

XIKUA Boletín Científico de la Escuela Superior de Tlahuelilpan
13° Congreso Internacional de Computación
Inteligencia artificial: Presente y futuro
Red Iberoamericana de Computación
Vol. 12, Número Especial (2024) 57-64

Un dispositivo IoT de fácil acoplamiento para la supervisión de la presión del gas LP
An easy-to-connect IoT device for monitoring LP gas pressure

Eduardo C. Camargo-Burgos^a, Antonio Alarcón-Paredes^b, Gustavo A. Alonso-Silverio^c

Abstract:

Liquefied Petroleum Gas (LPG) or commonly known as LP Gas is a flammable fuel used in different activities within the home, from cooking food, heating, emergency power generators, among others, therefore, its use must be following the relevant rules and regulations along with proper facilities which must be reviewed regularly as well as gas connections. Possible leaks of this fuel can cause serious accidents to the point of possibly causing explosions or poisoning due to excessive inhalation. In view of this problem, different works have been carried out focused on the timely detection of LP Gas leaks in a home environment, following this theme, in the present work a device was developed to monitor the content of LP Gas in tanks for home and semi industrial use, this by means of an ESP32-CAM module that captures photographs to a commercial analog pressure gauge, sending them for storage and processing to a web server, which through artificial vision techniques determine the value indicated by the pressure gauge, consulting the results and storing them through mobile devices.

Keywords:

Computer vision, Internet of Things (IoT), cloud computing, NodeMCU, Python.

Resumen:

Los Gases Licuados del Petróleo (GLP) o comúnmente denominados como Gas LP es un combustible inflamable usado en diferentes actividades dentro del hogar, desde la cocción de alimentos, en calefacción, generadores de energía de emergencia, entre otros, por lo tanto, su uso debe ser siguiendo las normas y regulaciones pertinentes junto a adecuadas instalaciones las cuales deben ser revisadas regularmente como también a las conexiones de gas. Ya que posibles fugas de este combustible desencadenan en graves accidentes al punto de posiblemente ocasionar explosiones o intoxicaciones por su inhalación excesiva. Ante esta problemática han sido realizados diferentes trabajos enfocados en la detección oportuna de fugas de Gas LP en un ambiente casa habitación, siguiendo esta temática, en el presente trabajo se desarrolló un dispositivo para el monitoreo del contenido de Gas LP en tanques de uso casero y semi industrial, esto por medio de un módulo ESP32-CAM que captura fotografías a un manómetro analógico comercial, enviándolas para su almacenamiento y procesamiento a un servidor web, el cual por medio de técnicas de visión artificial determinan el valor indicado por el manómetro, consultando los resultados y el almacenamiento de estos mismos a través de dispositivos móviles.

Palabras Clave:

Visión artificial, Internet of Things (IoT), cloud computing, NodeMCU, Python

^a Autor de Correspondencia, Universidad Autónoma de Guerrero (UAGro), <https://orcid.org/0009-0003-1205-7469>, Email: 16325885@uagro.mx

^b Universidad Autónoma de Guerrero (UAGro), <https://orcid.org/0000-0002-9785-1252>, Email: aalarcon@uagro.mx

^c Universidad Autónoma de Guerrero (UAGro), <https://orcid.org/0000-0002-2699-140X>, Email: gsilverio@uagro.mx

Introducción

Los Gases Licuados del Petróleo (GLP) o comúnmente denominados como Gas LP, son hidrocarburos combustibles de fase gaseosa obtenidos a partir del refinado del petróleo por destilación fraccionada de butano y propano que, al combinarlos con el aire, se convierte en una mezcla inflamable, es por ello que su instalación, manipulación y almacenamiento debe llevarse a cabo empleando las herramientas apropiadas y siguiendo las reglas de instalación adecuadas. Un instrumento importante por tener en consideración es el manómetro, el cual es usado para la medición del nivel de presión y contenido de hidrocarburos en recipientes. A pesar de que este hidrocarburo cuenta con un nivel de inflamabilidad muy bajo, es necesario y de importancia revisar periódicamente su instalación para la prevención de fugas y garantía de su estanqueidad, ya que esto puede desencadenar en posibles pérdidas humanas, esto debido a explosiones o bien por asfixia al inhalar este hidrocarburo en altas concentraciones ^[1].

La Internet de las cosas (IoT) es una tecnología que ha demostrado ser de utilidad en la implementación de diversos sistemas enfocados al monitoreo. El concepto de IoT define a la manera en la cual, diferentes sistemas electrónicos son capaces de interactuar y comunicarse entre sí, ya sea en el control o monitoreo de un proceso, sistema, actividad o cualquier objeto de interés dentro de un determinado ambiente, intercambiando la información y/o datos recopilados para reaccionar y realizar de manera automática una acción, un proceso o servicio (interviniendo o no, la mano humana) ante algún evento ocurrido ^[2].

