

# **PROYECTO FIN DE CARRERA**

Presentado a

**LA UNIVERSIDAD DE LOS ANDES  
FACULTAD DE INGENIERÍA  
DEPARTAMENTO DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA**

Para obtener el título de

**INGENIERO ELECTRÓNICO**

por

***Brayan Esteban Ariza Ariza***

---

***Cámara térmica para sistemas microfluídicos.***

---

Sustentado el día 11 de diciembre de 2019 frente al jurado:

## Composición del jurado

- *Asesor:* Dr. Johann F. Osma, Profesor Asociado, Universidad de Los Andes
- *Jurados :* Luis Felipe Giraldo, Profesor Asociado, Universidad de Los Andes

## Contenido

1	INTRODUCCIÓN .....	3
2	OBJETIVOS .....	3
2.1	Objetivo General.....	3
2.2	Objetivos Específicos.....	3
2.3	Alcance y productos finales .....	3
3	DESCRIPCIÓN DE LA PROBLEMÁTICA Y JUSTIFICACIÓN DEL TRABAJO .....	4
4	MARCO TEÓRICO, CONCEPTUAL E HISTÓRICO .....	4
4.1	Marco Teórico y conceptual.....	4
4.2	Marco Histórico .....	6
5	DEFINICION Y ESPECIFICACION DEL TRABAJO.....	6
5.1	Definición.....	6
5.2	Especificaciones .....	6
6	METODOLOGÍA DEL TRABAJO .....	7
6.1	Plan de trabajo .....	7
6.2	Búsqueda de información.....	8
6.3	Alternativas de desarrollo.....	8
7	MATERIALES .....	9
8	TRABAJO REALIZADO .....	8
8.1	Descripción del Resultado Final.....	14
8.2	Trabajo computacional.....	17
9	VALIDACIÓN DEL TRABAJO.....	17
9.1	Metodología de prueba.....	17
9.2	Validación de los resultados del trabajo .....	18
9.3	Evaluación del plan de trabajo .....	19
10	DISCUSIÓN.....	20
11	CONCLUSIONES .....	20
12	AGRADECIMIENTOS.....	21
13	REFERENCIAS.....	22
14	APENDICES .....	23

## 1 INTRODUCCIÓN

Los sistemas microfluídicos están llegando para revolucionar las técnicas de investigación en campos como la química, la biología, la ingeniería ambiental, entre otros, ya que permiten un control de sustancias en escalas muy pequeñas. Ya que estos dispositivos se diseñan integrando la mayoría de las funciones necesarias para un proceso, es posible diseñar diferentes componentes y procesos de ensayo antes o después de presentar cualquier reacción [1].

Con el fin de observar los resultados en los sistemas microfluídicos en cada una de sus etapas, se requiere de cierta variedad de sensores, como cámaras (usualmente microscopios), sensores de flujo, espectrofotómetro, etc., que varían de acuerdo con la aplicación o al estudio que se esté realizando. Una de las variables más importantes en muchas reacciones químicas es la temperatura de los reactivos y del producto, pues con este factor es posible hacer cálculos de absorción o expulsión de calor con el fin de calcular cosas como la cantidad de reactivos que reaccionan en un experimento. Actualmente no existe un sistema comercial que permita realizar estas mediciones en sistemas microfluídicos, por lo que este proyecto se basa en la implementación de una cámara térmica para analizar estos sistemas en el espectro infrarrojo largo, en otras palabras, un sistema que permita conocer la temperatura interna a lo largo de una parte específica de un sistema mediante un mapa de calor.

## 2 OBJETIVOS

### ***2.1 Objetivo General***

Diseñar e implementar un sistema de medición térmica para ambientes, procedimientos o sistemas microfluídicos, con la capacidad de obtener una lectura de temperatura de una superficie en dos dimensiones.

### ***2.2 Objetivos Específicos***

- Estudiar y caracterizar sensores que permitan la medición de la temperatura sin contacto físico.
- Diseñar un sistema de medición, procesamiento y visualización de temperatura en dos dimensiones.
- Implementar el sistema de medición anterior en un sistema microfluídico a escala milimétrica.

### ***2.3 Alcance y productos finales***

El compromiso adquirido para el presente proyecto de grado es entregar un prototipo funcional que permita medir temperatura en un mapa de calor y que este sea observable en tiempo real, ya sea mediante una interfaz gráfica o por visualización de datos. Con el fin de cumplir el compromiso se realizó la caracterización de una cámara térmica y el desarrollo de una interfaz que permita un manejo y adquisición de datos de manera cómoda, de modo que permite ser ubicado en cualquier espacio.

### 3 DESCRIPCIÓN DE LA PROBLEMÁTICA Y JUSTIFICACIÓN DEL TRABAJO

Los sistemas microfluídicos (Lab on a Chip) son sistemas que han ido tomando fuerza en los últimos años debido a sus ventajas frente a laboratorios convencionales y su capacidad de simular entornos biológicos como venas o vasos capilares. Ya que estos sistemas son muy pequeños, se hace difícil sensar ciertas características como la presión, espectrofotometría, **temperatura**, entre otros.

La medición de la temperatura suele ser una necesidad esencial en algunos procesos químicos y físicos, pues con ella y otros factores es posible saber si una reacción ha tenido lugar dentro del sistema. Incluso, con un mapa del calor emitido en una reacción es posible calcular la cantidad de reactivos que han reaccionado en un proceso.

Ya que los sensores de temperatura tradicionales no son una opción convencional dentro de estos sistemas, la implementación de un sensor que permita realizar la medición de la temperatura externamente es una necesidad para quienes suelen trabajar estos sistemas.

### 4 MARCO TEÓRICO, CONCEPTUAL E HISTÓRICO

#### 4.1 Marco Teórico y Conceptual

##### 4.1.1 Sistemas microfluídicos

Los sistemas microfluídicos, también llamados Lab on a Chip, son sistemas con la capacidad de simular métodos y procesos de laboratorio, sistemas fluidicos biológicos como venas y vasos, entre otros. La principal ventaja de estos chips es el tamaño en que son fabricados, pues los canales pueden llegar a medir escalas nanométricas, haciendo posible analizar incluso células, partículas o reacciones con pequeñas cantidades de reactivos [2].



**Figura 1.** Sistema microfluídico Toroide, reducción de perfil de velocidad.

##### 4.1.2 Radiación infrarroja

La radiación infrarroja es la porción del espectro electromagnético que va desde el color rojo hasta las microondas. El rango infrarrojo se divide en tres, el cercano con longitudes de onda entre  $0.75\mu\text{m}$  y  $2.5\mu\text{m}$ , el medio con longitudes de onda entre  $2.5\mu\text{m}$  hasta alrededor de  $50\mu\text{m}$  y el lejano de  $50\mu\text{m}$  a  $1000\mu\text{m}$ . El infrarrojo medio es

el que usualmente se designa como térmico, pues es en este rango en donde están las temperaturas más habituales en la tierra (a partir de unos 100 K) [11].

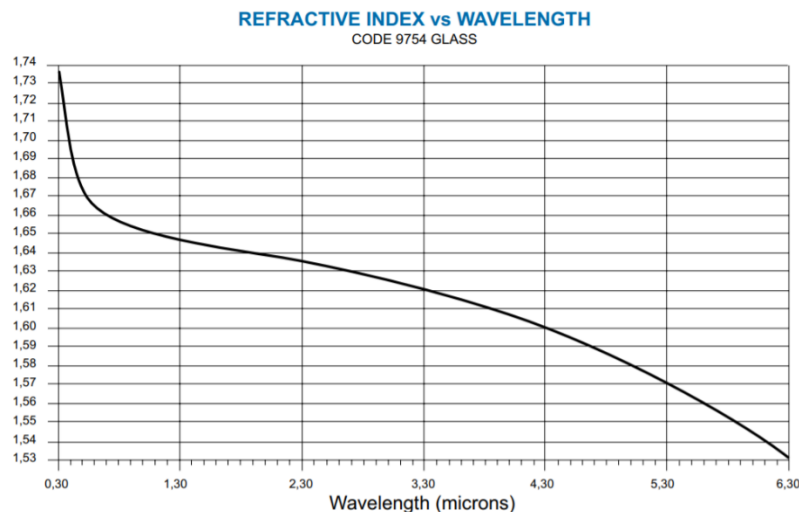
La temperatura de un cuerpo se calcula teniendo en cuenta dos factores, la energía emitida por el objeto a medir y la temperatura propia del sensor infrarrojo con la que se tiene un punto de referencia para medir temperaturas. La radiación emitida por un cuerpo comprende de dos propiedades principales, la frecuencia y la potencia. Para tener una lectura de temperatura se captura la radiación infrarroja y la frecuencia donde la potencia sea mayor corresponde a la temperatura del cuerpo en la una relación inversamente proporcional a esta frecuencia [12].

### 4.1.3 Refracción de la Luz en vidrio y acrílico.

La refracción de la luz indica el cambio en la velocidad de transmisión de una onda a través de un material. Esto afecta directamente el ángulo de transmisión y reflexión del material, dando idea de la deformación que va a sufrir la luz al atravesarlo.

El índice de refracción varía de acuerdo con la longitud de onda de la radiación, por lo que es necesario conocer el índice exacto para la frecuencia de la luz infrarroja. En el caso del vidrio, el índice de refracción es aproximadamente de  $n=1.63$  para longitud de onda de  $\lambda=2.6\mu\text{m}$ . Como se puede observar en la Figura 2, el índice de refracción disminuye a medida que incrementa la longitud de onda, esto quiere decir que, para temperaturas más bajas, donde la longitud de onda es mayor, el índice de refracción hará menos interferencia [8].

Para el acrílico, el índice de refracción es de  $n = 1.491$  para una longitud de onda de  $\lambda=598\text{nm}$ , que, aunque sea la correspondiente al espectro visible, en el acrílico este índice no varía mucho en cuanto a la frecuencia de la onda [7]. Por otro lado, el índice de refracción en los cristales tiende a decrecer a medida que aumenta la longitud de onda, por lo que el comportamiento es prometedor para el desarrollo del proyecto [7].



