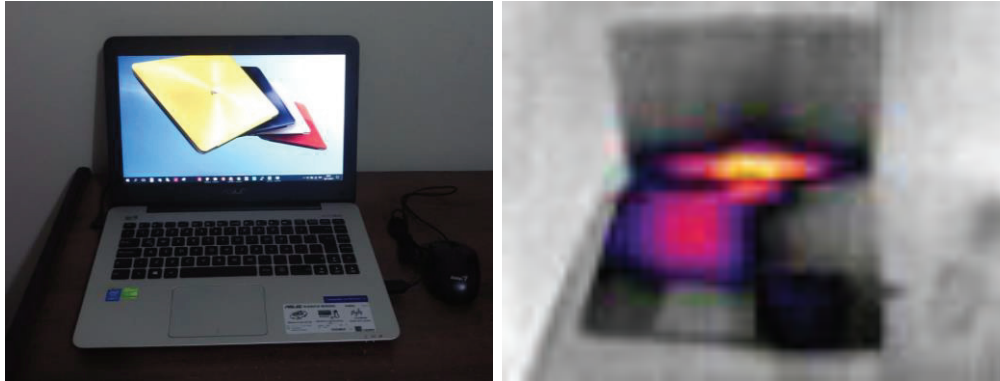


Figura 3.10 Imagen Normal y Térmica, inspección electrónica.

3.9 PRUEBA 8: INSPECCIÓN MECÁNICA

La revisión de partes mecánicas con visión infrarroja permite detectar información importante acerca del estado del equipamiento. En el sector industrial existe gran variedad de procesos que provocan recalentamiento del equipo y con una detección temprana se puede arreglar cualquier tipo de anomalías.

Durante esta prueba se sometió el prototipo a la inspección de un vehículo, para realizar el monitoreo de los frenos en la llanta. Primeramente se revisó el estado de los frenos cuando el vehículo estuvo en reposo, y como era de esperarse no se observó ninguna presencia de calor (figura 3.11).

Al repetir el monitoreo luego de un viaje de 3 horas atravesando la ruta Pimampiro - Quito se pudo constatar el cambio de temperatura que sufren dichos frenos (figura 3.11).

Esta clase de inspecciones podrían servir para la prevención de accidentes, ya que al observar que la temperatura es excesivamente alta, se puede actuar a tiempo para realizar un mantenimiento al vehículo. En el caso de esta inspección

se llegó a la conclusión de que la temperatura sobre los frenos de la llanta era la normal para un viaje de este tipo.

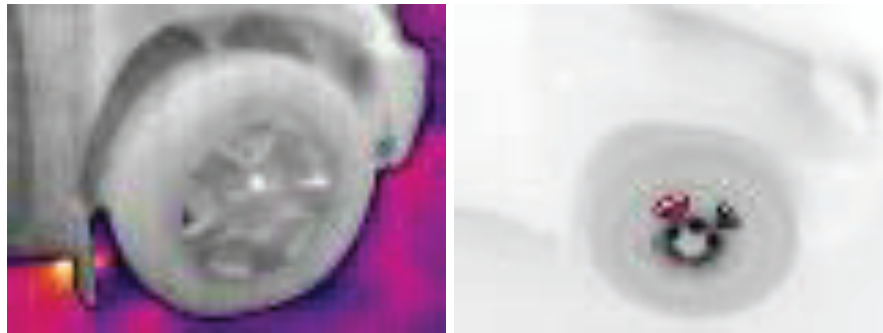


Figura 3.11 Imagen Normal y Térmica, inspección mecánica.

3.10 PRUEBA 9: VISIÓN NOCTURNA

Las cámaras de visión infrarroja permiten detectar la radiación térmica de los cuerpos y traducirla en una imagen visible para el ojo humano. Funcionan independientemente de las condiciones de iluminación que presente el ambiente, es decir no existe problema al utilizar la cámara durante la noche.

Otra característica importante es que las cámaras térmicas permiten ver a través del humo y de condiciones climáticas como la neblina y el polvo. La radiación térmica penetra mejor que la luz visible a través de los objetos que oscurecen la atmósfera.

La prueba de visión nocturna consiste en lograr observar a una persona durante la noche en un ambiente con luz visible mínima. Primeramente se puede observar en la figura 3.12 el ambiente visible, debido a la presencia de dos focos, que permiten distinguir a la persona en la escena.



Figura 3.12 Imagen normal con luz visible.

Luego se apaga los focos del ambiente y como resultado se tiene que la silueta de la persona en la escena es casi imperceptible al ojo humano. Pero al utilizar el prototipo de cámara térmica para monitorear el ambiente, se comprueba que la persona en la escena es fácilmente visible gracias a la radiación térmica que emite como se muestra en la figura 3.13.

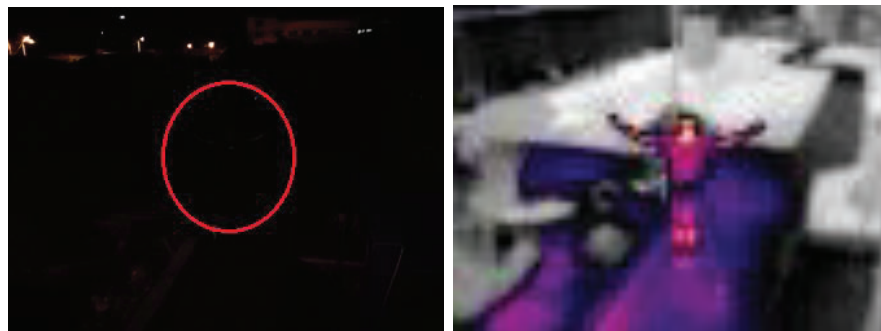


Figura 3.13 Imagen Normal y Térmica, visión nocturna.

3.11 PRUEBA 10: TRANSFERENCIA DE ARCHIVOS

El sistema para la transferencia de archivos hacia la aplicación web está constituido por varios componentes, como el módem USB, el protocolo de transferencia de archivos, el comando utilizado para la transferencia y el servidor remoto que acoge los archivos enviados.

Para verificar que los archivos se enviaron correctamente, se comparó el directorio local de la Raspberry Pi 2B, con el repositorio remoto del servidor después de realizar de varias capturas con el prototipo.

Se pudo verificar en la aplicación web que no se mostraban todas las imágenes que habían sido capturadas, por lo tanto, se utilizó el botón táctil que permite reenviar las capturas y luego se comprobó nuevamente que las imágenes térmicas estaban actualizadas y en su totalidad (205 elementos). Se comprueba esta situación al observar la cantidad de archivos que contiene tanto el directorio local (figura 3.14) como el directorio remoto (figura 3.15).

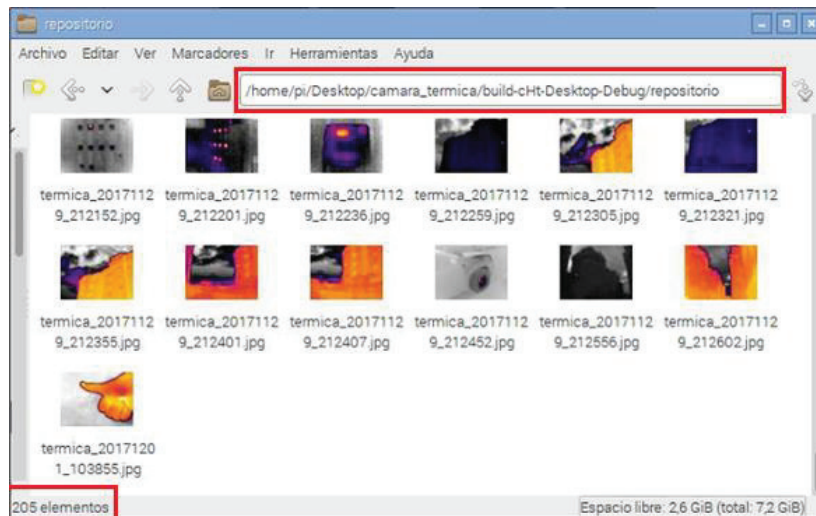


Figura 3.14 Directorio Local.