Prueba de ello, es la propuesta de un sistema para el monitoreo de manómetros, reportado por ^[3], el cual, utilizando el controlador Raspberry Pi 4 Modelo B, la webcam MetFut 1080P y el servidor de nube virtual ThingSpeak, realiza el registro, digitalización y graficación del comportamiento de los valores de presión obtenidos por un manómetro analógico. Por medio de la captura de fotografías a dicho manómetro y aplicando en ellas, técnicas de visión artificial, realiza primeramente la identificación de la aguja del manómetro, y posteriormente determina en que posición se encuentra dentro de la escala numérica de los valores de presión marcados en el manómetro, indicando así, la lectura del nivel de presión de un recipiente. Dicho sistema muestra resultados favorables con un ± 2 por ciento de tolerancia de error, sin embargo, al encontrarse enfocado solo al área industrial, limita su alcance hacia un público más amplio, así como también, los sensores y dispositivos utilizados en su desarrollo son de alto costo o bien no estar disponibles de manera accesible a todo público, además, al utilizar un servidor comercial con funciones establecidas, puede resultar en el desperdicio de recursos en funciones no utilizadas asimismo, limita su capacidad en convertirlo en un sistema no escalable.

Existen otros trabajos realizados con el fin de alertar a los usuarios, la presencia de fugas de Gas LP, como los realizados por ^[4] y ^[5], siendo estos, propuestas de dispositivos con un enfoque inteligente para el monitoreo y detección de Gas LP en el ambiente, los cuales alertan a sus usuarios por medio de un mensaje SMS o llamada telefónica, utilizando el módulo GSM Modem. Cada uno de ellos monitorea dentro del área en donde han sido colocados, la concentración de gas LP en el aire, esto por medio de sensores. En la propuesta de ^[5], hace uso del sensor MQ-2, el cual, cuando detecta Gas LP en el aire, notifica a su usuario por medio de una llamada telefónica y un mensaje SMS acerca de este evento, en cambio, la propuesta de ^[4] hace uso de los sensores MQ-5, MQ-6 y el sensor DHT11 para la lectura de la temperatura y humedad, estos sensores recopilan datos e información del ambiente, enviándolos

por medio de un NodeMCU DEVKIT 1.0 hacia una plataforma central, en donde se procesan, almacenan (dentro de una base de datos SQL) y analizan en tiempo real, estos valores, notificando de manera automática al usuario por medio de un mensaje SMS, la detección de Gas LP en el ambiente, dando así una respuesta rápida en la prevención de accidentes o daños. No obstante, los datos obtenidos en este sistema solo son almacenados de manera local, lo que impide que los usuarios puedan acceder a ellos externamente, además, en ambos sistemas, el alertar a los usuarios solamente por medio de servicios de telefonía celular, genera que dichos sistemas sean completamente dependientes de un dispositivo móvil, así como también, del servicio de SMS o llamadas, generando esto, costos adicionales al sistema, estos costos varían tarifas según la compañía telefónica en la cual se encuentre registrado el teléfono del usuario, por lo tanto, si el servicio de telefonía no es cubierto, los sistemas perderían su funcionalidad.

Otro trabajo realizado, es el propuesto por ^[6] siendo este la descripción de las distintas etapas que ha atravesado el desarrollo de una plataforma denominada Gaszen, la cual, fue diseñada para el monitoreo de niveles de Gas LP en tiempo real en hogares o empresas, esto por medio de sensores y dispositivos electrónicos diseñados por ellos mismos, los cuales, deben ser acoplados a recipientes de tanques de Gas LP, así como también, conectar estos dispositivos a una red inalámbrica para el envío de los valores obtenidos hacia una aplicación móvil o web, permitiendo así a sus usuarios acceder a esta información desde cualquier lugar y momento. Sin embargo, por la robustez de la plataforma, el acceso a este servicio es exclusivamente para aquellas personas y empresas que cuente con un nivel adquisitivo adecuado para pagar tanto la cantidad de dispositivos necesarios como también el servicio de dicha plataforma, asimismo, el potencial de este sistema no podría ser aprovechado al máximo dentro de entornos hogareños de tamaño reducido o bien, llegar a resultar para algunos usuarios algo complejo de manejar.

El presente trabajo describe un sistema de monitoreo de Gas LP, diseñado para su uso en entornos domésticos o industriales, siendo sus principales características, su facilidad de instalación y bajo costo, ya que, se basa en el uso de un manómetro analógico accesible y comercial junto con plataformas de hardware abierto. El sistema está integrado por una placa de desarrollo open hardware, ESP32-CAM, la cual cuenta con un módulo WiFi y Bluetooth junto a una cámara OV2640 de 2MP para la captura y envío de fotografías hacia servidores web. Con dicha cámara, son capturadas fotografías a un manómetro, enviándolas hacia un servidor web (PythonAnywhere) para su procesamiento, mediante el lenguaje de programación Python y algoritmos de visión artificial, obteniendo de ello, un historial gráfico de los registros del nivel cuantitativo y cualitativos del volumen de Gas LP obtenidos de un recipiente (tanque), almacenando estos datos en Google Drive, brindando así a sus usuarios la posibilidad de acceder a dicha información desde cualquier lugar mediante una aplicación web desarrollada con el framework Flask.

Objetivos

El objetivo principal de este proyecto es desarrollar un dispositivo electrónico destinado al monitoreo de tanques de Gas LP en entornos domésticos o industriales. La particularidad de este dispositivo radica en su bajo costo y facilidad de instalación, ya que se basa en el uso de un manómetro analógico comercial junto con plataformas de hardware abierto y servicios web.

