

DISEÑO DE UN SISTEMA DIGITAL PARA LA LECTURA DE CONTADORES DE  
AGUA POR MEDIO DE CÁMARAS E IA.

KATHERINE PAOLA PARRA TARAZONA  
JULIO ANDRÉS SÁNCHEZ ABELLO

UNIVERSIDAD INDUSTRIAL DE SANTANDER  
FACULTAD DE INGENIERÍAS FISICOMECÁNICAS  
ESCUELA DE INGENIERÍAS  
ELÉCTRICA, ELECTRÓNICA Y DE TELECOMUNICACIONES  
BUCARAMANGA

2023

DISEÑO DE UN SISTEMA DIGITAL PARA LA LECTURA DE CONTADORES DE  
AGUA POR MEDIO DE CÁMARAS E IA.

KATHERINE PAOLA PARRA TARAZONA  
JULIO ANDRÉS SÁNCHEZ ABELLO

Trabajo de Grado para optar al título de  
Ingeniero Electrónico

Director

Jaime Guillermo Barreo Pérez

Magíster en potencia eléctrica

UNIVERSIDAD INDUSTRIAL DE SANTANDER  
FACULTAD DE INGENIERÍAS FISICOMECAICAS  
ESCUELA DE INGENIERÍAS  
ELÉCTRICA, ELECTRÓNICA Y DE TELECOMUNICACIONES  
BUCARAMANGA

2023

## **DEDICATORIA**

A Dios por brindarme fortaleza, sabiduría y paciencia para poder culminar satisfactoriamente esta etapa y meta en mi vida.

A mi madre Jenny J. Abello y mi abuela Ligia Noa (Q.E.P.D) por su paciencia, comprensión y apoyo en los momentos más difíciles, su amor incondicional, sus palabras de aliento y tranquilidad en los momentos justos.

A mis hermanos, mis tías y familiares por apoyarme durante este proceso universitario en los momentos difíciles y por siempre estar pendientes de mis necesidades.

A mi amigo incondicional Oscar Ayala, por acompañarme en este recorrido y ser mi apoyo emocional y académico, por sus consejos y lecciones de vida que siempre llevare presentes en cada meta que me proponga, igual a María F. Guevara, Luisa Guevara.

Mi compañera de trabajo de grado Katherine P. Parra Tarazona, por su apoyo, conocimientos brindados y paciencia durante este proceso.

A mis amigos gracias por su compañía y apoyo, agradecido con la vida por haber tenido la dicha de poderlos conocer y compartir momentos que estarán siempre en mi memoria.

**JULIO ANDRÉS SÁNCHEZ ABELLO**

A Dios por darme la fuerza y sabiduría para culminar esta etapa llena de aprendizajes.

A mis padres Herman Parra y Otilia Tarazona, por ser un apoyo incondicional a lo largo de este proceso, por su paciencia, comprensión y palabras de aliento que me han impulsado a dar lo mejor de mí en cada momento, gracias por confiar y creer en mí, este logro también es suyo.

A mis hermanos, quienes siempre estuvieron ahí para ayudarme y animarme a seguir adelante en los peores momentos, gracias por su apoyo incondicional y confiar en mis capacidades, ustedes han sido parte fundamental de mi crecimiento personal y académico.

A mis amigos que me brindaron su compañía y me animaron a lo largo de este camino, a todos ellos les deseo muchos éxitos.

**KATHERINE PARRA TARAZONA**

## **AGRADECIMIENTOS**

A Dios por brindarnos la oportunidad de crecer profesionalmente, guiándonos y ayudándonos en cada desafío que enfrentamos durante nuestro recorrido universitario, agradezco por concedernos salud y entendimiento para completar esta etapa de nuestras vidas de manera satisfactoria.

A nuestras familias por apoyarnos a través de las dificultades, por compartir nuestras victorias y derrotas, por creer siempre en nosotros y ser la voz de aliento que nos levantaba en los momentos más difíciles.