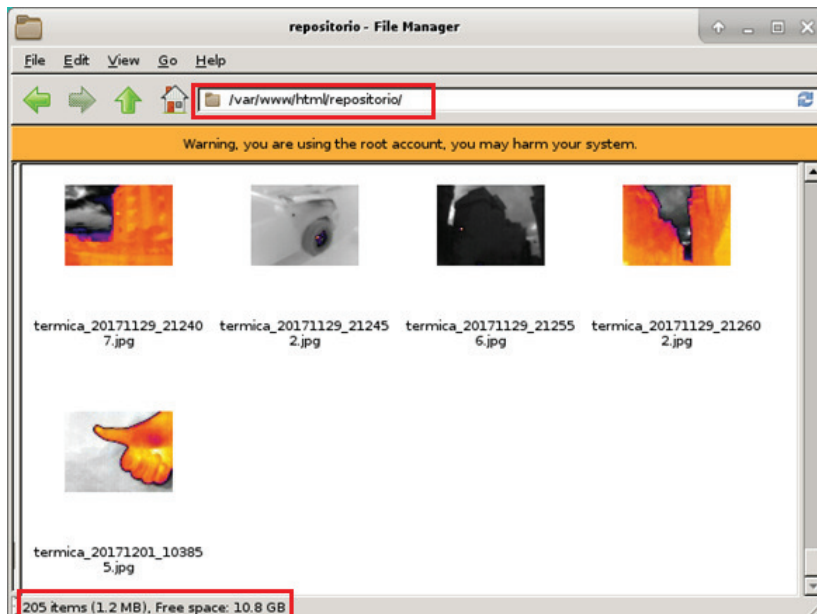


Figura 3.15 Directorio Remoto.

CAPÍTULO 4: CONCLUSIONES Y RECOMENDACIONES

A continuación se detallan las respectivas conclusiones y recomendaciones obtenidas en base a la realización del proyecto.

4.1 CONCLUSIONES

- El sensor de cámara térmica *LWIR Flir Lepton* permite detectar la radiación térmica de los cuerpos para poder transformarla en imagen visible al ojo humano, se constató que al monitorear el ambiente las imágenes térmicas presentan una gama de colores, en la cual predomina el color amarillo de un objeto caliente, y si dicha temperatura es lo suficientemente alta en comparación al resto de objetos del cuadro de captura, produce que estos objetos sean difíciles de diferenciar ya que adquieren un color gris claro.
- El sensor de cámara térmica utilizado para la elaboración del prototipo cuenta con una resolución de 80 x 60 píxeles, lo cual genera cierta dificultad al momento de monitorear el ambiente, ya que no se logra diferenciar con gran precisión los distintos objetos que lo conforman.
- El programa QT Creator es una herramienta óptima para el desarrollo de aplicaciones con Interfaz Gráfica de Usuario, se caracteriza por ser un software libre y multiplataforma. Cuenta con una versión gratuita que contiene una extensa librería para la realización de distintos proyectos y para el caso de procesamiento de imágenes dichas librerías son de gran utilidad. Es posible ingresar al sitio web <http://doc.qt.io/qt-5/classes.html> para observar en mejor detalle el funcionamiento de cada librería, o también se puede utilizar la sección de ayuda contenida dentro del mismo programa.
- La pantalla táctil utilizada para el prototipo es de tipo resistiva, por lo tanto se necesita realizar una gran presión sobre la misma para poder accionar sus respectivas funciones; además no es posible usar varios puntos de

presión al mismo tiempo, imposibilitando funciones como el acercamiento de pantalla.

- El diseño inicial del conector para el soporte de la carcasa sufrió un quiebre durante las pruebas del prototipo, debido a la forma frágil que presentaba y por el respectivo peso del prototipo; por lo tanto se reemplazó por un conector de mejor diseño, adherido a las paredes de la carcasa, ofreciendo de tal manera un buen soporte y estabilidad al prototipo de cámara térmica.
- El prototipo de cámara térmica permite la diferenciación de temperatura de los objetos en base a una gama de colores conformada por el amarillo, violeta, azul, negro, hasta los colores grises, lo que representa objetos desde temperaturas altas hacia las temperaturas bajas respectivamente, pero cabe recalcar que el prototipo no brinda la posibilidad de medir la magnitud de temperatura de dichos objetos.

4.2 RECOMENDACIONES

- El sensor de cámara térmica podría ser reemplazado por un sensor con mejores prestaciones como por ejemplo una mejor resolución, es decir de 160 x 120 pixeles, brindando de tal manera la posibilidad de obtener imágenes de mayor calidad y así poder diferenciar con mejor precisión los objetos analizados.
- Para tener una referencia de la temperatura que posee la superficie de un objeto se aconseja posicionar una mano cerca del objeto que se esté monitoreando y de esta manera poder comparar la temperatura del objeto con el calor corporal.
- No es recomendable utilizar el prototipo de cámara térmica bajo la lluvia, ya que la carcasa no es totalmente hermética, y puede ocasionar que el equipo se dañe al tener contacto con el agua.

- Se puede cambiar la pantalla táctil del prototipo por una de tipo capacitiva, para obtener una mejor sensibilidad a la presión de los dedos, ya que su funcionamiento se basa en detectar la capacitancia de un cuerpo, es decir, su carga eléctrica. Para realizar dicho cambio se debe tomar en cuenta el tamaño de la pantalla, con el objetivo de que pueda caber sobre el corte de la carcasa. (La pantalla utilizada en el prototipo es de 5 pulgadas).
- Para obtener una opción de conexión inalámbrica adicional a la del módem USB, se puede acoplar un adaptador WiFi compatible con la placa Raspberry Pi 2B. Esta recomendación sirve para no consumir los datos de la red celular y para mejor comodidad al momento de reprogramar el equipo, ya que el hecho de conectarse mediante una red cableada resulta incómodo.
- La batería del equipo puede ser reemplazada por una con mayor cantidad de carga útil, para que de esta manera se obtenga una mayor autonomía de trabajo.
- Se debe apagar el equipo según los pasos descritos en el manual de uso, ya que el hecho de apagar bruscamente puede ocasionar desperfectos sobre el prototipo.
- La batería del prototipo solamente se puede recargar cuando el equipo no se encuentra en funcionamiento.
- Se podría implementar un módulo GPS (*Global Positioning System*, Sistema de Posicionamiento Global) sobre el prototipo de cámara térmica, para mostrar la ubicación de cada una de las imágenes que se capturen en función de sus datos de latitud y longitud.

REFERENCIAS

- [1] Teleformación. *Ondas Electromagnéticas*. [En línea]. Disponible en: [http://teleformacion.edu.aytolacoruna.es/FISICA/document/fisicaInteractiva/Onda sbachillerato/ondasEM/ondasEleMag_indice.htm](http://teleformacion.edu.aytolacoruna.es/FISICA/document/fisicaInteractiva/Onda%20bachillerato/ondasEM/ondasEleMag_indice.htm) [Último acceso: 04 de Octubre de 2016].
- [2] Radiansa. *¿Que son los campos eléctricos y magnéticos de baja frecuencia?* [En línea]. Disponible en: <http://www.radiansa.com/contaminacion-electromagnetica/campos-electromagneticos/campos-electricos-magneticos.htm> [Último acceso: 04 de Octubre de 2016].
- [3] Unicrom. *Composición onda electromagnética*. [En línea]. Disponible en: <http://unicrom.com/composicion-onda-electromagnetica/> [Último acceso: 04 de Octubre de 2016].
- [4] Vix. *El espectro visible de luz*. [En línea]. Disponible en: <https://www.vix.com/es/btg/curiosidades/2011/10/02/el-espectro-visible-de-luz> [Último acceso: 05 de Octubre de 2016].
- [5] Marbelo. *Tutorial de Teledetección*. [En línea]. Disponible en: <https://marbelo.webs.ull.es/rs2.pdf> [Último acceso: 05 de Octubre de 2016].
- [6] Khanacademy. *La luz: ondas electromagnéticas, espectro electromagnético y fotones*. [En línea]. Disponible en: <https://es.khanacademy.org/science/physics/light-waves/introduction-to-light-waves/a/light-and-the-electromagnetic-spectrum> [Último acceso: 07 de Octubre de 2016].
- [7] Espectrometría. *Espectro electromagnético*. [En línea]. Disponible en: http://www.espectrometria.com/espectro_electromagntico [Último acceso: 11 de Octubre de 2016].
- [8] Astrofísica y Física. *¿Qué es el Espectro Electromagnético?* [En línea]. Disponible en: <http://www.astrofiscayfisica.com/2012/06/que-es-el-espectro-electromagnetico.html> [Último acceso: 11 de Octubre de 2016].