Los objetivos específicos considerados son: Programar un módulo ESP32-CAM para capturar fotografías del manómetro y enviarlas a servidores web para su posterior procesamiento, realizar el procesamiento de las fotografías utilizando algoritmos de visión

artificial para determinar el nivel de gas LP indicado por el manómetro, almacenar las fotografías capturadas y los resultados del procesamiento en Google Drive, desarrollar una aplicación web utilizando el framework Flask que integre todas las funcionalidades anteriormente mencionadas, y finalmente, brindar a los usuarios la posibilidad de acceder a la información de monitoreo desde cualquier lugar y dispositivo móvil, a través de la aplicación web.

Metodología y Procesos de Desarrollo

La metodología por seguir en el desarrollo del proyecto puede ser consultada a continuación (ver Figura 1):

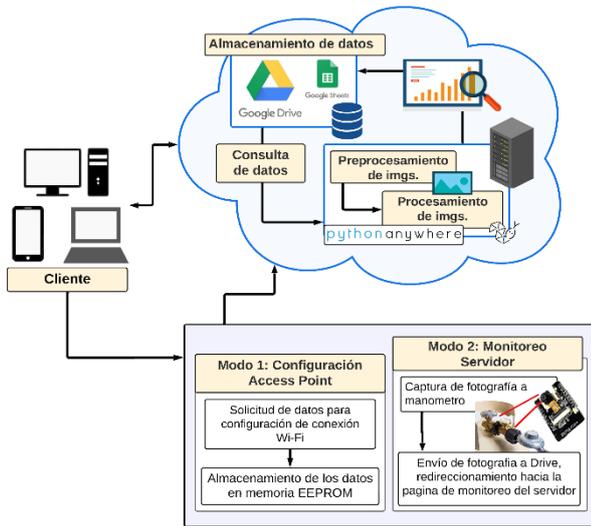


Figura 1. Esquema de la metodología propuesta.

El sistema se encuentra orientado a un entorno web, en donde todo usuario (denominado como cliente), realiza peticiones de servicios a servidores web, y estos últimos, retornan una respuesta a cada petición solicitada. En este caso, el cliente realiza la petición al ESP32-CAM (configurado como servidor) de capturar una fotografía al manómetro, posteriormente el ESP32-CAM envía esta fotografía a Google Drive para su almacenamiento, una vez confirmado el correcto almacenamiento por parte de Google Drive, el ESP32-CAM redirige al cliente hacia la aplicación web alojada en el servidor PythonAnywhere.

La aplicación web del servidor PythonAnywhere, solicita a Google Drive la última fotografía almacenada, en dicha imagen, la aplicación web implementa algoritmos de visión artificial para determinar el nivel de Gas LP del recipiente que se encuentre conectado al manómetro, retornando finalmente al cliente, el resultado obtenido.

Para llevar a cabo el desarrollo del sistema, se consideró la construcción de un dispositivo electrónico, el cual integra a un NodeMCU, específicamente el modelo ESP32-CAM, quien se encontrará interactuando con una aplicación web (implementada con el framework de Flask), la aplicación web, se encontrará alojada dentro de un servidor web basado en Python, haciendo uso de la API de Google Drive y librerías para el procesamiento digital de imágenes, permitiendo así, realizar el monitoreo de un manómetro analógico conectado a un tanque de Gas LP. Los materiales y tecnologías por utilizar pueden ser consultados en la Tabla 1 y la Tabla 2.

Tabla 1. Lista de materiales y componentes electrónicos usados en la construcción del dispositivo propuesto.

Cantidad	Nombre	Imagen
1	ESP32-CAM (junto a la cámara OV2640)	
3	Resistencia de carbón, de 1 Watt, al 5% de tolerancia, de 330 Ohms	
1	Resistencia de carbón, de 1/2 Watt, al 5% de tolerancia, de 1k Ohm	
3 (1 de c/u)	LED de 5mm (color verde claro, azul y rojo)	
2	Switch miniatura, de balancín, de 1 polo, 1 tiro, 2 posiciones	
1	Gabinete plástico de 13.5 x 4.9 x 7.5 cm	
1	Manómetro medidor regulador adaptador para tanque de Gas LP	
1	Manguera con regulador QCC1 para Gas LP/Propano 1.5 Metros	

Los componentes electrónicos junto con el módulo ESP32-CAM, se incorporaron en un circuito electrónico, siendo este, la parte física que conformará al sistema. Este dispositivo tendrá las funciones de capturar fotografías y comunicarse con los servicios de la nube (Google y PythonAnywhere). La descripción del circuito electrónico propuesto (ver Figura 2) puede ser consultada a continuación:

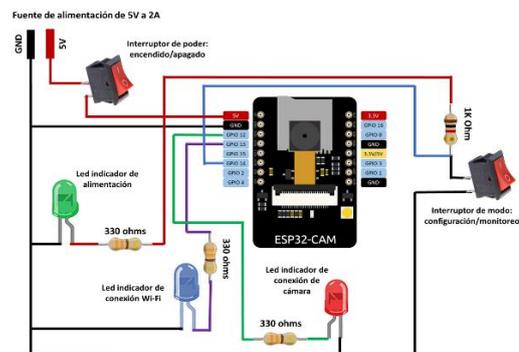


Figura 2. Circuito electrónico del dispositivo propuesto.