Al Msc. Jaime Guillermo Barrero Pérez, director del proyecto, por brindarnos la experiencia, la dedicación, la paciencia y por orientarnos en el desarrollo del trabajo de grado.

A nuestros amigos, por motivarnos con sus palabras de fuerza que en algún momento necesitamos y por ser parte de los momentos memorables.

A nuestros profesores a lo largo de la carrera, por ser parte de este proceso y la formación tanto personal al igual que la profesional, por sus conocimientos y experiencias compartidas.

**Katherine Parra Tarazona y Julio A. Sánchez Abello**

## CONTENIDO

	pág.
<b>INTRODUCCIÓN</b>	<b>14</b>
<b>1. OBJETIVOS</b>	<b>16</b>
<b>2. ESTADO DEL ARTE</b>	<b>17</b>
2.1. INTELIGENCIA ARTIFICIAL.	18
2.1.1. Redes neurales convolucionales (CNN).	19
2.2. APRENDIZAJE AUTOMÁTICO.	20
2.2.1. Aprendizaje automático supervisado.	20
2.2.2. Aprendizaje automático no supervisado.	21
2.2.3. Aprendizaje profundo.	21
2.3. TRANSFERENCIA DE APRENDIZAJE.	22
2.4. PERCEPTRÓN SIMPLE.	22
2.4.1. Perceptrón multicapa (MLP).	24
2.4.2. Retropropagación.	25
2.5. SISTEMAS EMBEBIDOS.	26
2.6. MICROCONTROLADOR.	27
2.6.1. Cámara.	28
2.6.2. Batería.	30
2.6.3. Protocolo de comunicación inalámbrica.	31
2.6.4. Protocolo de transferencia de datos.	32
2.7. PROGRAMA DE DISEÑO.	33
2.7.1. SolidWorks.	33
2.8. IMPRESIÓN 3D.	33

2.8.1. Tipos de Materiales Plásticos para Impresión 3D.	34
<b>3. NORMATIVA DEL CONSUMO DE AGUA POTABLE</b>	<b>35</b>
3.1. TARIFAS DE CONSUMO DE AGUA POTABLE.	36
3.1.1. Consumo básico.	36
3.1.2. Consumo complementario y suntuario.	37
3.1.3. Cálculo del servicio de acueducto.	38
3.2. NORMA TÉCNICA PARA MEDIDORES DE AGUA.	39
3.3. CONTADORES DE AGUA.	40
3.3.1. Tipos de contadores de agua.	40
3.3.2. Contador de agua modelo R200.	42
3.3.3. Lectura de contador de agua.	43
<b>4. PROTOCOLO DE TRANSMISIÓN DE DATOS</b>	<b>45</b>
4.1. TRANSFERENCIA DE DATOS AL SERVIDOR LOCAL.	46
<b>5. BASE DE DATOS.</b>	<b>50</b>
5.1. BASE DE DATOS DE SEGMENTACIÓN.	51
5.2. BASE DE DATOS DE DETECCIÓN.	51
5.2.1. Base de datos de detección para modelo YOLOv8.	52
5.2.2. Base de datos de detección para modelo Detectron2.	53
5.3. ESTRUCTURA EN 3D.	54
5.3.1. Cuerpo principal.	54
5.3.2. Tapa superior.	56
5.3.3. Cuerpo secundario.	56
<b>6. IMPLEMENTACIÓN DE LAS REDES NEURONALES</b>	<b>57</b>
6.1. MODELO DE SEGMENTACIÓN.	58
6.1.1. Entrenamiento red YOLOv8.	58

6.2. MODELOS DE DETECCCIÓN.	60
6.2.1. Entrenamiento <i>Detectron2</i> .	61
6.2.2. Entrenamiento <i>YOLOv8</i> .	62
<b>7. INTERFAZ GRÁFICA</b>	<b>63</b>
<b>8. RESULTADOS</b>	<b>67</b>
8.1. DESEMPEÑO RED NEURONAL <i>YOLOv8m-seg</i> SEGMENTACIÓN.	67
8.2. COMPARATIVA ENTRE <i>YOLOv8</i> Y <i>DETECTRON2</i> EN DETECCIÓN.	69
<b>9. RECOMENDACIONES</b>	<b>71</b>
<b>10. CONCLUSIONES</b>	<b>72</b>
<b>BIBLIOGRAFÍA</b>	<b>74</b>
<b>ANEXOS</b>	<b>80</b>