- [9] UAEM. *Espectro Electromagnético*. [En línea]. Disponible en: <https://sites.google.com/a/uaem.mx/satelites-artificiales/espectro-electromagnetico> [Último acceso: 11 de Octubre de 2016].
- [10] Alpoma. *Termografía*. [En línea]. Disponible en: <http://alpoma.net/cart0/?p=431> [Último acceso: 12 de Octubre de 2016].
- [11] Hyperphysics. *Radiación de Cuerpo Negro*. [En línea]. Disponible en: <http://hyperphysics.phy-astr.gsu.edu/hbasees/mod6.html> [Último acceso: 12 de Octubre de 2016].
- [12] López Dóriga. *La ciencia de las terapias de calor infrarrojo lejano y la eliminación de toxinas*. [En línea]. Disponible en: <https://lopezdoriga.com/vida-y-estilo/la-ciencia-de-las-terapias-de-calor-infrarrojo-lejano-y-la-eliminacion-de-toxinas/> [Último acceso: 12 de Octubre de 2016].
- [13] Física Cuántica. *Espectro Electromagnético*. [En línea]. Disponible en: <https://edbar01.wordpress.com/segundo-corte/espectro-electromagnetico/> [Último acceso: 12 de Octubre de 2016].
- [14] Xenics. *TECNOLOGÍAS INFRARROJAS*. [En línea]. Disponible en: <http://www.xenics.com/es/tecnolog%C3%ADas-infrarrojas> [Último acceso: 13 de Octubre de 2016].
- [15] Sinais. *Termografía infrarroja*. [En línea]. Disponible en: <http://www.sinais.es/Recursos/Curso-vibraciones/intro/termografia.html> [Último acceso: 13 de Octubre de 2016].
- [16] Amperis. *Principios básicos de la termografía*. [En línea]. Disponible en: <http://www.amperis.com/productos/camaras-termograficas/#termografia> [Último acceso: 15 de Octubre de 2016].
- [17] Electrónica Básica. *¿Qué es un bolómetro?* [En línea]. Disponible en: <http://www.electronica-basica.com/bolometro.html> [Último acceso: 18 de Octubre de 2016].
- [18] Construmática. *Microbolómetro*. [En línea]. Disponible en: <http://www.construmatica.com/construpedia/Microbol%C3%B3metro> [Último acceso: 18 de Octubre de 2016].

[19] Techtransfer. *Infrared Thermography and Predictive Maintenance*. [En línea]. Disponible en: <https://www.techtransfer.com/blog/infrared-thermography-predictive-maintenance/> [Último acceso: 20 de Octubre de 2016].

[20] Sofradir. *Uncooled detectors achieve*. [En línea]. Disponible en: <http://www.sofradir-ec.com/wp-uncooled-detectors-achieve.asp> [Último acceso: 21 de Octubre de 2016].

[21] Gisiberica. *CTR200*. [En línea]. Disponible en: <http://www.gisiberica.com/camaras%20termicas/CTR200.htm> [Último acceso: 21 de Octubre de 2016].

[22] Mundohvacr. *Ventajas de la Termografía*. [En línea]. Disponible en: <https://www.mundohvacr.com.mx/mundo/2015/01/ventajas-de-la-termografia/> [Último acceso: 26 de Octubre de 2016].

[23] Flir Media. *Guía de termografía para mantenimiento predictivo*. [En línea]. Disponible en: http://www.flirmedia.com/MMC/THG/Brochures/T820264/T820264_ES.pdf [Último acceso: 26 de Octubre de 2016].

[24] Extech Instruments. *Termómetro de infrarrojos de doble láser*. [En línea]. Disponible en: <http://www.extech.com.es/instruments/product.asp?catid=62&prodid=563> [Último acceso: 27 de Octubre de 2016].

[25] Amada Miyachi. *Thermal Conductivity*. [En línea]. Disponible en: <http://spanish.amadamiyachi.com/glossary/glossthermalconductivity> [Último acceso: 27 de Octubre de 2016].

[26] Alicia in Chemistry. *CONDUCTIVIDAD TERMICA*. [En línea]. Disponible en: <http://aliciainchemistry.blogspot.com/2015/11/conductividad-termica.html> [Último acceso: 28 de Octubre de 2016].

[27] Termografía Básica. *Termografía*. [En línea]. Disponible en: <http://termografiabasica.blogspot.com/2015/10/termografia-nivel-1.html> [Último acceso: 01 de Noviembre de 2016].

[28] Slideplayer. *Midiendo emisividad desconocida*. [En línea]. Disponible en: <http://slideplayer.es/slide/3395189/17/images/48/Midiendo+emisividad+desconocida.jpg> [Último acceso: 01 de Noviembre de 2016].

[29] YouTube. *FLASHOVER. GRABADO CON CAMARA TERMICA. TEPESA 2010*. [En línea]. Disponible en: <https://www.youtube.com/watch?v=z3v68WbOnQ0> [Último acceso: 07 de Noviembre de 2016].

[30] Xataka. *Raspberry Pi*. [En línea]. Disponible en: <https://www.xataka.com/tag/raspberry-pi> [Último acceso: 07 de Noviembre de 2016].

[31] Computer Hoy. *Raspberry Pi: ¿Qué modelo me compro?* [En línea]. Disponible en: <http://computerhoy.com/noticias/hardware/raspberry-pi-que-modelo-me-compro-23811> [Último acceso: 08 de Noviembre de 2016].

[32] Cómo hacer. *Comparativa y análisis: Raspberry Pi vs competencia*. [En línea]. Disponible en: <https://comohacer.eu/comparativa-y-analisis-raspberry-pi-vs-competencia/> [Último acceso: 08 de Noviembre de 2016].

[33] Somos Binarios. *Raspberry pi: 1 Primeros pasos*. [En línea]. Disponible en: <https://www.somosbinarios.es/raspberry-pi-1-primeros-pasos/> [Último acceso: 09 de Noviembre de 2016].

[34] Espberry. *Análisis de Raspbian: Programas y Características*. [En línea]. Disponible en: <https://espberry.wordpress.com/2016/03/07/instalar-mis-aplicaciones-en-raspbian/> [Último acceso: 09 de Noviembre de 2016].

[35] Gen Beta. *PIXEL Desktop, el nuevo entorno para Raspbian en Raspberry Pi*. [En línea]. Disponible en: <https://www.genbeta.com/sistemas-operativos/pixel-desktop-el-nuevo-entorno-para-raspbian-en-raspberry-pi> [Último acceso: 09 de Noviembre de 2016].

[36] Redalyc. *Telefonía móvil celular: origen, evolución, perspectivas*. [En línea]. Disponible en: <http://www.redalyc.org/articulo.oa?id=181517913002> [Último acceso: 11 de Noviembre de 2016].

- [37] Slideshare. *Telefonía Móvil*. [En línea]. Disponible en: <https://es.slideshare.net/wladi13/telefona-mvil-15640056> [Último acceso: 11 de Noviembre de 2016].
- [38] Slideshare. *Comunicaciones Inalámbricas. Telefonía Celular*. [En línea]. Disponible en: <https://es.slideshare.net/jesuscarr19/tecnologas-celulares> [Último acceso: 14 de Noviembre de 2016].
- [39] Mail x Mail. *Telefonía móvil celular: origen, evolución, perspectivas*. [En línea]. Disponible en: <http://imagenes.mailxmail.com/cursos/pdf/7/telefoniacelular-movil-funcionamiento-generaciones-24457.pdf> [Último acceso: 15 de Noviembre de 2016].
- [40] Redes Telemáticas. *La transmisión de datos en la telefonía móvil*. [En línea]. Disponible en: <http://redestelematicas.com/la-transmision-de-datos-en-la-telefoniamovil/> [Último acceso: 16 de Noviembre de 2016].
- [41] GSM Spain. *Glosario de términos relacionados con tecnología*. [En línea]. Disponible en: <https://www.gsmspain.com/glosario/?palabra=HSDPA> [Último acceso: 17 de Noviembre de 2016].
- [42] Telefónica. *GPRS*. [En línea]. Disponible en: http://www.telefonica.com.sv/Empresas/svas_gprs.html [Último acceso: 21 de Noviembre de 2016].
- [43] Raspberry Pi. *RASPBERRY PI 2 MODEL B*. [En línea]. Disponible en: <https://www.raspberrypi.org/products/raspberry-pi-2-model-b/> [Último acceso: 22 de Noviembre de 2016].
- [44] Raspberry Shop. *Hardware Raspberry Pi*. [En línea]. Disponible en: <https://www.raspberrishop.es/hardware-raspberry-pi.php> [Último acceso: 24 de Noviembre de 2016].
- [45] GitHub. *Groupgets*. [En línea]. Disponible en: <https://github.com/groupgets/LeptonModule> [Último acceso: 25 de Noviembre de 2016].

[46] Xataka Home. *Probamos la nueva Raspberry Pi 2: A fondo*. [En línea]. Disponible en: <https://www.xatakahome.com/trucos-y-bricolaje-smart/probamos-la-nueva-raspberry-pi-2-a-fondo> [Último acceso: 26 de Noviembre de 2016].

[47] Sparkfun. *FLiR Dev Kit*. [En línea]. Disponible en: <https://www.sparkfun.com/products/13233> [Último acceso: 01 de Diciembre de 2016].

[48] GitBooks. *“Francisco Moya: Taller de Raspberry Pi”*. [En línea]. Disponible en: <https://franciscomoya.gitbooks.io/taller-de-raspberry-pi/content/es/> [Último acceso: 01 de Diciembre de 2016].

[49] Di y Makers. *USANDO EL PUERTO GPIO*. [En línea]. Disponible en: <http://diymakers.es/usando-el-puerto-gpio/> [Último acceso: 08 de Diciembre de 2016].