- **Interruptor de poder:** Este interruptor tiene la función de encender o apagar el dispositivo.
- **Interruptor de modo:** Este interruptor indica en qué modo de funcionamiento debe inicializarse el ESP32-CAM, siendo estos:
 - 1) **Modo configuración:** El ESP32-CAM es configurado en modo Access Point, solicitando al cliente, el nombre identificador de la red a la que se desea conectar (SSID) y su respectiva contraseña, almacenando esta información en su memoria EEPROM, para realizar cada vez que sea encendido el dispositivo, la conexión con la red deseada de manera automática.

2) *Modo monitoreo:* El ESP32-CAM es configurado en modo servidor, el cual se encontrará a la espera de las peticiones del cliente, ya sea para la lectura del manómetro o bien, la consulta del historial de las lecturas realizadas. Indicando así que, en este modo, el ESP32-CAM, estará en comunicación con el servidor PythonAnywhere.

Cabe mencionar que la comunicación entre el cliente y el ESP32-CAM (inicializado en cualquiera de sus dos modos), es a través de un protocolo de comunicación HTTP.

Los leds indicadores en el circuito electrónico tienen como función, alertar al usuario de sucesos ocurridos durante su ejecución, tales como:

- *Led indicador de alimentación:* Indicará al encenderse que el circuito se encuentra energizado correctamente por una fuente de alimentación.
- *Led indicador de conexión Wi-Fi:* Cuando el dispositivo fue inicializado en modo monitoreo, encenderá para indicar que la conexión entre la red Wi-Fi y el ESP32-CAM ha sido correcta.
- *Led indicador de conexión de cámara:* Este led indicará al encenderse que la cámara conectada al ESP32-CAM, se encuentra conectada correctamente y lista para realizar la captura de fotografías.

Tabla 2. Tecnologías usadas en el desarrollo del sistema para el dispositivo propuesto.

Lenguaje de programación Python (versión 3.7.0) junto a las librerías:	
Flask	Numpy
pydrive2	Pandas
OpenCV	urllib.request
Entorno de desarrollo Arduino (versión 2.1.0) junto a las librerías:	
WebServer.h	esp_camera.h
WiFiClientSecure.h	ESPmDNS.h
Servicios de Google:	
Apps Script	API Google Drive
Servidor de alojamiento web basado en el lenguaje de programación Python:	
PythonAnywhere	

Funcionamiento del sistema

Al inicializar el dispositivo en modo monitoreo, es configurado el ESP32-CAM como un servidor web, el cual se encontrará a la espera de un cliente. Una vez que un cliente haya establecido comunicación con el ESP32-CAM, el cliente realizará la petición de tomar una lectura del manómetro, siendo esto la captura de una fotografía a dicho instrumento por parte del ESP32-CAM. Posteriormente, esta fotografía por medio de un Script codificado con Apps Script, es enviada hacia una ubicación específica dentro de una cuenta de Google Drive. Google Drive generará y enviará una respuesta al ESP32-CAM, si esta operación fue realizada exitosamente o no, sí es así, es entonces que el ESP32-CAM redirecciona al cliente hacia la aplicación web del servidor

PythonAnywhere, para realizar el procesamiento de la fotografía, en caso contrario, será mostrado al cliente un mensaje indicando que esta acción no puede ser realizada.

Una vez establecida la comunicación entre el cliente y la aplicación web (servidor PythonAnywhere), la aplicación web solicita, mediante una credencial firmada para la API de Google Drive, la última fotografía almacenada dentro de la ubicación indicada de Google Drive. Al ya contar la aplicación con esta fotografía, es iniciado su procesamiento; este procedimiento conlleva etapas que son descritas e ilustradas (ver la Figura 3) a continuación:

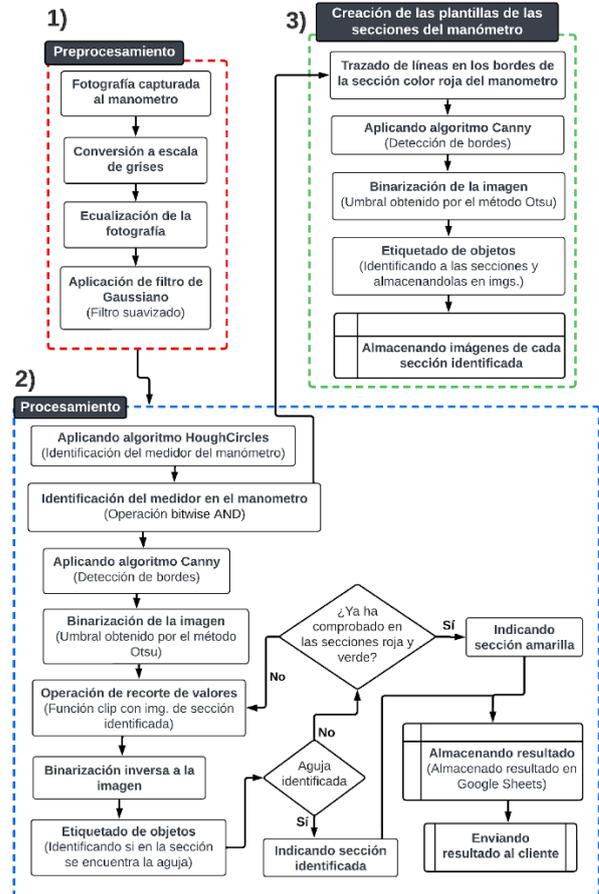


Figura 3. Diagrama de flujo del procesamiento de la fotografía.