**Figura 2.** Índice de refracción en función de la longitud de onda. Corning France, dataSheet code 9754 glass.

#### **4.2 Marco Histórico**

- (a) En la revista científica de química Analyst, se publicó un artículo sobre un sistema de censado térmico para reacciones bioquímicas en escalas micrométricas usando sensores térmicos de Cuarzo en corte-Y. Este consiste en un arreglo de sensores térmicos unido externamente a una cámara externa de reacciones, donde el calor llega a los sensores por radiación y conducción [13].
- (b) Lab on a chip publicó un artículo en el que se presenta un análisis térmico de un líquido con nano partículas de  $\text{Al}_2\text{O}_3$  en una plataforma para micro fluidos usando una cámara infrarroja HD. El estudio se basa en el proceso de la reacción de la sustancia, mas no en el sistema de medición como tal, lo que da paso a la implementación de un sistema como este [2].

### **5 DEFINICION Y ESPECIFICACION DEL TRABAJO**

#### **5.1 Definición**

El problema que plantea resolver el proyecto es el de medición y lectura de temperatura mediante mapas de calor, donde la resolución sea por lo menos de  $9\mu\text{m}^2$  ( $300\mu\text{m} \times 300\mu\text{m}$ ), para sistemas microfluídicos fabricados en vidrio o en acrílico. La lectura de estos mapas de calor tiene que ser lo suficientemente rápida como para dar la impresión de que es en tiempo real.

La solución para implementar será una herramienta de laboratorio, que sea ajustable a los sistemas con el fin de que la manufactura actual de estos no se vea afectada en proporciones relevantes. Con lo anterior se espera realizar una inversión no tan alta (alrededor de \$120 us) en un solo sistema, manteniendo los costos de producción de microsistemas en bajo costo (alrededor de \$2 us la unidad).

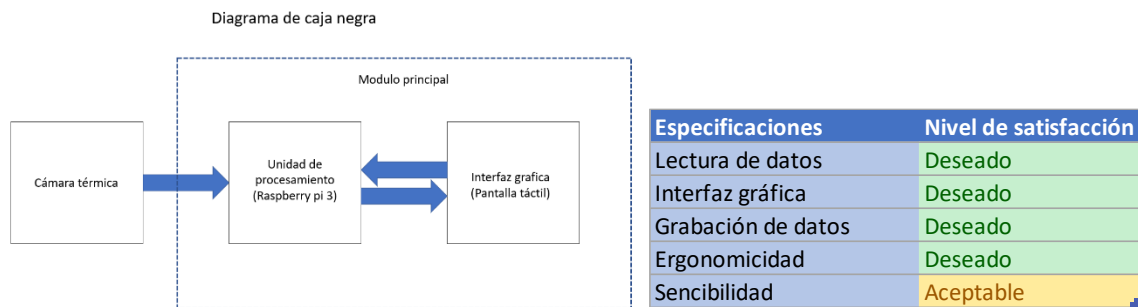
#### **5.2 Especificaciones**

El proyecto es un dispositivo que consta de dos partes físicas, la cámara térmica que mide los datos y el módulo principal que se encarga de procesar la información y de mostrarla en una pantalla led. El módulo de la cámara tiene como característica la adaptabilidad a los sistemas microfluídicos por medio de un sistema de agarre y fijamiento. El módulo de procesamiento se encarga de comunicarse con la cámara, obtener sus datos, mostrarlos en una interfaz gráfica, realizar procesamiento de imagen con el fin de mejorarla, grabar en video y grabar en archivo csv los datos.

Las restricciones que presenta el proyecto son la portabilidad, debe estar conectado a la toma de 120V, la cámara y el módulo principal están unidos por cables (se pueden extender hasta 50 cm dependiendo del cable), los cambios de temperatura registrados

dependen del espesor de la capa externa del sistema microfluídico (ya sea acrílico o vidrio) entre la materia a medir y la cámara.

Ya que la cámara va superpuesta en los sistemas microfluídicos limita la posibilidad de usar otros sistemas de medición al mismo tiempo.



**Figura 3.** Diagrama de caja negra del proyecto.

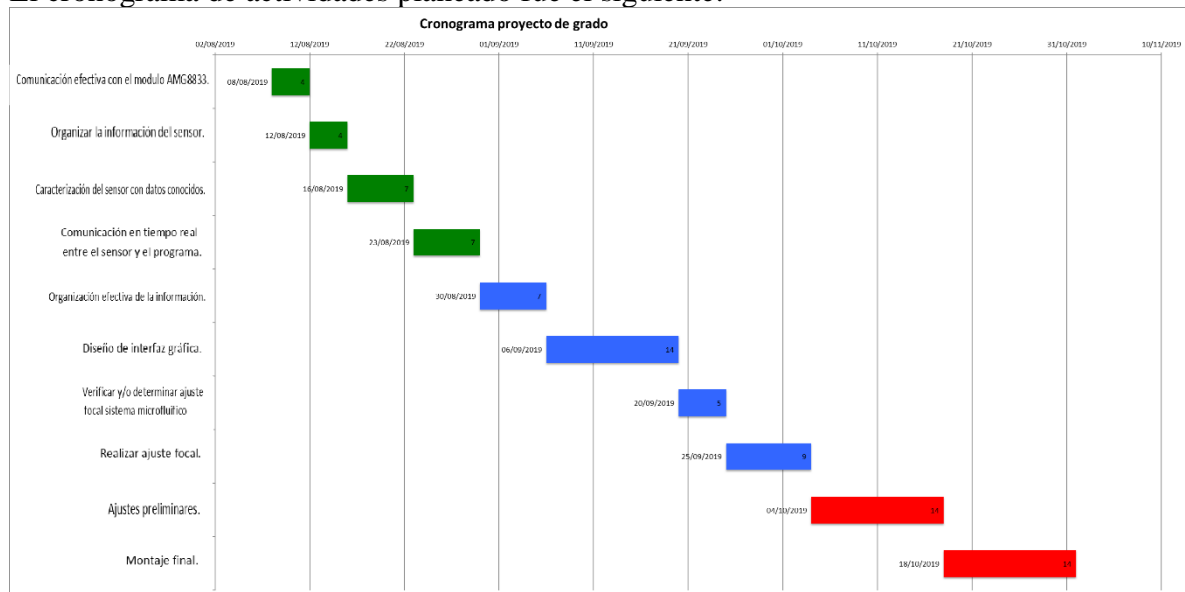
**Tabla 1.** Parámetros generales del proyecto.

## 6 METODOLOGÍA DEL TRABAJO

El plan de trabajo desarrollado se divide en secciones: Búsqueda de información, desarrollo básico de software para trabajar la cámara, diseño y realización de experimentos para la caracterización de la cámara, experimentos en sistemas microfluídicos y desarrollo la interfaz final con las funciones extras implementadas.

### 6.1 Plan de trabajo

El cronograma de actividades planeado fue el siguiente:



**Figura 4.** Cronograma de actividades planteado

En la Figura 4 se observa el cronograma de actividades en el cual se basó el proyecto.

Las reuniones con el asesor fueron cada semana a modo de seminario con el grupo de tesis (todos los miércoles) desde el miércoles 14 de agosto hasta el 27 de noviembre de 2019. Sin embargo, los informes de avances fueron presentados cada 2 semanas con el fin de tener avances concretos en cada informe.

## **6.2 Búsqueda de información**

La búsqueda de información en este proyecto estuvo dirigida a tres ítems principales: fundamentos de funcionamiento de la cámara, diseño y manufactura de sistemas microfluídicos y herramientas en desarrollo de software para el procesamiento de la información.

En cuanto al funcionamiento de la cámara, se encontró el principio de la radiación infrarroja en función de la temperatura en los cuerpos, los materiales sensibles a esta radiación, la tecnología que usan los sensores de temperatura infrarrojos y los protocolos de comunicación que usa la cámara a implementar (AMG8833). En términos de diseño y manufactura de sistemas microfluídicos se encontró bastante información acerca de sistemas fabricados en vidrio. Sin embargo, en la Universidad de los Andes, en el grupo de investigación CMUA, se trabaja en el desarrollo de sistemas microfluídicos de bajo costo, fabricados en acrílico, por lo que la principal fuente de información acerca de especificaciones y necesidades vino por asesoría de las personas que constantemente trabajan en este campo. Por último, para el desarrollo del software las principales fuentes de información fueron tutoriales que, en general, estaban orientados a herramientas o librerías como PyGame, scipy y Adafruit\_AMG88xx.

## **6.3 Alternativas de desarrollo**

En cuanto al desarrollo de dispositivos surgieron ideas alternativas al diseño planteado anteriormente. La principal alternativa considerada fue la de cambiar el material de fabricación de los microsistemas que quisieran ser medidos. Los microsistemas en acrílico presentaron un problema en cuanto a medición de la temperatura de la materia interna (Que será descrito en detalle más adelante), por lo que se propuso cambiar el acrílico por fluoruro de calcio o algún cristal que funcione como filtro IR. Esto facilitaría el paso de la radiación infrarroja, sin embargo, aumentaría los costos y tiempos de manufactura de los dispositivos, por lo que no se tuvo en cuenta al final.

Por otro lado, se tuvo opción de desarrollar el proyecto controlando la interfaz desde un computador externo mediante conexión WIFI. Esta solución haría que el sistema sea más económico de implementar y quizá más fácil de manipular, sin embargo, esto crea una dependencia a la disponibilidad de un computador y una conexión WIFI constante, generando inconvenientes a la disponibilidad del proyecto. Además, implicaría que



durante la realización de un experimento sea necesaria la manipulación de un equipo de uso diario, afectando la integridad tanto del equipo, como de las personas alrededor.