[50] E. R. Tatayo Vinueza, *Implementación de una mini-red de sensores inalámbricos para detección temprana de incendios forestales utilizando hardware y software open source*, Quito - Ecuador: EPN, 2017.

[51] Waveshare. *5inch HDMI LCD (B), 800×480, supports various systems*. [En línea]. Disponible en: <https://www.waveshare.com/5inch-HDMI-LCD-B.htm> [Último acceso: 12 de Diciembre de 2016].

[52] R. A. Escandón Vincent, *Implementación de un prototipo para la obtención de imágenes térmicas geo-referenciadas desde el dron dji phantom 3 standard*, Quito - Ecuador: EPN, 2017.

[53] LG blog. *¿Qué significa que una pantalla sea tipo “IPS”? ¿Por qué es mejor?* [En línea]. Diciembre en: <http://www.lgblog.cl/reviews/que-significa-que-una-pantalla-sea-tipo-ips-por-que-es-mejor/> [Último acceso: 13 de Diciembre de 2016].

[54] Hipertextual. *IPS vs AMOLED: explicación, ventajas e inconvenientes*. [En línea]. Disponible en: <https://hipertextual.com/archivo/2014/03/ips-vs-amoled/> [Último acceso: 15 de Diciembre de 2016].

- [55] Mophie. *Power reserve 2X*. [En línea]. Disponible en: <http://www.mophie.com/shop/power-reserve-2X> [Último acceso: 05 de Enero de 2017].
- [56] Raspberry Pi. *DOWNLOADS*. [En línea]. Disponible en: <https://www.raspberrypi.org/downloads/> [Último acceso: 07 de Enero de 2017].
- [57] Uno Cero. *Raspbian: Sistema operativo gratuito para la Raspberry Pi*. [En línea]. Disponible en: <https://www.unocero.com/noticias/raspbian-sistema-operativo-gratuito-para-la-raspberry-pi/> [Último acceso: 07 de Enero de 2017].
- [58] CNT. *Huawei E303*. [En línea]. Disponible en: <https://www.cnt.gob.ec/internet/dispositivo/huawei-e303/> [Último acceso: 10 Ene 2017].
- [59] HUAWEI. *E303. Características*. [En línea]. Disponible en: <http://m.huawei.com/bomobile/consumer/mobile-broadband/mobile-wifi/detail/e303-bo.htm> [Último acceso: 10 de Enero de 2017].
- [60] ASUS. *X455LB*. [En línea]. Disponible en: <https://www.asus.com/Laptops/X455LB/specifications/> [Último acceso: 11 de Enero de 2017].
- [61] Raspberry Pi. *Raspberry Pi Software Guide*. [En línea]. Disponible en: <https://www.raspberrypi.org/learning/software-guide/> [Último acceso: 12 de Enero de 2017].
- [62] Modmypi. *Boost USB Current*. [En línea]. Disponible en: <https://www.modmypi.com/blog/boost-usb-current> [Último acceso: 12 de Enero de 2017].
- [63] Raspberry Pi. *Video options in config.txt*. [En línea]. Disponible en: <https://www.raspberrypi.org/documentation/configuration/config-txt/video.md> [Último acceso: 14 de Enero de 2017].
- [64] Luis V. Márquez. *“Servicios SFTP/SCP”*. [En línea]. Disponible en: <https://luisvmsri.files.wordpress.com/2013/02/sri-tema5-3.pdf> [Último acceso: 14 de Enero de 2017].

[65] Sparkfun. *FLIR Lepton Hookup Guide*. [En línea]. Disponible en: https://learn.sparkfun.com/tutorials/flir-lepton-hookup-guide?_ga=2.264481607.936162645.1507187600-898698457.1498797747 [Último acceso: 14 de Enero de 2017].

[66] Carlini. *Internet móvil con el modem 3G USB Huawei E303 en la Raspberry Pi*. [En línea]. Disponible en: <http://carlini.es/internet-movil-con-el-modem-3g-usb-huawei-e303-en-la-raspberry-pi/> [Último acceso: 17 de Enero de 2017].

[67] Proxicast. *Huawei E3131 modem status LED colors*. [En línea]. Disponible en: <http://www.proxicast.com/AbsoluteFM/afmviewfaq.aspx?faqid=271> [Último acceso: 17 de Enero de 2017].

[68] Mintaka Conciencia. *UMTSkeeper: keep your UMTS/GPRS/GSM connection alive automatically*. [En línea]. Disponible en: <http://mintakaconciencia.net/squares/umtskeeper/> [Último acceso: 18 de Enero de 2017].

[69] Wiki Qt. *Qt Creator/es*. [En línea]. Disponible en: https://wiki.qt.io/Qt_Creator/es [Último acceso: 24 de Enero de 2017].

[70] Xataka Móvil. *Digia compra Qt a Nokia y va a llevarlo a Android, iOS, y Windows 8*. [En línea]. Disponible en: <https://www.xatakamovil.com/desarrollo/digia-compra-qt-a-nokia-y-piensa-en-llevarlo-a-todas-las-plataformas> [Último acceso: 24 de Enero de 2017].

[71] YouTube. *Instalación y configuración de Qt en la Raspberry Pi*. [En línea]. Disponible en: <https://www.youtube.com/watch?v=tGGR9jogIBo> [Último acceso: 25 de Enero de 2017].

[72] Xmcarne. *Instalar Qt en Raspberry Pi 2*. [En línea]. Disponible en: <http://www.xmcarne.com/blog/instalar-qt-en-raspberry-pi-2/> [Último acceso: 26 de Enero de 2017].

[73] Qt Documentation. *Qt Creator Manual*. [En línea]. Disponible en: <http://doc.qt.io/qtcreator/index.html> [Último acceso: 31 de Enero de 2017].

[74] La web del programador. *Curso de QT - Aprenda Qt4 desde hoy mismo*. [En línea]. Disponible en: <https://www.lawebdelprogramador.com/cursos/QT/7375-Aprenda-Qt4-desde-hoy-mismo.html> [Último acceso: 01 de Febrero de 2017].

[75] Qt Documentation. *Signals & Slots*. [En línea]. Disponible en: <http://doc.qt.io/qt-4.8/signalsandslots.html> [Último acceso: 02 de Febrero de 2017].

[76] AWS. *Tipos de instancias de Amazon EC2*. [En línea]. Disponible en: <https://aws.amazon.com/es/ec2/instance-types/> [Último acceso: 06 de Febrero de 2017].

[77] R. S. Pucha Ortiz, *Implementación de una mini-red de sensores inalámbricos para detección temprana de incendios forestales utilizando hardware y software open source*, Quito - Ecuador: EPN, 2016.

[78] Qt Documentation. *All classes*. [En línea]. Disponible en: <http://doc.qt.io/qt-5/classes.html> [Último acceso: 06 de Febrero de 2017].

ANEXOS

ANEXO A: INSTALACIÓN DEL SISTEMA OPERATIVO RASPBIAN.

Luego de formatear la tarjeta micro SD, es necesario descargar la imagen del sistema operativo Raspbian. Para el presente proyecto se utiliza el método de instalación con la aplicación NOOBS (New Out Of Box Software), la cual facilita la implementación de diversas distribuciones de Linux sobre la placa Raspberry Pi.

La aplicación NOOBS se obtiene de la página de la organización Raspberry Pi en el siguiente enlace: <https://www.raspberrypi.org/downloads/noobs/>, y se procede a descargar el paquete denominado "Download ZIP".

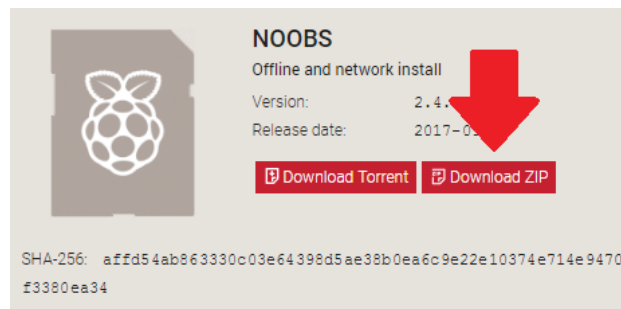


Figura 2.63 Descarga del paquete NOOBS [61].

En cuanto ya se tenga el archivo ZIP descargado con el paquete NOOBS, se procede a descomprimir sus archivos sobre el computador, y posteriormente se transfiere todos estos archivos hacia la tarjeta micro SD que ya se encuentra formateada previamente.

Los archivos copiados en la tarjeta micro SD contienen todos los programas necesarios para proceder con la instalación de la aplicación NOOBS y posteriormente elegir el sistema operativo Raspbian.

Después que ya se tenga los archivos copiados sobre la tarjeta micro SD, se procede a quitarla con seguridad del computador, y se inserta en la ranura de tarjeta micro SD de la Raspberry Pi 2B.