- 1) *Preprocesamiento de la imagen:* En primera instancia, la imagen es convertida a escala de grises, sobre la cual se lleva a cabo una ecualización con la finalidad de mejorar el contraste y resaltado de detalles y texturas. Posteriormente, se aplica un filtro Gaussiano para eliminar posibles ruidos o artefactos generados; esta imagen está lista para su procesamiento.
- 2) *Procesamiento de la imagen:* Con el fin de identificar el área circular del manómetro en la imagen, se aplicó la transformada de Hough, siendo éste un método usado en el reconocimiento de patrones en imágenes, específicamente aquellos que conformen a líneas o círculos [7]. Mediante el método HoughCircles, se realizó la identificación de objetos con formas circulares, cuya área y posición es determinada por el mismo método. Del total de objetos identificados es seleccionado aquel de mayor área, y posteriormente, dibujado en una imagen binaria siendo el área de color blanco, la forma del círculo. Usando a esta imagen binaria como una máscara para segmentar, es aplicada una operación AND con la imagen

preprocesada, resultando de esto, una nueva imagen en la cual solo se encuentre el objeto de interés, siendo en este caso el medidor del manómetro.

El manómetro utilizado en este sistema cuenta con tres escalas de medición basadas en colores: verde, amarillo y rojo, las cuales son indicativas de un nivel de gas normal, un nivel de gas bajo y una alerta de relleno del tanque, respectivamente. Por tal situación, se propone realizar una segmentación en la fotografía capturada a este manómetro, con el objetivo de identificar a estas secciones de color y crear plantillas que ayuden en la determinación de la posición de la aguja en el manómetro, esto por medio de una binarización por Otsu y la aplicación del algoritmo de Canny [8] para la identificación de bordes; este procedimiento es detallado a continuación.

- 3) *Creación de las plantillas de las secciones del manómetro:* En este punto se realizó una pequeña etapa de prueba y error para la identificación de las secciones anteriormente descritas; cabe mencionar que esta etapa no es realizada por el servidor cada vez que deba procesar una fotografía, solamente es realizada cuando sea necesaria la creación de las plantillas. Primeramente, para ayudar al algoritmo de Canny en la correcta identificación de los contornos que conforman a estas secciones, fue realizada en la sección de color rojo, trazos de líneas rectas alrededor de sus bordes, esto debido a que al encontrarse la aguja del manómetro en esta sección (no estando conectado el manómetro a un tanque), generaba problemas en su identificación.

Una vez realizada la acción anterior e identificados los contornos de las secciones por Canny, es aplicado un umbral obtenido por el método Otsu, para crear una imagen binaria, en dicha imagen es realizado un etiquetado de componentes conexos, obteniendo de ello, información estadística de cada componente (objeto) identificado, tales como su área en píxeles, su posición dentro de un cuadro delimitador, las coordenadas de sus centroides, entre otras. Analizando de manera visual, trazando cada objeto en una imagen binaria, así como también analizando el valor de su área en píxeles, se realizó una depuración de objetos, hasta seleccionar aquellos que correspondan a las secciones deseadas. Cada una de las secciones identificadas de los niveles indicadores de Gas LP del manómetro, son almacenadas sus imágenes binarias (ver Figura 4), distinguiéndolas por un número: el valor de 3 para la sección verde, 2 para la sección amarilla y el valor de 1 para la sección roja. Finalizando así la etapa de la creación de las plantillas.

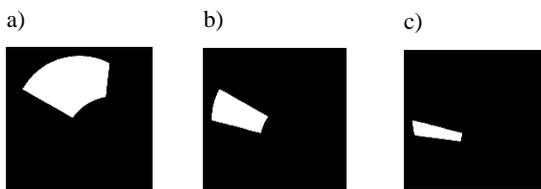


Figura 4. Imágenes de cada sección identificada, a) área color verde, b) área color amarillo y c) área color rojo.

Volviendo al punto en el cual se cuenta con la imagen del área del medidor del manómetro identificada, es aplicado el algoritmo Canny para identificar los contornos en esta imagen, y posteriormente, binarizarla usando como umbral el obtenido por Otsu.

A la imagen resultante es aplicada una operación de truncamiento o recorte de valores (clip) con cada una de las plantillas creadas (ver Figura 4) iniciando primero con la plantilla que identifica al área de color verde, y si la aguja no puede ser encontrada dentro de esta

sección, continuará su búsqueda en la plantilla del área que identifica al color rojo, se realiza la búsqueda primeramente en estas dos secciones porque gracias a sus colores oscuros en la imagen original, resaltan de mejor manera una vez aplicado el algoritmo Canny y la binarización.