## 7 MATERIALES

- I. Raspberry Pi 3 modelo B.  
Es un sistema embebido desarrollado por la compañía Raspberry Pi compatible con diferentes sistemas operativos que ofrece diferentes tipos de conexiones a periféricos [2].
- II. Cámara térmica AMG8833  
Es una cámara que mide la intensidad de ondas entre 5 y 12  $\mu\text{m}$ , es decir, luz infrarroja. El lente de la cámara, referencia AMG88, es fabricado por Panasonic [3], pero el embebido completo es fabricado por Adafruit Industries [4].
- III. Pantalla táctil con driver ili9341, tiene resolución de 240x320 y es fabricada por ILITEK.
- IV. Termómetro infrarrojo laser AstroAi, es un sensor de temperatura a distancia por medio de radiación infrarroja. Es fabricado distribuido por la empresa AstroAi [5].
- V. Plancha de calentamiento y agitador magnético ms-h280, fabricado por DLAB pro.
- VI. Láminas de polimetilmetacrilato (acrílico), fabricadas por Central de acrílicos LTDA.
- VII. Láminas de MDF, fabricadas por Duratex Colombia.
- VIII. Bomba perfusora Medcaptain MP-30, es una bomba de jeringa que permite realizar bombeo de fluidos con velocidad contante, distribuida por MEDCAPTAIN MEDICAL TECHNOLOGY CO., LTD.

## 8 TRABAJO REALIZADO

Para empezar, se realizó un estudio a cerca de las posibles tecnologías con las que podría ser realizado el proyecto. Para esto se evaluaron tres tecnologías: El efecto seebeck (termocupla), cambio de resistencia en materiales y la tecnología infrarroja. El efecto seebeck tiene varias ventajas como el rango de medición y la linealidad (depende del material), sin embargo, implementar esta tecnología dentro de un sistema microfluídico es un reto de manufactura de microsistema incrementen considerablemente. Algo similar ocurre con el cambio de resistencia de un material, como el cobre, en función de la temperatura, pues, aunque la teoría es sencilla, se requieren complejos procedimientos para integrar metales como el cobre en estos sistemas. Por último, la tecnología infrarroja presenta un costo por sensor elevado, sin embargo, con esta se presenta la oportunidad de realizar mediciones sin realizar grandes modificaciones a los sistemas microfluídicos que han sido y serán desarrollados.

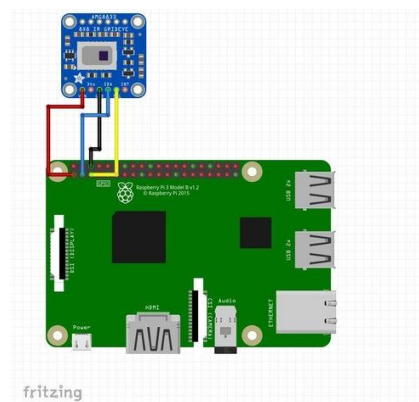
### Cámara térmica

Para implementar la lectura de la temperatura se escogió la cámara térmica AMG8833, desarrollada por Panasonic. Esta cámara cuenta con 64 sensores infrarrojos ubicados en

una matriz de 8x8 píxeles. El tamaño de la abertura del lente es de 2.6mm en cada lado y su abertura es de 60°. La lectura de los datos se realiza mediante protocolo de comunicación I<sup>2</sup>C.

## Conexión entre la cámara y la raspberry pi:

Para el módulo de procesamiento se implementó una raspberry Pi 3b, con el fin de realizar el procesamiento en el menor tiempo posible. La tarjeta Raspberry Pi 3b cuenta con un par de pines especiales para la comunicación en I<sup>2</sup>C, protocolo con el que se comunica la cámara. Ya que estos pines son fijos, es necesario que los pines SDA y SCL se conecten a los mismos pines en la tarjeta, tal como se muestra en la Figura 5.

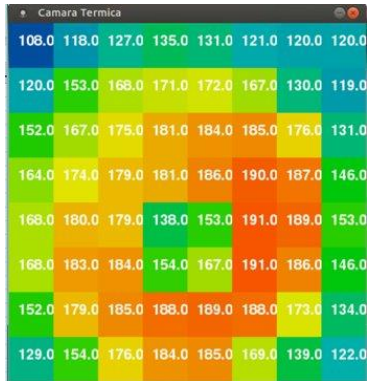


**Figura 5.** Conexión entre la cámara y la tarjeta Raspberry.

Una vez realizada la conexión física, se procede a arrancar el sistema operativo de la tarjeta, que para el caso se instaló el OS de software libre Ubuntu mate 16.4. Aunque los protocolos de comunicación están en la tarjeta, es necesario activar los puertos para que sea posible leerlos. Para realizar la comunicación con la tarjeta se instaló la librería de Adafruit\_AMG88xx, que es desarrollada para realizar una comunicación efectiva con la cámara.

## Primer desarrollo del software

Una de las primeras tareas a realizar es la caracterización de la cámara, pero para que el experimento fuese más cómodo de llevar, se llevó a cabo el desarrollo de una interfaz que permitiera la visualización de los datos en una interfaz gráfica. Con ayuda de la librería grafica Pygame, se realizó la primera interfaz que se muestra en la Figura 6, en donde cada rectángulo hace parte del mapa de calor leído. Sobre cada pixel se imprime la información leída por la cámara.



**Figura 6.** Primera interfaz desarrollada.

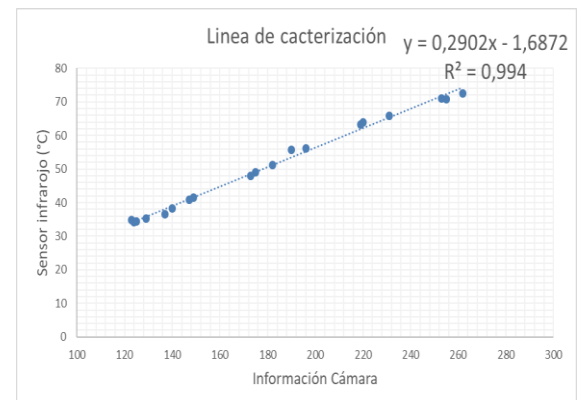
### Primer experimento de caracterización

Con el fin de realizar la caracterización de la cámara y así saber la relación exacta entre la información leída y la temperatura medida, se utilizó un termómetro infrarrojo como referencia de temperatura. El experimento consistió en fijar la cámara y el termómetro a una base de madera (MDF) cortada a la medida para sostener la raspberry Pi junto a los elementos ya mencionados. Con la ayuda de una plancha de calentamiento ms-h280 se calentó agua a diferentes temperaturas y se midió tanto con la cámara como con el termómetro la temperatura.

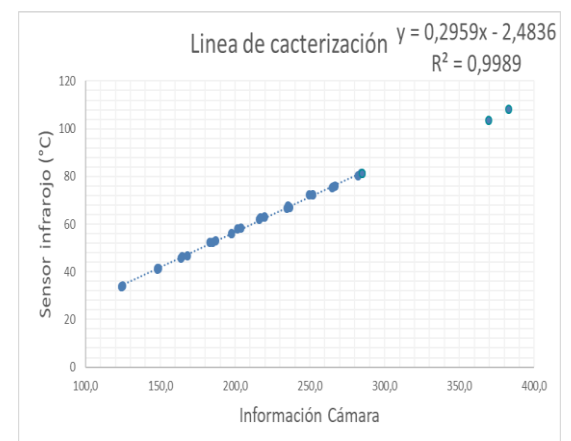
Las mediciones se realizaron apuntando el termómetro al centro del vaso de precipitado con agua. Donde se esperaba que la temperatura fuese más uniforme. La línea de caracterización obtenida se puede observar en la Figura 7 y como en ella se puede ver, los datos están muy dispersos de la línea de tendencia, por lo que puede que lo sea una medida confiable.

### Segundo experimento de caracterización

Con el objetivo de obtener una mejor caracterización, se diseñó un segundo experimento, en el que, en lugar de calentar agua, se midió directamente la plancha de calentamiento, pues a diferencia del agua, la superficie de la plancha al ser de cerámica tiene la cualidad de ser un excelente conductor térmico, por lo que la distribución de calor sería mejor. Por otro lado, la superficie de esta plancha no es reflectante ni translúcida, por lo que las medidas de luz infrarroja presentarían menos distorsiones. En el momento de fijar la base con la cámara y el termómetro, se notó que había un área donde la información de temperatura era casi uniforme y se decidió tomar los píxeles de esta área para realizar un promedio entre ellos. Al momento de obtener los datos se notó que la plancha tenía unas pequeñas variaciones de temperatura aun cuando estaba en el punto estable, debido a esto se decidió tomar varios datos por cada temperatura configurada, pero no realizar un promedio entre estos, sino que serían ingresados todos para realizar la línea de tendencia a partir de estos. Cabe resaltar que las variaciones que presentaba la plancha eran en su mayoría decrecientes, por lo que realizar un promedio podría generar un error con tendencia negativa.

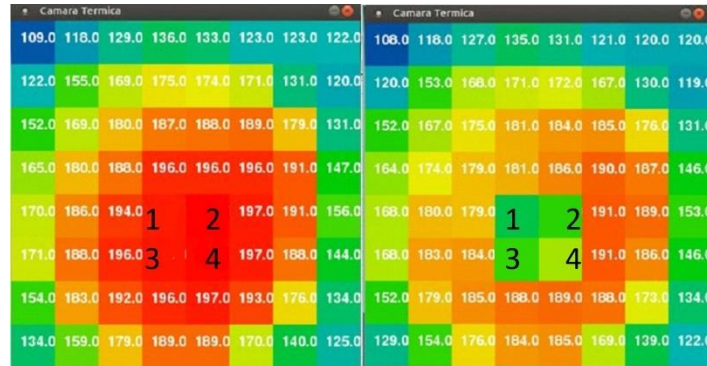


**Figura 7.** Línea de caracterización



**Figura 8.** Línea de caracterización

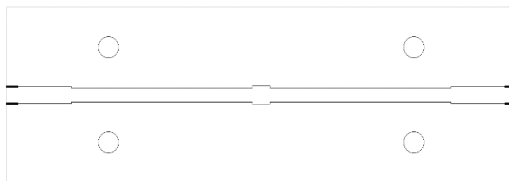
Con el fin de asegurarse de que el termómetro fue enfocado en la misma área que fue seleccionado, con la ayuda de un objeto frio fue posible ver el contraste de temperatura en la cámara y así realizar la correcta toma de datos.