Nombre	Fecha de modificación	Tipo	Tamaño
defaults	24/11/2016 15:09	Carpeta de archivos	
os	29/11/2016 15:15	Carpeta de archivos	
overlays	24/11/2016 14:59	Carpeta de archivos	
bcm2708-rpi-b.dtb	24/11/2016 14:59	Archivo DTB	14 KB
bcm2708-rpi-b-plus.dtb	24/11/2016 14:59	Archivo DTB	14 KB
bcm2709-rpi-2-b.dtb	24/11/2016 14:59	Archivo DTB	15 KB
bcm2710-rpi-3-b.dtb	24/11/2016 14:59	Archivo DTB	16 KB
bootcode	24/11/2016 15:10	BIN File	18 KB
BUILD-DATA	24/11/2016 15:10	Archivo	1 KB
INSTRUCTIONS-README	24/11/2016 15:09	Documento de texto	3 KB
recovery.cmdline	24/11/2016 15:10	Archivo CMDLINE	1 KB
recovery.elf	24/11/2016 15:10	Archivo ELF	619 KB
recovery	24/11/2016 15:10	Archivo de imagen de disco	2.522 KB
recovery.rfs	24/11/2016 15:10	Archivo RFS	23.128 KB
RECOVERY_FILES_DO_NOT_EDIT	24/11/2016 15:10	Archivo	0 KB
recovery7	24/11/2016 15:09	Archivo de imagen de disco	2.592 KB
riscos-boot	24/11/2016 15:09	BIN File	10 KB

Figura 2.64 Archivos que contiene el paquete NOOBS.

Se conecta la placa Raspberry Pi 2B a la fuente de alimentación e inmediatamente se puede observar la programación inicial de la Raspberry Pi sobre la pantalla que se utilice.

Luego de este inicio se llega al punto en donde se debe escoger el sistema operativo para su respectiva instalación, en este caso se elige la opción Raspbian. También en la parte inferior se puede elegir el idioma y el tipo de teclado.

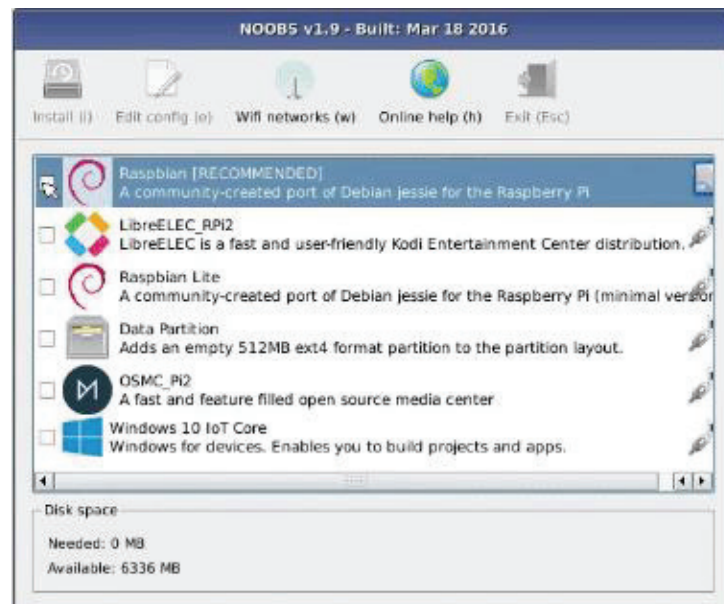


Figura 2.65 Elección del sistema operativo Raspbian en la aplicación NOOBS [61].

Después de esto comienza el programa de instalación del sistema operativo Raspbian tardando varios minutos hasta finalizar dicho proceso y se muestra una pequeña ventana informando que la instalación fue realizada satisfactoriamente.

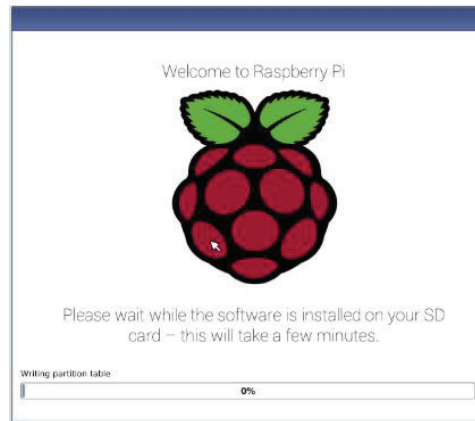


Figura 2.66 Inicio del proceso de Instalación de Raspbian [61].



Figura 2.67 Transcurso del proceso de Instalación de Raspbian [61].



Figura 2.68 Finalización del Proceso de Instalación de Raspbian [61].

Luego de finalizar la instalación se muestra inmediatamente la interfaz de inicio del sistema operativo Raspbian indicando el detalle de su entorno gráfico funcionando con “PIXEL”.

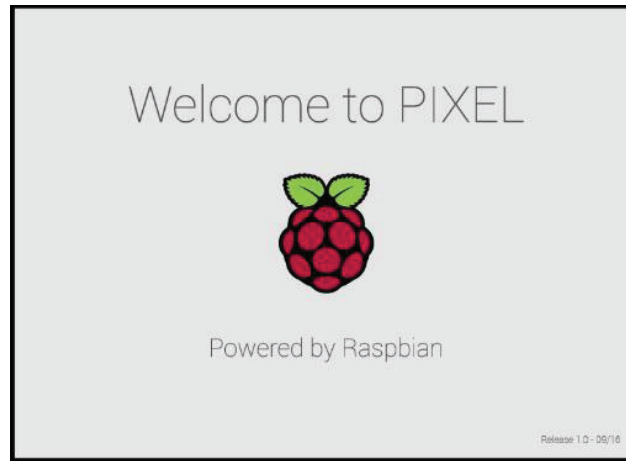


Figura 2.69 Entorno Gráfico de PIXEL sobre el sistema operativo Raspbian [61].

Finalmente se tiene el sistema operativo Raspbian funcionando sobre la placa Raspberry Pi 2B, y se muestra la interfaz de su escritorio.

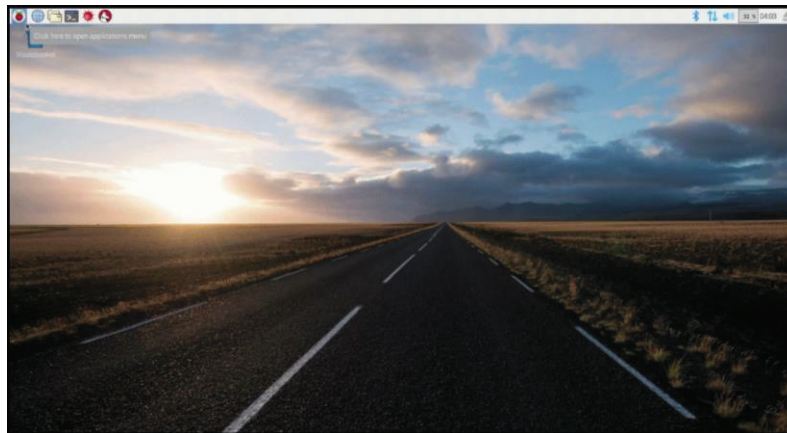


Figura 2.70 Interfaz gráfica del escritorio del sistema operativo Raspbian [61].

El procedimiento inicial que se realiza sobre la Raspberry Pi 2B, es la configuración de su acceso a Internet, para esto se procede a conectar un cable de red al puerto Ethernet.

La configuración por defecto que viene preestablecida sobre la tarjeta Raspberry Pi 2B corresponde a su usuario “pi” y su respectiva contraseña “raspberrypi”. El

usuario y contraseña de la Raspberry Pi 2B se utiliza para poder conectarse remotamente al dispositivo mediante el protocolo de comunicación SSH.

Este tipo de comunicación se utiliza para las respectivas configuraciones de los programas que se utilizan para el prototipo. En este caso se realiza la programación desde un computador, ya que se tiene mayor facilidad de manipulación de los periféricos y mejor visualización de la interfaz gráfica del sistema operativo Raspbian.

ANEXO B: INSTALACIÓN Y CONFIGURACIÓN DEL ENTORNO DE DESARROLLO INTEGRADO QT CREATOR.

Primeramente se procede a actualizar el sistema, por lo tanto se abre la ventana de terminal y se ejecuta las sentencias “*sudo apt-get update*” y “*sudo apt-get upgrade*”. Luego se procede a ejecutar la sentencia “*sudo apt-get install qt4-dev-tools*” para instalar los paquetes que se utilizarán para compilar y desarrollar con el entorno de Qt [72].

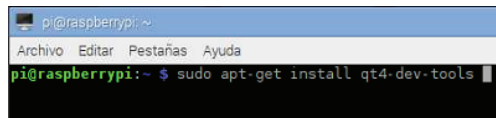


Figura 2.71 Instalación de las herramientas de compilación de Qt.

Para la instalación del IDE Qt Creator sobre el sistema operativo Raspbian se ejecuta la sentencia “*sudo apt-get install qtcreator*” [72].