De la operación clip, se obtiene como resultado una nueva imagen binaria, en la cual los píxeles de color blanco indicarían el área que conforma a la sección, mientras que si llegarán a existir píxeles en color negro, indicarán que hay un objeto dentro de dicha sección, siendo este posible objeto, la aguja, para corroborar esto, la imagen resultante de la operación clip, es binarizada inversamente y aplicado un etiquetado de componentes conexos para validar si dentro de la imagen existe más de un objeto, y si un objeto cuenta con un área en píxeles mayor a 1500, se indicaría que ha sido identificada la aguja, finalizando así, el procesamiento de la imagen. En dado caso de que no haya sido encontrada la aguja dentro de esta sección, se realizaría nuevamente una comprobación de objetos, pero usando la siguiente plantilla (sección color roja), si la aguja no ha sido identificada dentro de la sección de color rojo o verde, por default, se indicará que la aguja se encuentra dentro de la sección de color amarilla.

Finalmente, es enviado al cliente el resultado obtenido del procesamiento de la fotografía, los resultados a mostrar en la aplicación web serán, la fotografía procesada, resaltando en ella a la aguja identificada, dibujando un rectángulo con bordes de color verde alrededor de ella, así como también acompañando a esta imagen, un texto que indique en que sección de color del manómetro fue identificada la aguja. A su vez, la aplicación web enviará este resultado obtenido a Google Drive, específicamente a una hoja de Google Sheets, siendo esta hoja el historial de las lecturas realizadas, los datos enviados como también el contenido de la hoja de datos es: fecha y hora de la lectura realizada, el valor indicador de la sección en donde fue identificada la aguja (un valor 3 para la sección color verde, 2 para la sección amarilla y el valor 1 para la sección roja), y por último, la url que permita consultar la fotografía antes de ser procesada. Finalizando así la lectura del manómetro.

Una representación gráfica del procesamiento que es realizado a cada fotografía capturada del manómetro, junto con lo anteriormente descrito, puede ser consultado en la Figura 5.

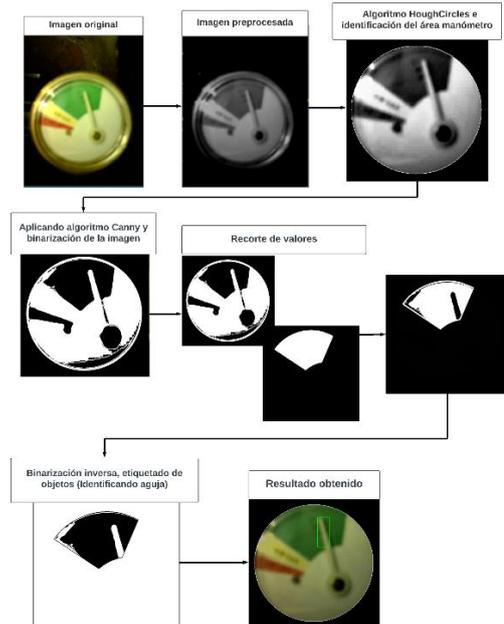


Figura 5. Procesamiento de la imagen para la identificación de la aguja.

4) *Consulta del historial de lecturas:* Para la consulta del historial de las lecturas realizadas, el dispositivo debe estar inicializado en modo monitoreo, es decir que el ESP32-CAM debe estar configurado como un servidor web. El cliente al establecer una comunicación con el ESP32-CAM, del menú de opciones en su página web de inicio del ESP32-CAM, deberá seleccionar la opción para la consulta de lecturas del manómetro, al hacer esto, el ESP32-CAM redirige al cliente hacia la aplicación web del servidor PythonAnywhere, la aplicación web solicita primeramente a Google Drive el contenido de la hoja de Google Sheets (esto por medio de la credencial firmada para la API de Google Drive), una vez obtenidos estos datos, la aplicación web retorna esta información al cliente, en forma de una lista ordenada desde la fecha más reciente hasta la más antigua, el historial de las lecturas registradas, solicitando al cliente que seleccione la que desea consultar. Cuando el cliente selecciona una del total de opciones, la aplicación web, crea una página web cuyo contenido sea: la fecha y hora en que fue realizada la lectura, la fotografía capturada y un texto que indique en que sección de color del manómetro fue identificada la aguja en el procesamiento de la fotografía.

Resultados

Mediante la integración de los materiales y componentes electrónicos mencionados anteriormente (consultar Tabla 1) en la construcción del circuito (ver Figura 2) y el dispositivo electrónico, se logró obtener como resultado lo presentado en la Figura 6.

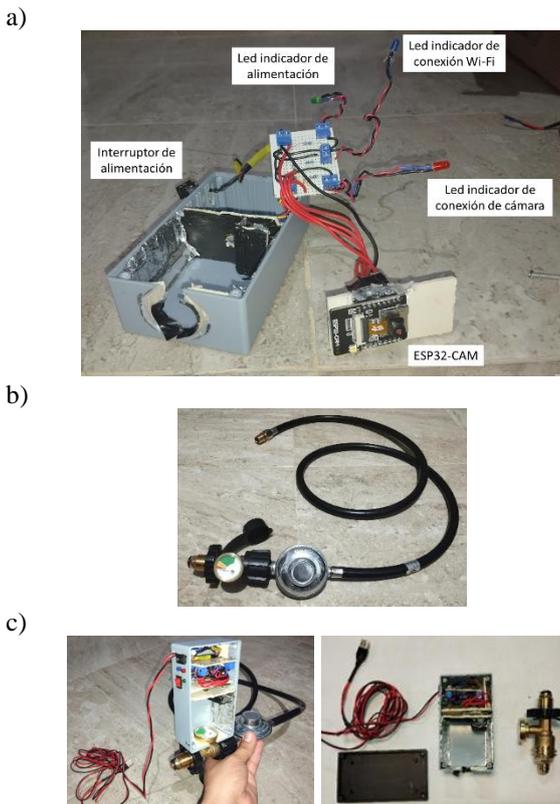


Figura 6. Construcción del dispositivo electrónico propuesto. a) Interior del dispositivo electrónico propuesto, b) Manómetro utilizado, c) Unificación de materiales.