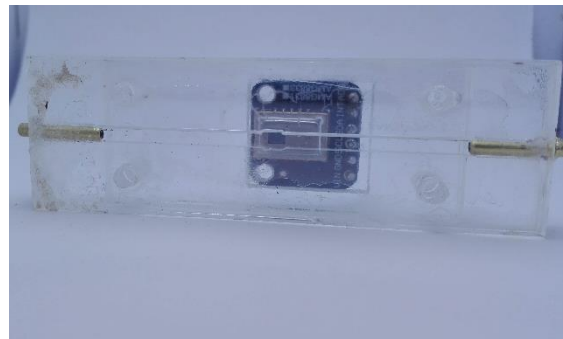
**Figura 9.** Selección área de caracterización.

## Prueba en sistemas microfluídicos

Una vez caracterizada la cámara se llevó a cabo el diseño de un sistema en acrílico sencillo por el que se bombearía agua a determinada temperatura. El lente de la cámara fue puesto en contacto con el acrílico de la capa aislante, quedando expuesto a una cavidad de la misma dimensión a este, con el fin de observar la temperatura del agua que fluye a través del canal.



**Figura 11.** Sistemas fluídico para prueba con agua.



**Figura 10.** Montaje primer experimento sistemas microfluídicos.

El sistema de la Figura 1110 es con el que se realizó la prueba de agua caliente. Los agujeros que se pueden observar sirven para ajustar la carcasa que se diseñó para la cámara.

**Manufactura sistema microfluídico:** Para realizar la manufactura de un sistema microfluídico se puede usar la herramienta AutoCAD, CorelDRAW, entre otras. Para diseñar estos dispositivos, pensando en las herramientas que el laboratorio de CMUA tiene para estos, es necesario tener en cuenta que las dimensiones de estos suelen ser

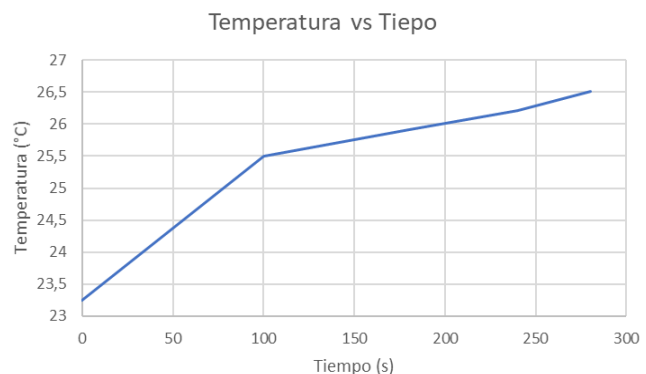
de 7.5 cm de largo por 2.5 cm de ancho, las medidas de un portaobjetos de microscopio. Una vez realizado el corte, las piezas son dejadas en una solución de alcohol y agua en una relación de por lo menos 50% alcohol durante 10 minutos con el fin de limpiarlas. Para limpiarlos más profundamente, esta solución se realiza dentro del sonicador, que mediante ondas sónicas remueven las impurezas de las piezas. El siguiente paso seleccionar el método con el que serán pegadas las piezas, los más comunes para el acrílico es con metil metacrilato o con metanol al 95%. La ventaja del primero es el tiempo y facilidad de pegado, pues solo se debe aplicar, unir las piezas, sostener por unos minutos y listo. El problema que presenta este pegante en la opacidad, pues al solidificarse se cristaliza, generando manchas blancas entre las placas. Para la aplicación de proyecto esto no es conveniente, la idea es mantener la transparencia del material lo más posible. En cuando al metanol, si se realiza el proceso adecuado la opacidad no es un problema: Para realizar la unión de las placas de debe hacer uso de la plancha de calentamiento y una prensa mecánica; El proceso a seguir es: precalentar la plancha a 110 °C, ubicar la primera capa sobre la plancha junto con la prensa de sistemas microfluídicos, depositar metanol al 95% sobre la superficie de dicha capa, colocar la siguiente capa, realizar presión con ayuda de la prensa, esperar 12 minutos, se retiran las placas y se realiza choque frio (puede ser con la misma solución de la limpieza) durante 30 segundos, se repite el proceso para las capas que sean necesarias.

El sistema manufacturado para el primer experimento es el mostrado en la Figura 10. En esta imagen se puede ver también la forma en la que va ubicada la cámara térmica.

#### Experimento sistema microfluídico 1

El primer experimento con sistema microfluídico consistió en hacer correr un flujo de agua a 34°C y medir con la cámara la temperatura vista al otro lado del acrílico. La bomba se configuró para que la velocidad del flujo sea de 200 ml/h, con este flujo se asegura que el canal esté siempre a tope.

La temperatura leída en la cámara fue uniforme, todos los pixeles indicaban alrededor de 22.5 °C al principio del experimento. Ya que la cámara apuntaba a todo el canal, se decidió tomar la medida del pixel del medio, en donde en teoría, la radiación infrarroja sería mayor. Al empezar el experimento no se notó un gran cambio en la medida de la temperatura sino hasta que transcurridos unos minutos. Después de 4 minutos con el agua fluyendo el experimento se detuvo y se encontró que la máxima temperatura medida fue de 26.5°C. Por la forma de los datos se puede intuir que la temperatura aún no se establecía, pues la respuesta de temperatura de un objeto tiende a



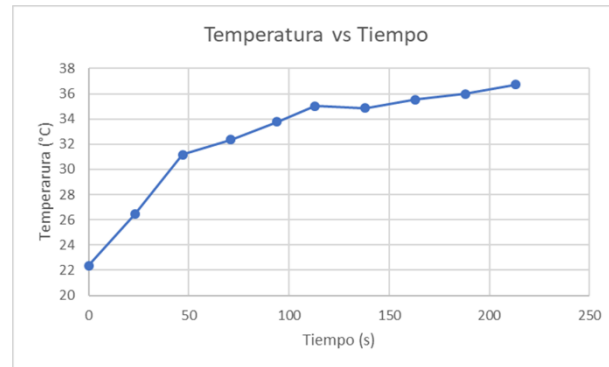
**Figura 12.** Respuesta en temperatura sistema microfluídico.

converger a la temperatura que se le está emitiendo.

Gracias al experimento anterior surgió la hipótesis de que se la cámara estaba recibiendo radiación del ambiente, pues el sistema está fabricado totalmente en acrílico y el fluido que corría a través de él era agua pura, permitiendo que luz de todas partes llegue a la cámara.

#### Experimento sistema microfluídico 1

El experimento 2 en sistema microfluídico se basó en el experimento 1, con la diferencia de que el sistema sería aislado lumínicamente mediante cinta aislante negra dejando libre solo el área en donde enfocaría la cámara. Se inició el experimento bombeando agua a 37°C a través del canal con la misma velocidad, 200ml/h.



**Figura 13.** Respuesta en temperatura sistema microfluídico 2.

Como resultado se obtuvo una respuesta exponencial inversa que se estabiliza alrededor de 37°C, temperatura a la que está el agua. En este experimento se comprobó la hipótesis generada en el anterior experimento. Recordando la teoría, la temperatura calculada es la que se relaciona con la frecuencia con mayor potencia y en este caso todo el ambiente estaba emitiendo radiación infrarroja en la frecuencia equivalente a 22°C. Al aislar el sistema la radiación infrarroja del agua se evidenció mucho más rápido que en el experimento anterior. En la Figura 13 se ve como en solo 15 segundos ya se evidenciaba un cambio de temperatura en 4°C.

Con la respuesta del experimento 2 se concluyó que la respuesta en temperatura y radiación infrarroja presentada por los sistemas microfluídicos sería óptima para las aplicaciones comunes en los laboratorios del CMUA, por lo que cumpliría con el objetivo de medir un cambio de temperatura dentro de un sistema microfluídico. Una vez concluido esto, se continuó con la siguiente parte del proyecto, el desarrollo de la interfaz gráfica intuitiva para los usuarios.

### 8.1 Descripción del Resultado Final

El resultado final del proyecto se divide en los ítems mencionados a continuación:

#### Lectura de datos

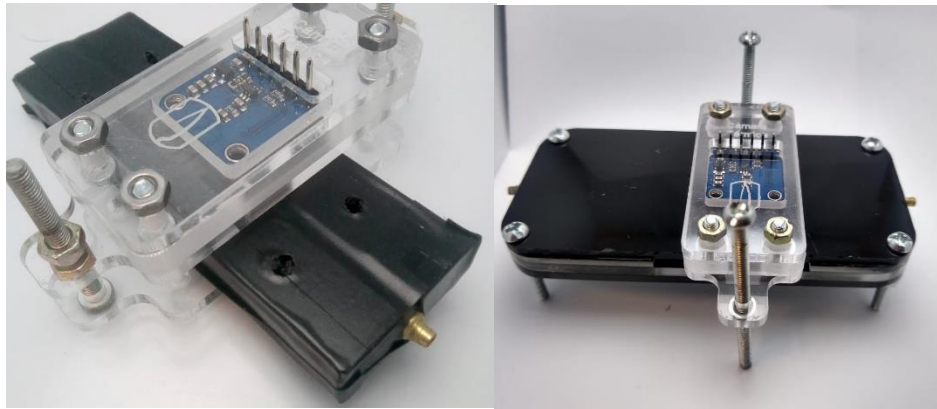
La lectura de los datos hace referencia a la cámara térmica AMG8831 usada para medir la temperatura de los sistemas microfluídicos. Esta sección físicamente es un módulo



separado del módulo de procesamiento con el objetivo de que sea cómodo de ubicar en los sistemas microfluídicos.