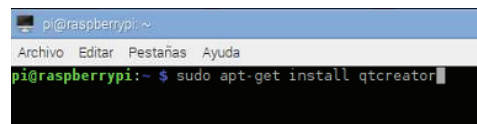


Figura 2.72 Instalación del IDE Qt Creator.

Después de transcurrir varios minutos en la instalación de los paquetes se verifica que el programa ha sido descargado satisfactoriamente sobre la placa Raspberry Pi 2B. Además se puede observar otros programas instalados como “Qt 4 Assistant”, “Qt 4 Designer” y “Qt 4 Linguist”.

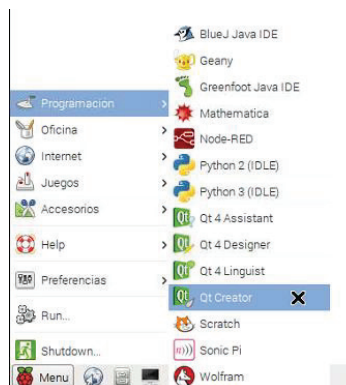


Figura 2.73 Programa Qt Creator instalado en la Raspberry Pi 2B.

Para completar el proceso de instalación se debe configurar el “*IDE Qt Creator*” con cierto tipo de parámetros necesarios para realizar una compilación idónea de la aplicación que se utilizará como interfaz gráfica de usuario sobre la Raspberry Pi 2B. Por tanto se abre el programa Qt Creator y en la barra de herramientas se dirige hacia la pestaña “*Tools*” y se selecciona “*Options*”.

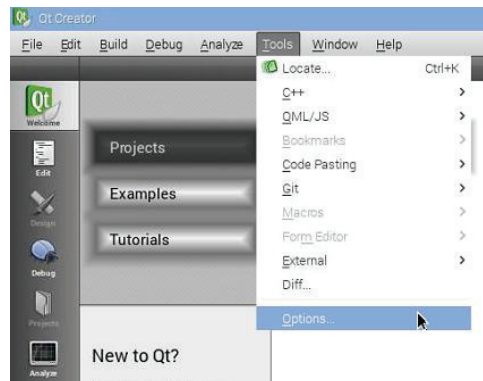


Figura 2.74 Configuración de las opciones del IDE Qt Creator.

En la ventana que se muestra se escoge la opción “*Build & Run*” (Compilar y Ejecutar). La pestaña “*General*” se deja configurada tal como está originalmente.

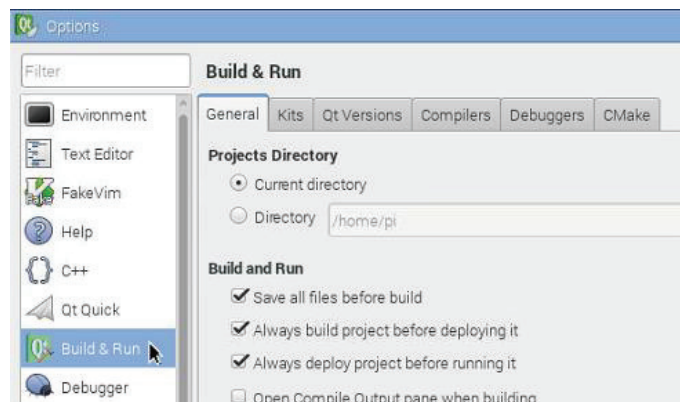


Figura 2.75 Configuración de Build & Run.

En la pestaña “*Qt Versions*” se debe asegurar que se encuentre establecida la versión Qt 4.8.6 (Qt4) y que la localización de qmake sea la correcta “*/usr/lib/arm-linux-gnueabi/qt4/bin/qmake*”. Qmake sirve para construir el programa portable que permitirá la ejecución de las aplicaciones creadas en “*Qt Creator*”.

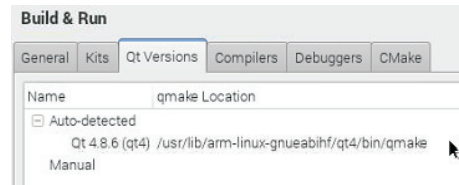


Figura 2.76 Configuración de la Versión de Qt Creator.

En la pestaña “Compilers” se debe establecer el compilador GCC (*GNU Compiler Collection*, Colección de Compiladores GNU), para lo cual con el botón “Browse” se selecciona el archivo “/usr/bin/gcc”, y además se puede editar el nombre del compilador, en este caso el nombre escogido es “GCC-PI”.

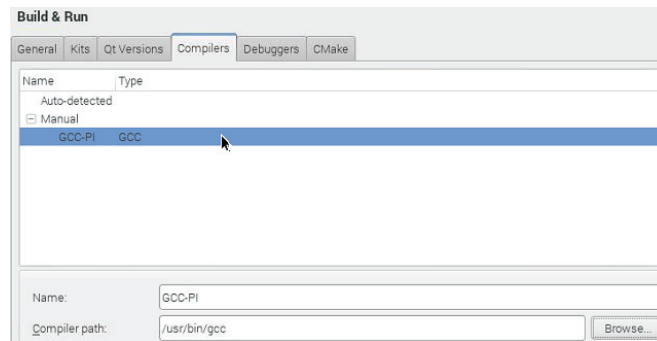


Figura 2.77 Configuración del Compilador de Qt Creator.

De igual manera en la pestaña “Debuggers” con el botón “Browse” se selecciona el archivo “/usr/bin/gdb”, para establecer el depurador, en este caso se tiene el GDB (*GNU Debugger*, Depurador GNU), es el depurador estándar para el compilador GNU. El depurador sirve para eliminar los errores de programación de un proyecto [73].

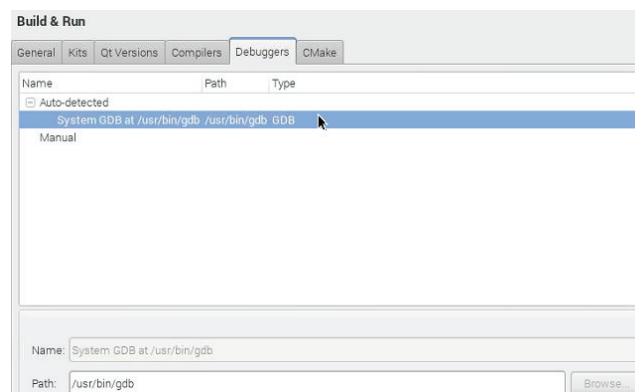


Figura 2.78 Configuración del Depurador de Qt Creator.

En la pestaña “CMake” no se realiza ningún cambio. Y por último luego de haber configurado todos los anteriores parámetros se verifica en la pestaña “Kits” que los valores son los correctos. El kit agrupa la configuración utilizada para crear y ejecutar proyectos. El kit utilizado para el proyecto de cámara térmica es por defecto “Desktop” [73].

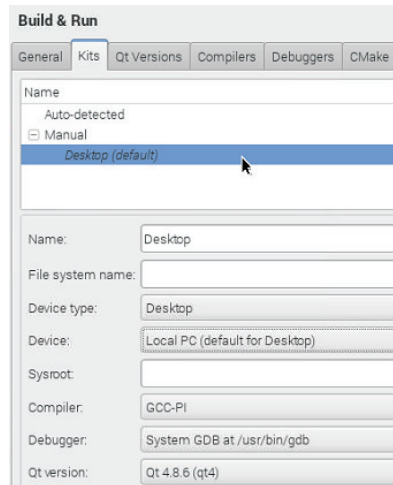


Figura 2.79 Configuración del Kit de Qt Creator.

Y así finalmente luego de establecer todos los parámetros de configuración se tiene listo y funcional el programa Qt Creator para generar cualquier tipo de proyecto.

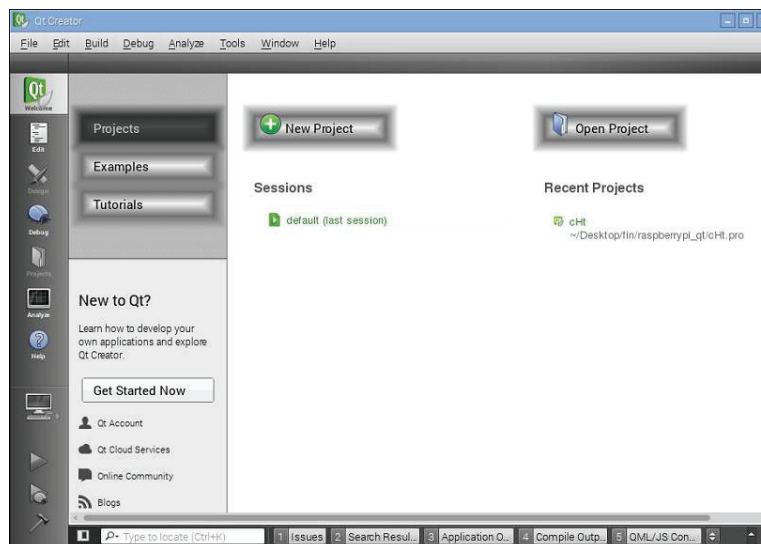


Figura 2.80 Entorno de Desarrollo de Qt Creator.