## LISTA DE FIGURAS

	<b>pág.</b>
Figura 1. Sistema global de proceso de una red neuronal.	19
Figura 2. Arquitectura y función de transferencia de un perceptrón simple.	23
Figura 3. Arquitectura y función de activación de un perceptrón multicapa.	25
Figura 4. Datos de facturación para el servicio de acueducto.	39
Figura 5. Despiece del contador de agua modelo R200.	41
Figura 6. Lectura dial del contador de agua.	44
Figura 7. Etiquetado para detección con <i>Roboflow</i> .	53
Figura 8. Estructura principal.	55
Figura 9. Estructura secundaria.	56
Figura 10. Procesamiento de las imágenes guardadas en el servidor local.	64
Figura 11. Interfaz gráfica de consumo Diario y Semanal.	65
Figura 12. Interfaz gráfica de consumo Mensual y Anual.	65
Figura 13. Interfaz gráfica Visualización y Ref contadores.	66
Figura 14. Proceso de segmentación.	68
Figura 15. Gráficas de la función de pérdida modelo <i>YOLOv8</i> para segmentación.	68
Figura 16. Proceso de detección.	70
Figura 17. Gráficas de la función de pérdida modelo <i>YOLOv8</i> para detección.	70

## LISTA DE TABLAS

	<b>pág.</b>
Tabla 1. Especificaciones Técnicas <i>ESP32-Cam</i> .	28
Tabla 2. Modelos de Cámaras Compatibles con la Placa <i>ESP32-Cam</i> .	29
Tabla 3. Tipos de baterías.	30
Tabla 4. Especificaciones técnicas <i>WiFi 802.11n</i> y <i>Bluetooth 4.2</i>	31
Tabla 5. Tarifa de servicio público de agua potable para uso residencial.	37
Tabla 6. Tarifa de servicio público de agua potable para uso no residencial.	37
Tabla 7. Caudales e Incertidumbre para contador modelo <i>R200</i> , según <i>NTC ISO 4064-1 : v2017</i> .	43
Tabla 8. Base de datos inicial para segmentación.	51
Tabla 9. Base de datos aumentada para segmentación <i>YOLOv8</i> .	52
Tabla 10. Base de datos inicial para detección	52
Tabla 11. Base de datos aumentada para detección con <i>YOLOv8</i> y <i>Detectron2</i> .	53
Tabla 12. Hiperparámetros para entrenamiento en segmentación modelo <i>YOLOv8m</i> .	60
Tabla 13. Hiperparámetros para entrenamiento en detección modelo <i>Detectron2</i> .	62
Tabla 14. Hiperparámetros para entrenamiento en detección modelo <i>YOLOv8</i> .	62
Tabla 15. Resultados de segmentación <i>YOLOv8</i> .	67
Tabla 16. Comparación entre <i>YOLOv8</i> y <i>Detectron2</i> .	69

## LISTA DE ANEXOS

	<b>pág.</b>
Anexo A. Norma Técnica Interna de Medidores de Agua.	80
Anexo B. Especificaciones Técnicas y de Calibración de Contador de Agua.	81
Anexo C. Planos de la Estructura de Fácil Acople.	82
Anexo D. Bases de datos.	83
Anexo E. Repositorio de GitHub.	84

## RESUMEN

**TÍTULO:** DISEÑO DE UN SISTEMA DIGITAL PARA LA LECTURA DE CONTADORES DE AGUA POR MEDIO DE CÁMARAS E IA. \*

**AUTOR:** KATHERINE PAOLA PARRA TARAZONA, JULIO ANDRÉS SÁNCHEZ ABELLO \*\*

**PALABRAS CLAVE:** INTELIGENCIA ARTIFICIAL, IMÁGENES, CONSUMO, AUTOMATIZACIÓN, IoT, PROCESOS, INALÁMBRICA, MEDIDORES, DIGITALIZACIÓN, CÁMARAS, PYTHON.