## Procesamiento de la información

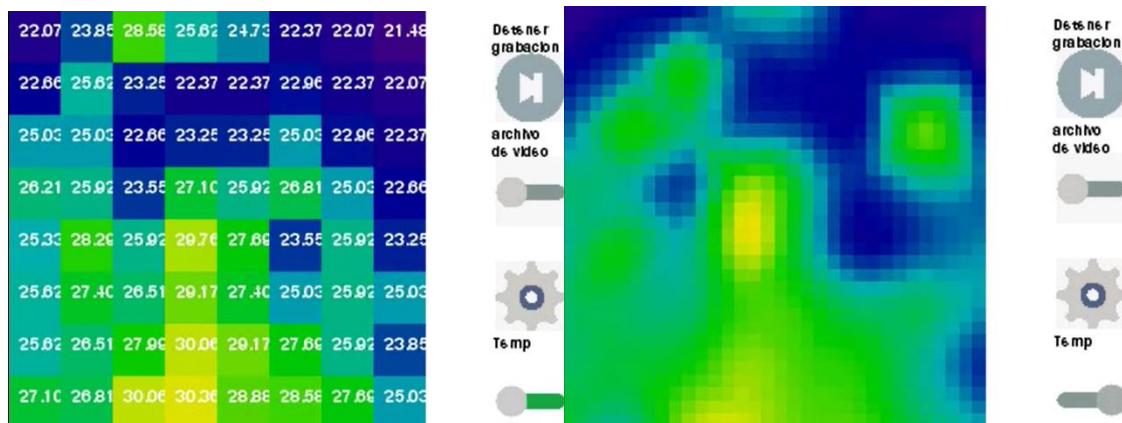
El procesamiento de la información en este proyecto hace referencia principalmente al ajuste que se le realiza a la información leída con el fin de obtener los valores en temperatura, a la organización de esta en una matriz y al diseño de un mapa de calor con esta misma.



**Figura 14.** Módulo para la cámara térmica.

## Presentación de la información

La información y el panel de control del prototipo están disponibles en una pantalla táctil de 2,4 pulgadas con resolución de 240x320 píxeles. En esta pantalla se muestra un mapa de calor elaborado con la información leída por la cámara en dos modos, el primero es un mapa de 8x8 rectángulos con el color correspondiente a la temperatura del píxel. En cada uno de estos rectángulos se imprime la temperatura en grados centígrados con el objetivo de que la toma de datos sea más fácil. El segundo modo consiste en una imagen de 30x30 píxeles generada a partir de los de 64 píxeles que lee la cámara. Esto permite ver con más claridad que puede estar viendo la cámara, sin embargo, no da información exacta de los píxeles.



**Figura 15.** Interfaz gráfica del dispositivo.

En el costado derecho de la pantalla se encuentran 4 botones que tienen la función de iniciar o detener una grabación, seleccionar si la grabación es en video o en archivo CSV, abrir el panel de configuraciones y seleccionar el tipo de imagen que se desea mostrar.

### Diseño estético y funcional del prototipo

En ingeniería el diseño también es importante, no solo por como luce, sino por cómo funciona. Al momento de realizar un experimento es importante tener una referencia de medición fija, es decir, tener la posibilidad de realizar toda una toma de datos con la cámara fija. Para esto se ha diseñado una pequeña coraza en acrílico que además de proteger la cámara, permite fijarla a un sistema microfluídico con la posibilidad de variar la distancia a la que esta se ubica mediante un juego de tonillos y tuercas como se puede observar en la Figura 14. Por otro lado, el módulo principal tiene la cualidad de ser pequeño, por lo que puede ser ubicado en cualquier lado, siempre y cuando el cable de la cámara pueda ser conectado.

Cuando se desee realizar una grabación solo se deberá poner una memoria USB en cualquiera de los puertos disponibles, y las grabaciones serán grabadas automáticamente ahí. De lo contrario, mediante un cable HDMI, ratón y teclado es posible acceder a la raspberry y administrar los archivos, que son guardados en la carpeta de documentos.



**Figura 16.** Prototipo final.



## 8.2 Trabajo computacional

### Interpolación 2D

La interpolación es método matemático para encontrar o aproximar funciones a partir de puntos. Entre los métodos de interpolación más comunes se encuentra la interpolación línea, interpolación por spline, interpolación por polinomio de Newton, entre otros. En el caso del proyecto, la interpolación cubica es una buena aproximación a los modelos de conducción de calor, por lo que las aproximaciones no estarían tan erradas.

Para realizar la interpolación de la matriz se utilizó la ayuda de la librería `scipy.interpolate` que mediante el método `Griddata` realiza una interpolación cubica a los datos que ingresan como parámetro. Este método devuelve una matriz de 4 veces las dimensiones de la que ingresó como parámetro.

## 9 VALIDACIÓN DEL TRABAJO

### 9.1 Metodología de prueba

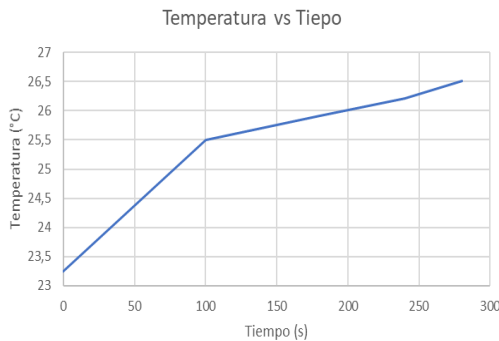
Las pruebas realizadas se dividen en dos, la prueba de medición, descrita en el experimento con sistemas microfluídicos 2 en la sección 8 del presente documento y la prueba de software que será descrita a continuación.

**Experimento de validación 1.** Como metodología de prueba para la medición de temperatura en sistemas microfluídicos se diseñó un sistema con los parámetros comunes en el CMUA con un canal básico por el que se dejaría correr un flujo de agua a 37°C. Durante todo el proceso se estarán tomando datos de temperatura en un solo lugar del microsistema con el fin de observar los cambios de temperatura en dicho lugar.

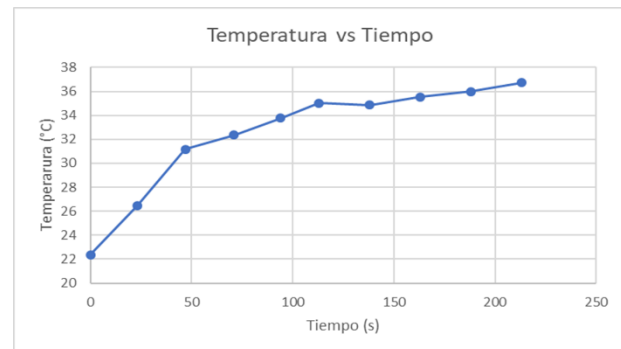
**Experimento de validación 2.** Para probar el software se diseñó un experimento similar, pero la cámara será fijada a 5 cm del microsistema con el objetivo de tener una imagen de cómo se comporta el calor en gran parte del sistema con ayuda de la imagen mejorada por interpolación. Este experimento quiere poner a prueba las funciones desarrolladas como la grabación en video y la interpolación de la información.

## 9.2 Validación de los resultados del trabajo

El experimento de validación 1 mostró un cambio en la temperatura muy lento para las aplicaciones que se pretendían en el proyecto, sin embargo, aquí se encontró un problema de interferencia de radiación externa, pues debido a que el material es translucido gran parte de la radiación externa estaba haciendo interferencia en la medida, haciendo que la temperatura dominante fuese a la que estaba el ambiente (alrededor de 21°C). Al aislar lumínicamente el mismo sistema los resultados fueron más alentadores ya que no se tenía dicha interferencia. En las figuras 17 y 18 se puede ver que mientras en el primer ejemplo tuvieron que pasar alrededor de 280 segundos para lograr un cambio de 4°C, mientras que en la segunda bastaron solo 15 segundos.

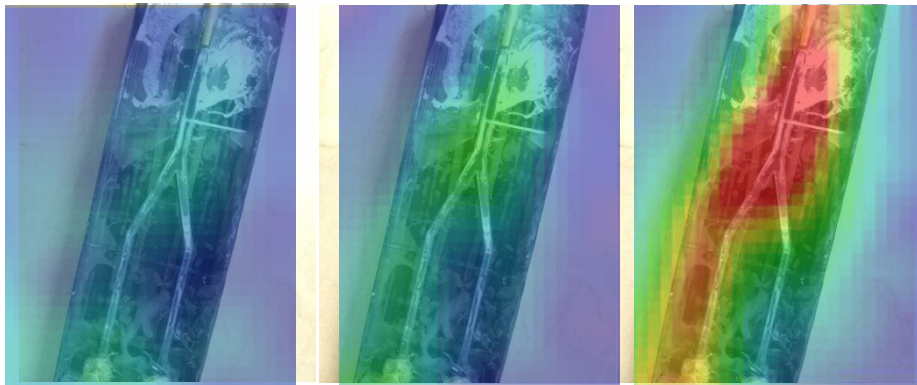


**Figura 18. Respuesta en temperatura sistema microfluídico.**



**Figura 17. Respuesta en temperatura sistema microfluídico 2.**

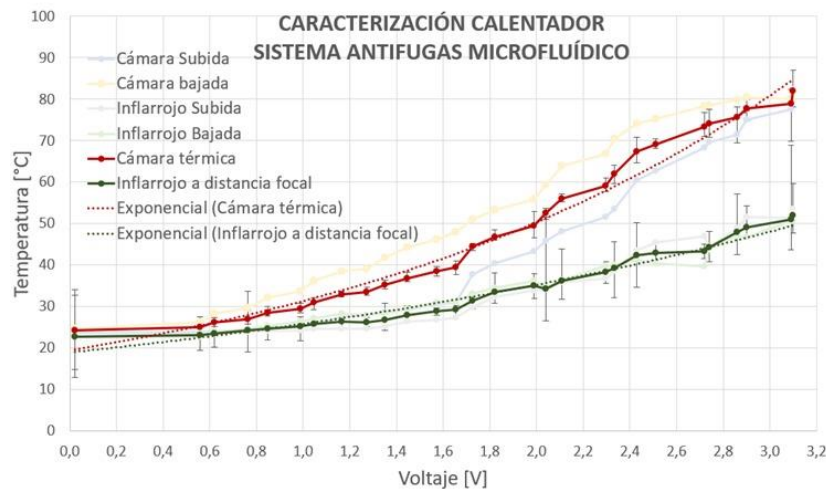
El experimento de validación 2 mostró que a pantalla utilizada presenta algunos problemas de precisión en el sistema táctil, sin embargo, esto no representa un problema relevante al proyecto. Por otro lado, al tomar un microsistema y bombear agua 90° centígrados se obtuvieron las siguientes imágenes.