ANEXO C: FLIR LEPTON® Long Wave Infrared (LWIR) Datasheet.

ORDERING INFORMATION

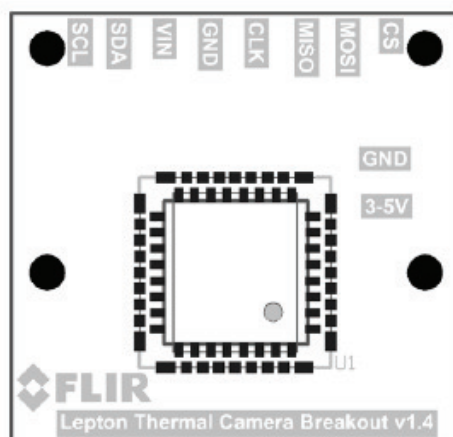
For the most current package and ordering information see the website at www.pureengineering.com.

ABSOLUTE MAXIMUM RATINGS

		UNIT
V_I	Input Voltage	Continuous 5.5 V
T_A	Operating temperature range	Over V_I range 0°C to 55°C

TERMINAL FUNCTIONS

TERMINAL		DESCRIPTION
NAME	NO.	
SCL	8	Camera Control Interface Clock, I2C
SDA	7	Camera Control Interface Data, I2C
VIN	6	3-5V Supply input
GND	5	Common Ground
SPI CLK	4	Video Over SPI Slave Clock
SPI MISO	3	Video Over SPI Slave Data Out
SPI MOSI	2	Video Over SPI Slave Data In
SPI CS	1	Video Over SPI Slave Chip Select, active low



ANEXO D: CÓDIGO FUENTE DE INTERFAZ GRÁFICA DE USUARIO.

- **Fichero de Cabecera “mainwindow.h”**

```

#ifndef MAINWINDOW_H
#define MAINWINDOW_H

#include <QMainWindow>
#include <QImage>
#include <QPushButton>
#include <QDesktopServices>
#include <QProcess>
#include <QStringList>
#include <QTimer>

class QLabel;
class LeptonThread;
class QGridLayout;

class MainWindow : public QMainWindow {
    Q_OBJECT

    enum { ImageWidth = 496, ImageHeight = 372 };

    QLabel *tituloLabel;
    QLabel *imageLabel;
    QLabel *notifiLabel;
    LeptonThread *thread;
    QGridLayout *layout;
    QPushButton *boton2;
    QPushButton *boton3;
    QPushButton *boton4;
    QPushButton *boton5;
    QDesktopServices *abrir;
    QProcess *webter;
    QProcess *apaga;
    QProcess *reinicio;
    QProcess *nube;
    QStringList *arreglo;
    QTimer *lapso;
    QLabel *etique1;
    QLabel *etique2;
    QLabel *etique3;
    QLabel *etique4;
    QLabel *etique5;

    unsigned short rawMin, rawMax;
    QVector<unsigned short> rawData;
    QImage rgbImage;

public:
    explicit MainWindow(QWidget *parent = 0);

public slots:
    void guardarCaptura();
    void actualizarImagen(unsigned short *, int, int);
    void archivo();
    void reenviarCapturas();
    void apagaSistema();
    void reinicioSistema();

};

#endif // MAINWINDOW_H

```

- **Fichero de Fuente Principal “main.cpp”**

```
#include <QApplication>

#include "mainwindow.h"

int main( int argc, char **argv )
{
    QApplication a(argc, argv);

    MainWindow *mainwindow = new MainWindow();
    mainwindow->show();

    return a.exec();
}
```

- **Fichero de Fuente “mainwindow.cpp”**

```
#include "mainwindow.h"

#include <QLabel>
#include <QPushButton>
#include <QGridLayout>
#include <QImage>
#include <QPixmap>
#include <QPainter>
#include <QTime>
#include <QString>
#include <QDesktopServices>
#include <QUrl>
#include <QProcess>

#include "LeptonThread.h"

// Establecimiento del modelo de color de la imagen receptada
MainWindow::MainWindow(QWidget *parent)
    : QMainWindow(parent)
    , rawData(LeptonThread::FrameWords)
    , rgbImage(LeptonThread::FrameWidth, LeptonThread::FrameHeight, QImage::Format_RGB888)
{

    // Establecimiento del diseño de la aplicacion
    QWidget *mainWidget = new QWidget();
    setCentralWidget(mainWidget);

    layout = new QGridLayout();
    mainWidget->setLayout(layout);

    // Establecimiento del titulo de la aplicacion
    tituloLabel= new QLabel("~ CAMARA TERMICA FLIR LEPTON ~");
    layout->addWidget(tituloLabel, 0, 2, Qt::AlignCenter);

    // Posicion de la imagen termica en la aplicacion
    imageLabel = new QLabel();
    layout->addWidget(imageLabel, 1, 2, 2, 1, Qt::AlignCenter);

    // Error de comunicacion del sensor con la interfaz GPIO
    QPixmap filler(ImageWidth, ImageHeight);
    filler.fill(Qt::red);
    imageLabel->setPixmap(filler);

    // Recepcion de los datos de imagen termica
    thread = new LeptonThread();
    connect(thread, SIGNAL(actualizarImagen(unsigned short *,int,int)), this,
    SLOT(actualizarImagen(unsigned short *, int,int)));

    thread->start();

    // Boton para capturar imagenes
```

```

QPushButton *snapshotButton = new QPushButton();
layout->addWidget(snapshotButton, 1, 3, 2, 1, Qt::AlignCenter);
connect(snapshotButton, SIGNAL(clicked()), this, SLOT(guardarCaptura()));
snapshotButton->setIcon(QIcon("/home/pi/Desktop/camara_termica/build-cHt-Desktop-
Debug/iconos/llama.png"));
snapshotButton->setMinimumSize(120,120);

// Boton de reenvio de capturas
boton2 = new QPushButton();
layout->addWidget(boton2, 2, 0, Qt::AlignCenter);
connect(boton2, SIGNAL(clicked()), this, SLOT(reenviarCapturas()));
boton2->setIcon(QIcon("/home/pi/Desktop/camara_termica/build-cHt-Desktop-
Debug/iconos/cloud.png"));
boton2->setMinimumSize(70,70);

// Boton para apagar el sistema
boton3 = new QPushButton();
layout->addWidget(boton3, 1, 0, Qt::AlignCenter);
connect(boton3, SIGNAL(clicked()), this, SLOT(apagaSistema()));
boton3->setIcon(QIcon("/home/pi/Desktop/camara_termica/build-cHt-Desktop-
Debug/iconos/off.png"));
boton3->setMinimumSize(70,70);

// Boton para abrir directorio de imagenes locales
boton4 = new QPushButton();
layout->addWidget(boton4, 2, 1, Qt::AlignCenter);
connect(boton4, SIGNAL(clicked()), this, SLOT(archivo()));
boton4->setIcon(QIcon("/home/pi/Desktop/camara_termica/build-cHt-Desktop-
Debug/iconos/carpeta.png"));
boton4->setMinimumSize(70,70);

// Boton para reiniciar el sistema
boton5 = new QPushButton();
layout->addWidget(boton5, 1, 1, Qt::AlignCenter);
connect(boton5, SIGNAL(clicked()), this, SLOT(reinicioSistema()));
boton5->setIcon(QIcon("/home/pi/Desktop/camara_termica/build-cHt-Desktop-
Debug/iconos/reboot.jpg"));
boton5->setMinimumSize(70,70);

abrir = new QDesktopServices();

webter = new QProcess();

apaga = new QProcess();

reinicio = new QProcess();

nube = new QProcess();

// Establecimiento de etiquetas
etique1 = new QLabel("APAGAR");
layout->addWidget(etique1, 1, 0, Qt::AlignBottom);

etique2 = new QLabel("REINICIAR");
layout->addWidget(etique2, 1, 1, Qt::AlignBottom);

etique3 = new QLabel("REENVIAR");
layout->addWidget(etique3, 2, 0, Qt::AlignBottom);

etique4 = new QLabel("ARCHIVO");
layout->addWidget(etique4, 2, 1, Qt::AlignBottom);

etique5 = new QLabel("ENVIAR");
layout->addWidget(etique5, 2, 3, Qt::AlignCenter);
}

const int colormap[] = {255, 255, 255, 253, 253, 253, 251, 251, 251, 249, 249, 249, 247,
247, 247, 245, 245, 243, 243, 243, 241, 241, 241, 239, 239, 239, 237, 237, 237, 235,
235, 235, 233, 233, 233, 231, 231, 231, 229, 229, 229, 227, 227, 227, 225, 225, 225, 223,
223, 223, 221, 221, 221, 219, 219, 219, 217, 217, 217, 215, 215, 215, 213, 213, 213, 211,
211, 211, 209, 209, 209, 207, 207, 207, 205, 205, 205, 203, 203, 203, 201, 201, 201, 199,
199, 199, 197, 197, 197, 195, 195, 195, 193, 193, 193, 191, 191, 191, 189, 189, 189, 187,
187, 187, 185, 185, 185, 183, 183, 183, 181, 181, 181, 179, 179, 179, 177, 177, 177, 175,
175, 175, 173, 173, 173, 171, 171, 171, 169, 169, 169, 167, 167, 167, 165, 165, 165, 163,
163, 163, 161, 161, 161, 159, 159, 159, 157, 157, 157, 155, 155, 155, 153, 153, 153, 151,