El dispositivo construido se acopló a un tanque casero de Gas LP de 20 kg para la captura de las mediciones proporcionadas por el manómetro analógico. Esto por medio de la toma de fotografías a

través del ESP32-CAM incorporado dentro del dispositivo (ver Figura 7).

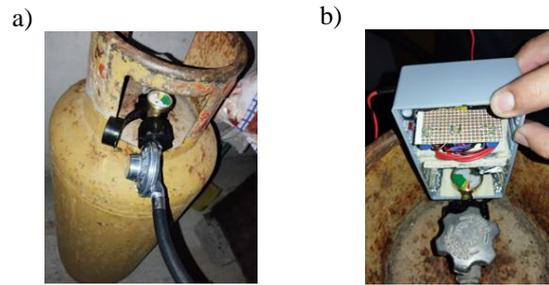


Figura 7. Acoplamiento del dispositivo propuesto, a) Colocación de manómetro al tanque de gas, b) Colocación del dispositivo propuesto al manómetro.

Una vez acoplado y encendido el dispositivo, fue realizada la comprobación de su correcto funcionamiento, para ello, se establece una comunicación con el ESP32-CAM (en modo monitoreo), y este último, retorna al cliente una página web en la cual, el cliente debe seleccionar una de las dos opciones mostradas en pantalla, siendo estas, la captura o la consulta de lecturas del manómetro, en este caso, es seleccionada la opción de “Tomar lectura del manómetro”. Después de haber realizado correctamente la captura de la fotografía al manómetro, el redireccionamiento del cliente hacia la aplicación web (alojada en el servidor PythonAnywhere), y el procesamiento de la fotografía por la aplicación, es mostrado al cliente el resultado obtenido (ver Figura 8).

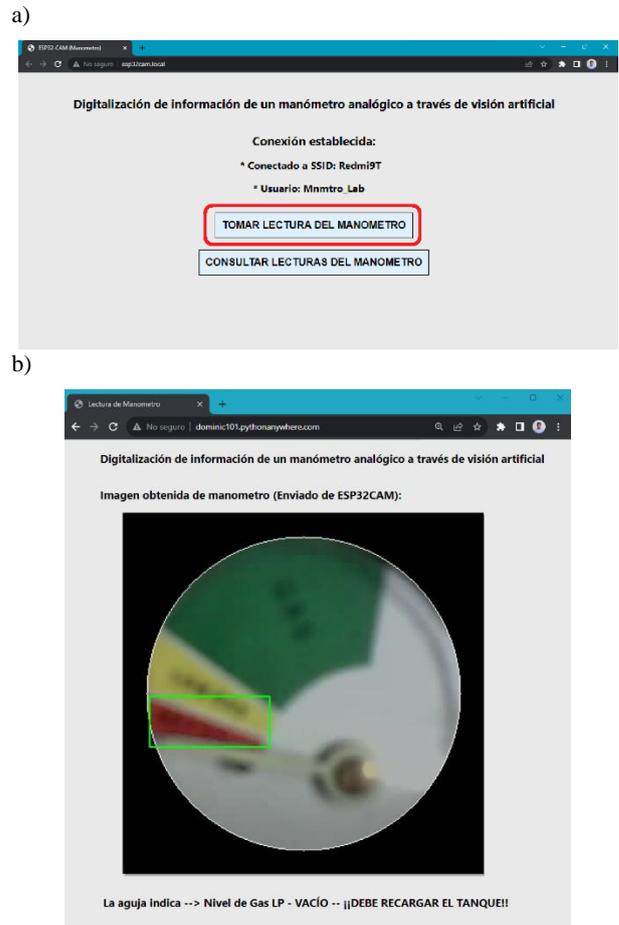


Figura 8. Lectura del manómetro a) Menú de opciones de la página web del ESP32-CAM, b) Resultado obtenido por la aplicación web alojada en el servidor PythonAnywhere.

Al comprobar que el dispositivo funcionaba correctamente, se decidió dejarlo operando. Fueron realizadas durante dos semanas ininterrumpidas, lecturas del manómetro, específicamente cuatro lecturas por día, dos por la mañana y dos por la tarde-noche, obteniendo de ello, un registro de 56 lecturas, es decir, 56 fotografías y 56 valores numéricos obtenidos de cada fotografía procesada por la aplicación web. Para obtener una representación visual y de fácil comprensión acerca de este monitoreo, se realizó la graficación de estos datos (ver Figura 9).