**Figura 19. Comportamiento del sistema a 10 S, 20 S y 40 S después de iniciado el flujo de agua.**

Es importante aclarar de las imágenes en la Figura 19 son las imágenes de la cámara superpuestas a una imagen del sistema microfluídico con el fin de dar una perspectiva al mapa de calor que se está generando.

Uno de los trabajos de validación mas importantes, pero que no llevó a cabo una metodología, fue la utilización del prototipo en la caracterización de un sistema detector de fugas mediante el calentamiento de una resistencia de cobre, el calentador fue caracterizado usando prototipo en desarrollo y un termómetro infrarrojo, los resultados se observan en la Figura 20. Este trabajo fue desarrollado en paralelo como proyecto de grado del estudiante Santiago Tovar cuando aún estaba en desarrollo.



**Figura 20.** Caracterización calentador sistema anti fugas microfluídico.

En la imagen anterior se puede realiza una comparación en las incertidumbres que presenta cada una de las medidas, pues si se observa con detenimiento se ve claramente que las barras de error en las curvas de la cámara tienden a ser más pequeñas que la del termómetro infrarrojo. Esta mejora en la precisión se debe a que, en lugar de un sensor, la cámara utiliza 64 sensores para generar un mapa, lo cual hace que la toma de datos sea mucho mas segura, en cuando al lugar que se está midiendo y que sea mas exacta, pues dado el caso, se puede llegar a realizar un promedio entre los pixeles mas cercanos para reducir el error.

### 9.3 Evaluación del plan de trabajo

El plan de trabajo desarrollado en este proyecto resultó bien para los objetivos que se plantearon, pues se logró desarrollar una herramienta de laboratorio fácil de usar, con buena linealidad y un rango de medición bastante bueno. Sin embargo, si el plan de trabajo hubiese estado mejor estructurado se hubiese podido realizar mas estudios y experimentos con el fin de mejorar las mediciones.

## 10 DISCUSIÓN

En resumen, el producto final entregado al CMUA cumple con las especificaciones planteadas en la propuesta del proyecto y deja abiertas nuevas posibilidades en cuanto al alcance de este. El proyecto incluso sobrepasó el alcance propuesto, pues se trata de una herramienta que permite medir reacciones térmicas dentro de sistemas microfluídicos, pero también ha quedado como una cámara térmica para registrar cualquier evento térmico gracias a su intuitiva interfaz y las funciones que se implementaron en ella. Como se ha mencionado anteriormente, se ha realizado un trabajo de caracterización de la cámara AMG8831 donde se encontró que las mediciones son precisas y exactas, pues en la regresión lineal la medida del  $R^2$  fue de 0.9989, valor que indica que los puntos son lineales casi a la perfección. Por otro lado, la tasa de muestreo de la cámara es de 10Hz, lo que permite que procesos térmicos fugaces o constantemente cambiantes puedan ser registrados sin problema, siempre y cuando la reacción suceda frente a lente, sin ningún material extra en medio.

En cuanto al módulo de procesamiento se considera que ha cumplido y sobrepasado el alcance que se había planteado, pues, aunque no estuviese en la propuesta, algunas de las funciones implementadas se tomaron en cuenta por conversaciones con el equipo de trabajo de CMUA, como la función de guardar los datos solo conectando una USB. En la interfaz gráfica se resumieron algunos botones que quizá puedan hacer falta más adelante, pues con el primer botón se inician y detienen las grabaciones, pero con el segundo botón se escoge el formato de grabación. Esto implica que no podrán ser grabados archivos de video y archivos de datos CSV al mismo tiempo.

Con el fin de que fuese más fácil de usar, pero con gran potencial en el análisis de datos, en el menú de configuraciones se agregaron las opciones de cambiar el mínimo y el máximo de temperatura que serán representados con los colores en la pantalla. Por ejemplo, si se desea hacer un experimento o trabajo en el se sabe que las variaciones de temperatura serán entre 50 °C y 60 °C, estos valores pueden ser ajustados en el menú de configuraciones (Tercer botón) con el fin de que se detecten más fácilmente los cambios en temperatura; en el caso en que el rango de temperatura a medir sea grande, estos valores pueden ser modificados para que los colores en el mapa no se saturen. Adicionalmente, en el menú de configuraciones se encuentra también la opción de calibración por dos puntos, permitiendo que la cámara pueda ser calibrada cuando se considere necesario.

## 11 CONCLUSIONES

Al momento de realizar las pruebas finales al sistema se encontró que para medir reacciones térmicas dentro de sistemas microfluídicos fabricados en acrílico se presenta una interferencia de radiación infrarroja por las características propias del acrílico, pues este también irradia este tipo de luz. Sin embargo, la cámara es capaz de capturar pequeños cambios de temperatura en cuestión de segundos que evidencian cambios

térmicos al interior de los microsistemas. Por otro lado, un retraso de 2 minutos en la medición de un cambio de temperatura en un sistema microfluídico es aceptable en la mayoría de los procesos exotérmicos. Adicionalmente, la implementación del prototipo en otros proyectos da cuenta de la precisión y exactitud de la cámara frente a otras tecnologías, además de la utilidad que representa.

Como trabajo futuro en este proyecto queda la investigación en posibles sustitutos al acrílico, de modo que representen menor interferencia a la radiación infrarroja y así lograr la medición de procesos térmicos con un menor tiempo de retraso. Adicionalmente, queda pendiente la implementación de un sistema de zoom óptico con el objetivo de aumentar la resolución de cada pixel, llegando a generar mapas de calor en tamaños micrométricos. Por último, el software es algo que siempre se puede optimizar y gracias a que se cuenta con una cámara adaptable a casi cualquiera cosa, esta puede ser utilizada junto con Machine Learning para calcular velocidades de flujo mediante el calentamiento de la sustancia, o para llegar a reconocer que proceso térmico se está llevando a cabo con base en los cambios de temperatura en determinada área.

El proyecto se puede considerar de costo moderado, pues la tecnología infrarroja es de las mas costosas para medición de temperatura. En el caso del prototipo presentado el costo de manufactura es alrededor de \$120 US, pues la Raspberry tiene un costo aproximado de \$50 dólares, la cámara \$42 y los componentes adicionales aproximadamente \$35. Hay que tener en cuenta la tarjeta utilizada está siendo utilizada a menos de la mitad de su capacidad de procesamiento, por lo que bien tiene la capacidad de ser mejorada en cuanto a software sin modificar los costos o de ser reemplazadas con tecnologías mas baratas que permitan realizar las funciones ya establecidas en el prototipo.

## 12 AGRADECIMIENTOS

La motivación mas fuerte en la culminación de este proyecto de grado y de mis metas hasta el momento es la aportada por mis padres y hermana, pues es mi familia mi mas grande y constante apoyo en la realización de mis metas, aunque ellos no vivan conmigo en la ciudad, su apoyo se hace sentir de la mejor manera.

Agradezco también a la vida por haberme permitido llegar hasta acá en esta prestigiosa universidad, pues a pesar de las probabilidades que tiene alguien con orígenes rurales y de escasos recursos, he logrado lo que quisiera para todos mis compatriotas.

Agradezco a la vida el haber encontrado personas en mi camino que me ha ayudado en la realización de proyectos y en mi realización personal como lo es el profesor Johann Osma, quien además de ser mi asesor, ha sido mi concejero académico y de vida aun cuando no está en sus labores.

Doy gracias a las personas que estuvieron siempre con humilde disposición a ayudar en cualquier duda, o proceso que necesitara como lo fueron Juliana Noguera y Andrés

Aranguren, quienes siempre estuvieron asesorándome y ayudándome en la realización del proyecto.

Por último, pero no menos importantes agradezco a mis compañeros Santiago Tovar, Oscar Corzo y Sergio Sánchez el ser apoyos incondicionales en lo académico y personal, probablemente sin ellos me hubiese estancado.

## 13 Referencias

- [1] U. A. d. Barcelona, «Una tecnología mejora la eficacia de los dispositivos de análisis de microfluidos,» Barcelona, 2016.
- [2] P. a. K. A. a. C. A. a. G. K. a. N. S. a. K.-z. K. a. K. K. Yi, «Thermal analysis of nanofluids in microfluidics using an infrared camera,» *Lab on a chip*, vol. 12, pp. 2520-5, 2012.
- [3] I. B. Bautista, «Integración de microsensores y estructuras microfluidicas para la medida de parámetros químicos en sistemas biológicos,» Barcelona, 2014.
- [4] R. a. M. S. a. T. W. Sanchez-Andica, «Fabricación y Evaluación de un Dispositivo Microfluídico usando la Técnica de Serigrafía,» 10 2014.
- [5] Panasonic, «Infrared Array Sensor “Grid-EYE”».
- [6] V. Software, «Tabla de índices de refracción (para  $n = 589 \text{ nm}$  )».
- [7] T. m. i. society, «Optical properties,» de *Characterization and Failure Analysis of PLASTIC*, AMS Interntional, 2003, pp. 43-44.
- [8] C. France, «Infrared Trasmitting glass 9754».
- [9] S. a. A.-B. M. a. y. W. a. Y. R. a. D. D. a. Y. E. Oboudi, «Dispersion Characterization of conductive polymer,» *International journal of electrochemical science*, vol. 10, pp. 1555-1562, 2015.
- [10] Maxipolimeros, «Maxipolimeros,» [En línea]. Available: <https://www.mexpolimeros.com/pol%C3%ADmeros%20transparentes.html>. [Último acceso: 10 08 2019].
- [11] T. E. o. E. Britannica, «Encyclopaedia Britranica,» [En línea]. Available: <https://www.britannica.com/science/infrared-radiation>. [Último acceso: 27 11 2019].
- [12] I. David, «Sciencing,» 24 04 2017. [En línea]. Available: <https://sciencing.com/infrared-thermometers-work-4965130.html>. [Último acceso: 3 12 2019].
- [13] K. a. K. P. a. P. M. B. a. T. S. Ren, «Monitoring biochemical reactions using Y-cut quartz thermal sensors,» *Analyst*, vol. 136, pp. 2904-2911, 2011.
- [14] «International Crystal Laboratories,» International Crystal Laboratories, [En línea]. Available: [http://www.internationalcrystal.net/optics\\_06.htm](http://www.internationalcrystal.net/optics_06.htm). [Último acceso: 12 10 2019].