```

```

151, 151, 149, 149, 149, 147, 147, 147, 145, 145, 145, 143, 143, 143, 141, 141, 141, 139,
139, 139, 137, 137, 137, 135, 135, 135, 133, 133, 133, 131, 131, 131, 129, 129, 129, 126,
126, 126, 124, 124, 124, 122, 122, 122, 120, 120, 120, 118, 118, 118, 116, 116, 116, 114,
114, 114, 112, 112, 112, 110, 110, 110, 108, 108, 108, 106, 106, 106, 104, 104, 104, 102,
102, 102, 100, 100, 100, 98, 98, 98, 96, 96, 96, 94, 94, 94, 92, 92, 92, 90, 90, 90, 88,
88, 88, 86, 86, 86, 84, 84, 84, 82, 82, 82, 80, 80, 80, 78, 78, 78, 76, 76, 76, 74, 74, 74,
72, 72, 72, 70, 70, 70, 68, 68, 68, 66, 66, 66, 64, 64, 64, 62, 62, 62, 60, 60, 60, 58, 58,
58, 56, 56, 56, 54, 54, 54, 52, 52, 52, 50, 50, 50, 48, 48, 48, 46, 46, 46, 44, 44, 44, 42,
42, 42, 40, 40, 40, 38, 38, 38, 36, 36, 36, 34, 34, 34, 32, 32, 32, 30, 30, 30, 28, 28, 28,
26, 26, 26, 24, 24, 24, 22, 22, 22, 20, 20, 20, 18, 18, 18, 16, 16, 16, 14, 14, 14, 12, 12,
12, 10, 10, 10, 8, 8, 8, 6, 6, 6, 4, 4, 4, 2, 2, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
6, 0, 31, 8, 0, 38, 10, 0, 45, 12, 0, 53, 14, 0, 60, 17, 0, 67, 19, 0, 74, 21, 0, 82, 23,
0, 89, 25, 0, 96, 27, 0, 103, 29, 0, 111, 31, 0, 118, 36, 0, 120, 41, 0, 121, 46, 0, 122,
51, 0, 123, 56, 0, 124, 61, 0, 125, 66, 0, 126, 71, 0, 127, 76, 1, 128, 81, 1, 129, 86, 1,
130, 91, 1, 131, 96, 1, 132, 101, 1, 133, 106, 1, 134, 111, 1, 135, 116, 1, 136, 121, 1,
136, 125, 2, 137, 130, 2, 137, 135, 3, 137, 139, 3, 138, 144, 3, 138, 149, 4, 138, 153, 4,
139, 158, 5, 139, 163, 5, 139, 167, 5, 140, 172, 6, 140, 177, 6, 140, 181, 7, 141, 186, 7,
141, 189, 10, 137, 191, 13, 132, 194, 16, 127, 196, 19, 121, 198, 22, 116, 200, 25, 111,
203, 28, 106, 205, 31, 101, 207, 34, 95, 209, 37, 90, 212, 40, 85, 214, 43, 80, 216, 46,
75, 218, 49, 69, 221, 52, 64, 223, 55, 59, 224, 57, 49, 225, 60, 47, 226, 64, 44, 227, 67,
42, 228, 71, 39, 229, 74, 37, 230, 78, 34, 231, 81, 32, 231, 85, 29, 232, 88, 27, 233, 92,
24, 234, 95, 22, 235, 99, 19, 236, 102, 17, 237, 106, 14, 238, 109, 12, 239, 112, 12, 240,
116, 12, 240, 119, 12, 241, 123, 12, 241, 127, 12, 242, 130, 12, 242, 134, 12, 243, 138,
12, 243, 141, 13, 244, 145, 13, 244, 149, 13, 245, 152, 13, 245, 156, 13, 246, 160, 13,
246, 163, 13, 247, 167, 13, 247, 171, 13, 248, 175, 14, 248, 178, 15, 249, 182, 16, 249,
185, 18, 250, 189, 19, 250, 192, 20, 251, 196, 21, 251, 199, 22, 252, 203, 23, 252, 206,
24, 253, 210, 25, 253, 213, 27, 254, 217, 28, 254, 220, 29, 255, 224, 30, 255, 227, 39,
255, 229, 53, 255, 231, 67, 255, 233, 81, 255, 234, 95, 255, 236, 109, 255, 238, 123, 255,
240, 137, 255, 242, 151, 255, 244, 165, 255, 246, 179, 255, 248, 193, 255, 249, 207, 255,
251, 221, 255, 253, 235, 255, 255, 24};

```

```

// Funcion de actualizacion de imagen termica
void MainWindow::actualizarImagen(unsigned short *data, int minValue, int maxValue){

    // Registro de raw data y sus valores min/max
    memcpy(&rawData[0], data, 2*LeptonThread::FrameWords);
    rawMin = minValue; rawMax = maxValue;

    // Transformacion de "rawData" a valores rgb en "rgbImage", a traves de "colormap"
    int diff = maxValue - minValue + 1;
    for (int y = 0; y < LeptonThread::FrameHeight; ++y) {
        for (int x = 0; x < LeptonThread::FrameWidth; ++x) {
            int baseValue = rawData[LeptonThread::FrameWidth*y + x]; // se toma el valor de
            entrada en [0, 65536)
            int scaledValue = 256*(baseValue - minValue)/diff; // se asigna el valor al
            intervalo [0, 256), y se establece el pixel en su valor de color anterior
            rgbImage.setPixel(x, y, qRgb(colormap[3*scaledValue],
            colormap[3*scaledValue+1], colormap[3*scaledValue+2]));
        }
    }

    // Actualizacion de la imagen en pantalla
    QPixmap pixmap = QPixmap::fromImage(rgbImage).scaled(ImageWidth, ImageHeight,
    Qt::KeepAspectRatio);
    QPainter painter(&pixmap);
    imageLabel->setPixmap(pixmap);
}

// Funcion para capturar, guardar y enviar imagen termica
void MainWindow::guardarCaptura() {

    // Se crea variable de fecha y hora, para asignar al nombre de archivo
    QString fechatiempo = QDateTime::currentDateTime().toString("_'yyyyMMdd_HHmss");

    // Se guarda imagen localmente en formato jpg con calidad 100%
    rgbImage.save(QString("/home/pi/Desktop/camara_termica/build-cHt-Desktop-
    Debug/repositorio/termica%1.jpg").arg(fechatiempo), "JPG", 100);

    // Se envia imagen capturada al servidor en la nube
    arreglo = new QStringList();
    arreglo->append("scp");
    arreglo->append("-i");
    arreglo->append("/home/pi/servidorawstermica2017.pem");
    arreglo->append(QString("/home/pi/Desktop/camara_termica/build-cHt-Desktop-
    Debug/repositorio/termica%1.jpg").arg(fechatiempo));
}

```

```

arreglo->append("ubuntu@18.221.244.121:/var/www/html/repositorio/");
nube->start("sudo",*arreglo);

// Se muestra aviso, informando que la imagen fue capturada
notifiLabel= new QLabel("Enviando\nCaptura...");
layout->addWidget(notifiLabel, 1, 3, Qt::AlignCenter);

lapso = new QTimer();
lapso->singleShot(7000, notifiLabel, SLOT(hide()));
}

// Funcion para abrir archivo local
void MainWindow::archivo() {

    // Se abre el directorio local que contiene las imagenes termicas
    abrir->openUrl(QUrl("/home/pi/Desktop/camara_termica/build-cHt-Desktop-
Debug/repositorio/", QUrl::TolerantMode));
}

// Funcion para reenviar imagenes capturadas
void MainWindow::reenviarCapturas() {

    // Opcion emergente para reenviar imagenes capturadas
    webter->start("/usr/local/bin/enviater");
}

// Funcion para apagar el sistema
void MainWindow::apagaSistema() {

    // Se ejecuta script para apagar el sistema
    apaga->start("/usr/local/bin/apagater");
}

// Funcion para reiniciar el sistema
void MainWindow::reinicioSistema() {

    // Se ejecuta script para reiniciar el sistema
    reinicio->start("/usr/local/bin/reiniter");
}
}

```