### DESCRIPCIÓN:

La toma de datos de los medidores es realizada manualmente por las empresas prestadoras de servicios, que van casa por casa registrando el consumo mensual de las mismas, lo que implica que el usuario no pueda llevar un seguimiento detallado de su consumo, supone errores humanos, baja eficiencia, baja efectividad, susceptibilidad a factores ambientales y costos adicionales por la mano de obra requerida. Esto lleva a la necesidad de crear sistemas y/o elementos que automaticen o digitalicen el proceso de recolección de datos en los medidores y faciliten al usuario el seguimiento de su consumo a lo largo del mes.

La inclusión de instrumentos inteligentes debe ser clave, ya que brindan al usuario la posibilidad de llevar un registro detallado del uso del servicio para así poder identificar malos hábitos de consumo, fugas en las tuberías, posible deterioro de estas, desgaste de los sellos o malos empalmes que generan consumos fantasmas, para así tomar las acciones necesarias en pro de evitar sobrecostos generados. En este proyecto se implementa un módulo o estructura de fácil acople para medidores de agua convencionales, por medio del cual se capturarán imágenes que serán enviadas inalámbricamente para su posterior análisis y digitalización por medio de IA.

---

\* Trabajo de grado

\*\* Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingenierías Eléctrica, Electrónica y telecomunicaciones. Director: Jaime Guillermo Barreo Pérez, Magíster en potencia eléctrica.

## ABSTRACT

**TITLE:** DESIGN OF A DIGITAL SYSTEM FOR THE READING OF WATER METERS BY MEANS OF CAMERAS AN AI. \*

**AUTHOR:** KATHERINE PAOLA PARRA TARAZONA, JULIO ANDRES SANCHEZ ABELLO \*\*

**KEYWORDS:** ARTIFICIAL INTELLIGENCE, IMAGES, CONSUMPTION, AUTOMATION, IoT, PROCESS, WIRELESS, METERS, DIGITIZATION, CAMERAS, PYTHON.

### **DESCRIPTION:**

Leaks in pipes, worn joints and poor connections are some of the main causes of water loss around the world <sup>1</sup>. With the aim of promoting a more conscious and sustainable behavior in the use of water, the project focuses on providing users with detailed information on their daily and monthly consumption through consumption curves and cumulative records. To achieve this, new technologies will be used in the billing and data collection processes, such as cameras that can obtain images of the meters and apply methods of digital image processing and pattern recognition through artificial intelligence. In this way, it seeks to raise awareness about the importance of conserving water and provide users with tools to monitor their consumption and detect possible leaks or bad consumption habits.

---

\* Bachelor Thesis

\*\* Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingenierías Eléctrica, Electrónica y telecomunicaciones. Director: Jaime Guillermo Barreo Pérez, Magíster en potencia eléctrica.

<sup>1</sup> Kiat Siong Ng, Pei-Yin Chen y Yuan-Chi Tseng. “A design of automatic water leak detection device”. En: *2017 IEEE 2nd International Conference on Opto-Electronic Information Processing (ICOIP)*. 2017 IEEE 2nd International Conference on Opto-Electronic Information Processing (ICOIP). Jul. de 2017, págs. 70-73. DOI: 10.1109/OPTIP.2017.8030701.

## INTRODUCCIÓN

Actualmente la lectura de los medidores se realiza de forma manual por las empresas prestadoras de servicios, que asignan un trabajador para registrar el consumo mensual de cada domicilio, este proceso puede conllevar a errores humanos, baja eficiencia, baja efectividad, susceptibilidad a factores ambientales y costos adicionales por la mano de obra requerida, además al tratarse de un proceso que solo se realiza una vez al mes, impide que el usuario pueda llevar un registro detallado del uso del servicio e identificar malos hábitos de consumo o posibles fugas.