- [15] «RaspberryPi,» [En línea]. Available: <https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/>.
- [16] Panasonic, «Infrared Array Sensor Grid-Eye (AMG88),» 2015.
- [17] «Adafruit company,» [En línea]. Available: <https://www.adafruit.com/product/3538>.
- [18] «AstroAi,» [En línea]. Available: <https://www.astroai.com/product/Astroai-Digital-Laser-Infrared-Thermometer-380>.
- [19] piddlerintheroot, «Instructables circuits,» [En línea]. Available: <https://www.instructables.com/id/Thermal-Camera-AMG833-Raspberry-Pi/>.

## 14 APENDICES

**Código implementado en Python para la lectura, procesamiento y presentación de temperatura de la cámara térmica AMG8831.**

```
1. from Adafruit_AMG88xx import Adafruit_AMG88xx
2. import pygame
3. import os
4. import math
5. import time
6. import threading
7. import numpy as np
8. from scipy.interpolate import griddata
9. from pygame.locals import *
10. import glob
11. import csv
12. from time import time
13. from colour import Color
14. from os import listdir
15. import subprocess
16. import signal
17.
18. import spidev
19. from time import sleep
20. from lib_tft24T import TFT24T
21. import RPi.GPIO as GPIO
22. GPIO.setmode(GPIO.BCM)
23. GPIO.setwarnings(False)
24.
25. #Variables TouchScreen
26. DC = 24
27. RST = 25
28. LED = 15
29. PEN = 26
30.
31. TFT = TFT24T(spidev.SpiDev(), GPIO)
32. # Initialize display and touch.
```

```
33. TFT.initTOUCH(PEN)
34.
35. calib_scale240 = float(240)/3788 # Likely about 285
36. calib_scale320 = float(320)/3777 # Likely about 384
37. calib_offset240 = calib_scale240 * 180 # Likely about 28
38. calib_offset320 = calib_scale320 * 318 # Likely about 25
39.
40.
41.
42.
43. # Variables del sistema
44. salirMainScrn = False
45. corriendoMainScrn = False
46. salir = False
47. imprimirTem = True
48.
49. grabandoCSV = False
50. salirGrabacionCSV = True
51.
52. grabandoVideo = False
53. grabarVideo = False
54.
55.
56. csv_pixels = []
57. tiempo_inicio_csv = time()
58. #low range of the sensor (this will be blue on the screen)
59. MINTEMP = 20
60.
61. #high range of the sensor (this will be red on the screen)
62. MAXTEMP = 30
63.
64. #how many color values we can have
65. COLORDEPTH = 1024
66.
67. nPixels = 64
68.
69. os.putenv('SDL_FBDEV', '/dev/fb1')
70. pygame.init()
71.
72. #initialize the sensor
73. sensor = Adafruit_AMG88xx()
74.
75. points = [(math.floor(ix / 8), (ix % 8)) for ix in range(0, 64)]
76. grid_x, grid_y = np.mgrid[0:7:32j, 0:7:32j]
77.
78.
79. #Screen is 240x320
80. scrn_height = 240
81. scrn_width = 320
82.
83. #sensor is an 8x8 grid so lets do a square
84. map_height = min(scrn_height,scrn_width)
85. map_width = map_height
86.
87. #the list of colors we can choose from
88. blue = Color("indigo")
89. colors = list(blue.range_to(Color("red"), COLORDEPTH))
90. FONDO = (32, 30, 32)
91. BLANCO = (255, 255, 255)
92. COLOR_TEXTO = (50, 60, 80)
93.
```



```
94. #create the array of colors
95. colors = [(int(c.red * 255), int(c.green * 255), int(c.blue * 255)) for c in color
s]
96.
97. displayPixelWidth = map_width / 8
98. displayPixelHeight = map_height / 8
99.
100. displayPixelWidthCubic = float(map_width) / 30
101. displayPixelHeightCubic = float(map_height) / 30
102.
103. #Variables interfaz
104. widthBoton = 90
105. heightBoton = 90
106.
107. click = False
108.
109. #Cargamos las imagenes que serviran como botones
110. play_boton_imagen = pygame.image.load("imagenes/playbutton.jpeg")
111. config_boton_imagen = pygame.image.load("imagenes/settings.jpeg")
112. csv_swicth_imagen = pygame.image.load("imagenes/csv.jpeg")
113. video_swicth_imagen = pygame.image.load("imagenes/video.jpeg")
114. swicth_on_imagen = pygame.image.load("imagenes/swicthOn.png")
115. swicth_off_imagen = pygame.image.load("imagenes/swicthOff.png")
116.
117. #Botones configuracion
118. minTem_right_boton_imagen = pygame.image.load("imagenes/forward.png")
119. minTem_left_boton_imagen = pygame.image.load("imagenes/backward.png")
120. maxTem_right_boton_imagen = pygame.image.load("imagenes/forward.png")
121. maxTem_left_boton_imagen = pygame.image.load("imagenes/backward.png")
122.
123. #Se escalan las imagenes cargadas
124. tam_boton = (scrn_width - map_width)/2
125. play_boton_imagen = pygame.transform.scale(play_boton_imagen, [tam_boton, tam_boto
n])
126. config_boton_imagen = pygame.transform.scale(config_boton_imagen, [tam_boton, tam_
boton])
127. csv_swicth_imagen = pygame.transform.scale(csv_swicth_imagen, [tam_boton, tam_boto
n])
128. video_swicth_imagen = pygame.transform.scale(video_swicth_imagen, [tam_boton, tam_
boton])
129. swicth_off_imagen = pygame.transform.scale(swicth_off_imagen, [tam_boton, tam_boto
n])
130. swicth_on_imagen = pygame.transform.scale(swicth_on_imagen, [tam_boton, tam_boton]
)
131.
132. tam_boton2 = tam_boton/2
133. minTem_right_boton_imagen = pygame.transform.scale(minTem_right_boton_imagen, [tam_
boton2, tam_boton2])
134. minTem_left_boton_imagen = pygame.transform.scale(minTem_left_boton_imagen, [tam_b
oton2, tam_boton2])
135. maxTem_right_boton_imagen = pygame.transform.scale(maxTem_right_boton_imagen, [tam_
boton2, tam_boton2])
136. maxTem_left_boton_imagen = pygame.transform.scale(maxTem_left_boton_imagen, [tam_b
oton2, tam_boton2])
137.
138.
139. #Se crea el rectangulo de cada boton
140. r_boton_play = play_boton_imagen.get_rect()
141. r_boton_config = config_boton_imagen.get_rect()
142. r_boton_csv = csv_swicth_imagen.get_rect()
143. r_boton_on = swicth_on_imagen.get_rect()
```

```

144.
145.r_boton_direction = minTem_right_boton_imagen.get_rect()
146.
147.botones = []
148.
149.#Asignacion de posiciones
150.#top_left = map_width + int(((scrn_width - map_width) - tam_boton )/2)
151.top_left = scrn_width - tam_boton
152.top_up = scrn_height/10
153.boton_separacion = scrn_height/4
154.r_boton_play.topleft = [top_left, top_up ]
155.r_boton_csv.topleft = [top_left, top_up + boton_separacion]
156.r_boton_config.topleft = [top_left, top_up + boton_separacion*2]
157.r_boton_on.topleft = [top_left, top_up + boton_separacion*3]
158.
159.botones.append({'imagen': play_boton_imagen, 'imagen_dos': play_boton_imagen, 'rect': r_boton_play, 'selected': False, 'texto1': ["Iniciar","grabacion"], 'texto2': ["Detener", "grabacion"]})
160.botones.append({'imagen': csv_swicth_imagen, 'imagen_dos': video_swicth_imagen, 'rect': r_boton_csv, 'selected': False, 'texto1': ["archivo","CSV"], 'texto2': ["archivo", "de video"]})
161.botones.append({'imagen': config_boton_imagen, 'imagen_dos': config_boton_imagen, 'rect': r_boton_config, 'selected': False, 'texto1': [""], 'texto2': [""]})
162.botones.append({'imagen': swicth_off_imagen, 'imagen_dos': swicth_on_imagen, 'rect': r_boton_on, 'selected': True, 'texto1': ["Temp"], 'texto2': ["Temp"]})
163.
164.
165.#Se crea la pantalla que contendra la interfaz
166.flags = pygame.FULLSCREEN | pygame.DOUBLEBUF | pygame.RESIZABLE
167.lcd = pygame.display.set_mode((scrn_width, scrn_height), flags)
168.pygame.display.set_caption('Camara Termica')
169.textFont = pygame.font.SysFont("comicsansms", int(displayPixelWidth/2))
170.textFont_botones = pygame.font.SysFont("comicsansms", int(tam_boton/3))
171.lcd.fill((255,0,0))
172.pygame.display.toggle_fullscreen()
173.pygame.mouse.set_visible(False)
174.lcd.fill((255,255,255))
175.pygame.display.update()
176.
177.
178.#Funciones
179.def constrain(val, min_val, max_val):
180.    return min(max_val, max(min_val, val))
181.
182.def map(x, in_min, in_max, out_min, out_max):
183.    return (x - in_min) * (out_max - out_min) / (in_max - in_min) + out_min
184.
185.def dibujar_botones_iniciales(lista_botones):
186.    for boton in lista_botones:
187.        if boton['selected']:
188.            lcd.blit(boton['imagen_dos'], boton['rect'])
189.            textBoton = boton['texto2']
190.        else:
191.            lcd.blit(boton['imagen'], boton['rect'])
192.            textBoton = boton['texto1']
193.        for i in range(len(textBoton)):
194.            texto=textBoton[i]
195.            textLabel = textFont_botones.render(texto, True, (0, 0, 0))
196.            lcd.blit(textLabel, (scrn_width - tam_boton - 2,boton['rect'][1]-
197.                18 + (i*displayPixelWidth/3)) )