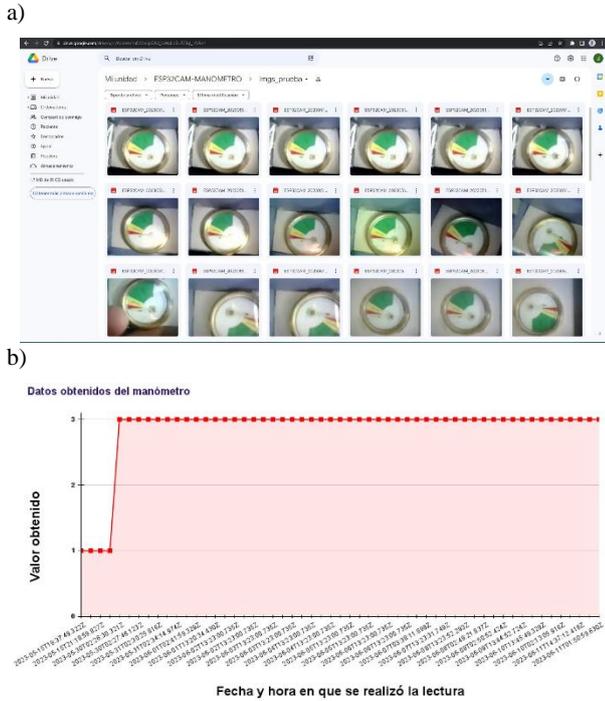


Figura 9. Resultados obtenidos de lecturas realizadas, a) Fotografías obtenidas por el ESP32-CAM y almacenadas en Google Drive, b) Gráfica de las lecturas obtenidas durante dos semanas, los primeros cuatro registros de valor 1, indica las pruebas del dispositivo con la válvula del tanque de Gas LP cerrada, mientras que los siguientes registros de valor 3, son los obtenidos una vez abierta la válvula del tanque de Gas LP y dejando operando al dispositivo propuesto.

Una comparativa entre dos lecturas realizadas puede ser apreciada en la Figura 10, siendo la primera (Figura 10 inciso a) la lectura realizada al manómetro cuando el tanque de gas LP se encontraba lleno, y la segunda (Figura 10 inciso b), la última lectura realizada al manómetro después de haber transcurrido dos semanas, comprobando así que el sistema logró identificar adecuadamente los cambios de posición de la aguja.

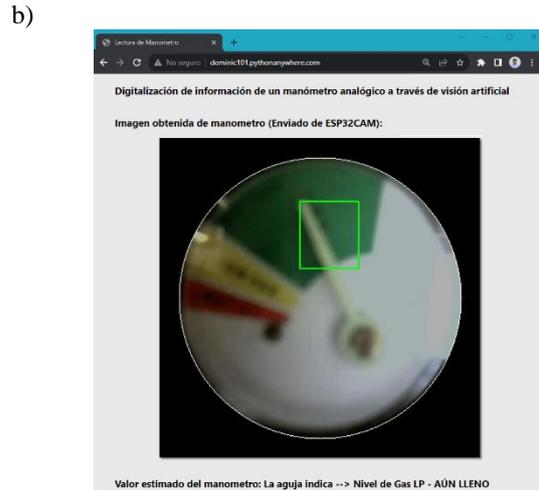
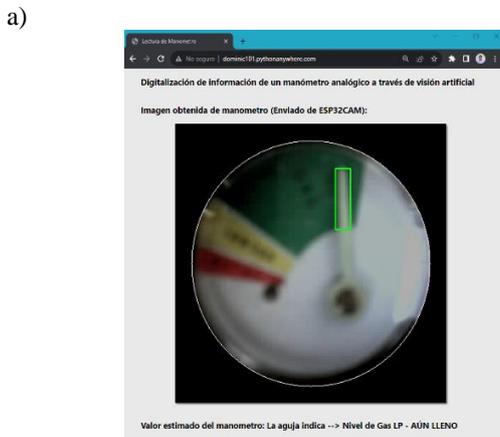


Figura 10. Resultados del procesamiento de las fotografías capturadas al manómetro, a) Lectura obtenida del manómetro en un tanque de gas LP lleno, b) Última lectura obtenida del manómetro después de haber transcurrido dos semanas.

Si bien las lecturas realizadas durante dos semanas no mostraron cambios representativos en los niveles de gas en el tanque, se decidió esperar a que transcurrieran otras dos semanas más a partir de la última lectura realizada, para realizar nuevamente más lecturas y con ello comprobar si el contenido de gas del tanque ha disminuido.

Una vez transcurridas las dos semanas en las cuales no fue realizada ninguna lectura del manómetro, se realizó una nueva, obteniendo de ella, una fotografía en la cual el manómetro indica que el nivel del tanque de gas se encuentra bajo (ver Figura 11, inciso a), se continuaron realizando nuevamente dos lecturas por día, una por la mañana y otra por la tarde-noche. A partir del tercer día, una lectura realizada al manómetro indicaba que el tanque de gas se encontraba vacío (ver Figura 11, inciso b), por lo tanto, correspondía su llenado o bien el cambio del tanque de gas y nuevamente acoplar el dispositivo. Al tener presente que fueron obtenidas estas variaciones en los niveles de gas indicados por el manómetro, fue realizada nuevamente la graficación de estos datos (ver Figura 12), para observar este nuevo comportamiento.