```

```

198.
199. def interfaz():
200.     global salir, map_height, map_width, displayPixelWidth, displayPixelHeight, lcd,
        botones, salirGrabacionCSV, grabarVideo
201.     dibujar_botones_iniciales(botones)
202.     while not salir:
203.         for event in pygame.event.get():
204.             if event.type == MOUSEBUTTONDOWN:
205.                 mouse = event.pos
206.                 presionarPantalla(mouse)
207.             if event.type == pygame.KEYDOWN:
208.                 # Permite terminar el programa
209.                 if event.key == pygame.K_q:
210.                     salir = True
211.                 # Alterna entre 'pantalla completa' y 'ventana'.
212.                 elif event.key == pygame.K_f:
213.                     pygame.display.toggle_fullscreen()
214.             pygame.time.wait(10)
215.
216. def presionarPantalla(mouse):
217.     global salirGrabacionCSV, grabarVideo, botones
218.     lcd.fill((255, 255, 255))
219.     for boton in botones:
220.         boton['selected'] = boton['selected'] ^ boton['rect'].colliderect([mouse[0]
        ]-1, mouse[1], 1, 1])
221.     salirGrabacionCSV = not (botones[0]['selected'] and not botones[1]['selected']
        )
222.     grabarVideo = (botones[0]['selected'] and botones[1]['selected'])
223.     dibujar_botones_iniciales(botones)
224.
225.
226. def mainSrcn():
227.     global corriendoMainScrn, csv_pixels
228.     corriendoMainScrn = True
229.     while not salirMainScrn:
230.
231.         #read the pixels
232.         try:
233.             pixelsMap = sensor.readPixels()
234.             # pixelsMap = np.random.rand(64,1)*100
235.             pixels = [map(p, MINTEMP, MAXTEMP, 0, COLORDEPTH - 1) for p in pixelsM
        ap]
236.
237.             if botones[3]['selected']:
238.                 for ix in range(8):
239.                     for jx in range(8):
240.                         pygame.draw.rect(lcd, colors[constrain(int(pixels[ix * 8 +
        jx]), 0, COLORDEPTH - 1)],
241.                                     (displayPixelHeight * ix, displayPixelWid
        th * jx, displayPixelHeight, displayPixelWidth))
242.                         textTemp = textFont.render(str((pixelsMap[ix * 8 + jx] / 0
        .25) * 0.2959 - 2.4836), True, (255, 255, 255))
243.                         lcd.blit(textTemp, (
244.                             displayPixelHeight * ix + int(displayPixelHeight / 5), dis
        playPixelWidth * jx + int(displayPixelWidth / 5)))
245.                     else:
246.                         # perdorm interpolation
247.                         bicubic = griddata(points, pixels, (grid_x, grid_y), method='cubic
        ')
248.
249.                 # draw everything

```

```

250.         for ix, row in enumerate(bicubic):
251.             for jx, pixel in enumerate(row):
252.                 pygame.draw.rect(lcd, colors[constrain(int(pixel), 0, COLO
RDEPTH - 1)],
253.                                     (7.5 * ix, 7.5 * jx, displayPixelHeightCu
bic, displayPixelWidthCubic))
254.             if grabandoCSV:
255.                 tiempo_actual_csv = time() - tiempo_inicio_csv
256.                 pixelsMap.insert(0, tiempo_actual_csv)
257.                 csv_pixels.append(pixelsMap)
258.
259.             pygame.display.update()
260.         except Exception as e:
261.             print(e)
262.         corriendoMainScrn = False
263.
264. x_file_csv = 0
265. def grabar_csv():
266.     global grabandoCSV, x_file_csv, tiempo_inicio_csv, csv_pixels
267.     tiempo_inicio_csv = time()
268.     grabandoCSV = True
269.     directorios = listdir("/media/pi")
270.     if len(directorios) > 0:
271.         raiz = "/media/pi/" + directorios[0] + "/"
272.     else:
273.         raiz = "/home/pi/Documents/"
274.
275.     dir = listdir(raiz)
276.     if not "csvFiles" in dir:
277.         try:
278.             os.system("mkdir " + raiz + "csvFiles")
279.         except Exception as e:
280.             print(e)
281.
282.     filename = raiz + "csvFiles/camara_termica.csv"
283.     if not raiz + "csvFiles/camara_termica" + str(x_file_csv) + ".csv" in glob.glob(raiz + "csvFiles/*.csv"):
284.         filename = raiz + "csvFiles/camara_termica" + str(x_file_csv) + ".csv"
285.     else:
286.         x_file_csv += 1
287.         grabar_csv()
288.     archivo = open(filename, "w")
289.     csv_escritor = csv.writer(archivo)
290.     #Encabezado del archivo
291.     text_header = ["Tiempo"]
292.     for i in range(nPixels):
293.         text_header.append("Pixel " + str(i))
294.     csv_escritor.writerow(text_header)
295.     archivo.close()
296.
297.     while not salirGrabacionCSV:
298.         archivo = open(filename, "a")
299.         csv_escritor = csv.writer(archivo)
300.         for i in range(len(csv_pixels)):
301.             csv_escritor.writerow(csv_pixels[i])
302.         csv_pixels = []
303.         archivo.close()
304.         pygame.time.wait(1000)
305.     grabandoCSV = False
306.
307. x_file_video = 0

```

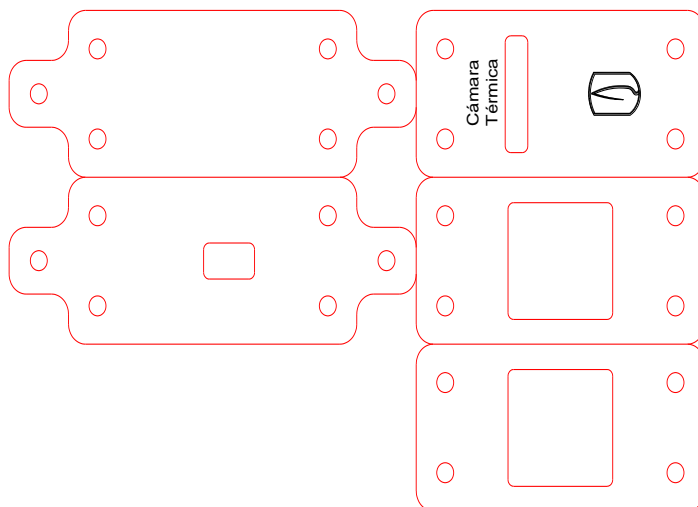
```

308. def iniciarGrabarVideo():
309.     global p, grabandoVideo, x_file_video
310.     grabandoVideo = True
311.     directorios = listdir("/media/pi")
312.     if len(directorios) > 0:
313.         raiz = "/media/pi/" + directorios[0] + "/"
314.     else:
315.         raiz = "/home/pi/Documents/"
316.
317.     filename = raiz + "videoFiles/camara_termica" + str(x_file_video) + ".mkv"
318.     dir = listdir(raiz)
319.     if not "videoFiles" in dir:
320.         try:
321.             os.system("mkdir " + raiz + "videoFiles")
322.         except Exception as e:
323.             print(e)
324.
325.     if not raiz + "videoFiles/camara_termica" + str(x_file_video) + ".mkv" in glob.
        glob(raiz + "videoFiles/*.mkv"):
326.         filename = raiz + "videoFiles/camara_termica" + str(x_file_video) + ".mkv"
327.     else:
328.         x_file_video += 1
329.         iniciarGrabarVideo()
330.     comando = "ffmpeg -f alsa -r 10 -f x11grab -
        s $(xdpyinfo | grep dimensions | awk '{print $2;}') -i ${DISPLAY} -
        c:v libx264rgb -crf 0 -preset:v ultrafast -
        af aresample=async=1:first_pts=0 "+ filename
331.     p = subprocess.Popen(comando, shell=True, stdin=None, stderr=subprocess.STDOUT
        , stdout=subprocess.PIPE, close_fds=True)
332.
333. def detenerVideo():
334.     global p, grabandoVideo
335.     os.killpg(os.getpgid(p.pid), signal.SIGINT)
336.     grabandoVideo = False
337.
338. def touchScreen():
339.     press = False
340.     mouse = []
341.     while not salir:
342.         while not TFT.penDown() and not salir:
343.             if press:
344.                 presionarPantalla(mouse)
345.                 press = False
346.                 pygame.time.wait(10)
347.                 x = TFT.readValue(TFT.X)
348.                 y = TFT.readValue(TFT.Y)
349.                 x2 = int(y * calib_scale320 - calib_offset320)
350.                 y2 = int(x * calib_scale240 - calib_offset240)
351.                 mouse = [x2, y2]
352.                 press = True
353.
354. def configuracion():
355.     global salirMainScrn, salirGrabacionCSV, grabarVideo
356.     salirMainScrn, salirGrabacionCSV = True
357.     grabarVideo = False
358.     backgroundColor = (178, 181, 178)
359.     pygame.draw.rect(lcd, backgroundColor, (0,0, map_width, map_height))
360.
361.
362.

```

```
363.
364. def principal():
365.     global salirMainScrn, salirGrabacionCSV
366.     threading.Thread(target=thouchScreen).start()
367.     while not salir:
368.         if not salirMainScrn and not corriendoMainScrn:
369.             threading.Thread(target=mainSrcn).start()
370.         if not salirGrabacionCSV and not grabandoCSV:
371.             threading.Thread(target=grabar_csv).start()
372.         if grabarVideo and not grabandoVideo:
373.             iniciarGrabarVideo()
374.         elif not grabarVideo and grabandoVideo:
375.             detenerVideo()
376.         pygame.time.wait(100)
377.         salirMainScrn = True
378.         salirGrabacionCSV = True
379.
380.
381.
382.
383.
384. threading.Thread(target=interfaz).start()
385. threading.Thread(target=principal).start()
```

## Planos carcaza cámara AMG8833



## Planos carcasa modulo principal