El avance tecnológico generado en la última década en las telecomunicaciones y en los sistemas inalámbricos nos ofrecen una alternativa para la automatización de este tipo de procesos, el uso de nuevas tecnologías nos permiten reducir costos en la mano de obra, llevar un mejor control, mejorar la comunicación entre las empresas prestadoras de servicio y el usuario, minimizando errores humanos que ocurren a la hora de realizar la toma de datos por las empresas prestadoras de servicio teniendo en cuenta que la tecnología es más asertiva y facilita la exactitud en la toma de datos. Los planes de monitorización de consumo se encaminan a la toma de decisiones de forma oportuna, mejorando así el servicio y optimizando los costos de facturación mitigando el impacto por consumos fantasmas.

En este proyecto se presenta una solución a esta problemática, la cual consiste en construir un sistema embebido que permita a través del procesamiento de imágenes obtener la lectura de contadores de agua, para lograr esto se diseñó un módulo o estructura de fácil acople impresa en *3D* para medidores de agua modelo *R200*. Este modelo de contador cumple con las normativas de la *AMB S.A E.S.P.*, el sistema cuenta con una *ESP32 - Cam* que se encarga de capturar las imágenes que son enviadas inalámbricamente a un sistema de procesamiento de datos para su posterior análisis y digitalización por medio de *IA*.

Adicionalmente se busca monitorear el consumo de agua de una residencia o comercio por medio de una interfaz gráfica intuitiva y amigable para el usuario, haciendo uso de herramientas tecnológicas que faciliten la consulta de datos aproximados sobre el consumo diario, semanal, mensual y anual, esto proporcionará a los usuarios la información necesaria para controlar su consumo e identificar posibles malos hábitos de consumo, fugas en las tuberías, desgaste en los sellos o empalmes deficientes, así, podrán tomar acciones oportunas para evitar desperdicios de agua.

## 1. OBJETIVOS

### Objetivo general

- Diseñar un sistema que permita a través del procesamiento de imágenes obtener la lectura de contadores de agua, en un entorno residencial y/o comercial por medio de cámaras e IA.

### Objetivos específicos

- Diseñar una estructura de fácil acople para contadores de agua modelo *R200* que permita la correcta ubicación del sistema embebido y su cámara para la toma de datos.
- Crear una base de datos con imágenes de contadores de agua modelo *R200* para entrenar una Red Neuronal.
- Seleccionar una arquitectura apropiada para la implementación en el controlador, que nos permita reconocer correctamente los dígitos del contador de agua modelo *R200*.
- Determinar el protocolo de conexión inalámbrica apropiado para el envío de la información recolectada para su posterior análisis e interpretación.

## 2. ESTADO DEL ARTE

El agua es fundamental para el sostenimiento de la vida, sin embargo, tener acceso a ella es un privilegio que nunca debe tomarse a la ligera, pues este preciado líquido es escaso. Según el artículo “*¿Cuánta agua potable hay en la Tierra?*” escrito por *AQUA FUNDACIÓN* nos indica que, aunque la tierra está compuesta en su mayor parte por agua, sólo un 2,5 % es considerada agua dulce y sólo el 0.007 % es potable y está disponible para beber <sup>2</sup>, esta cantidad se reduce cada vez más por causa de la contaminación generada por el ser humano. Con una población en crecimiento y una baja proporción de agua potable, es necesario conservar este preciado líquido, ahorrar agua no solo depende de científicos, hidrólogos o ambientalistas, por el contrario, la tarea de cada uno de nosotros debe ser preservarla adoptando buenos hábitos de consumo en la vida cotidiana.

Una de las causas que impiden llevar un mejor control de los hábitos de consumo y fugas de agua, es la ausencia de sistemas automatizados para la medición del servicio de agua en las residencias y/o comercios, ya que las empresas prestadoras del servicio no cuentan con sistemas automatizados que permitan al usuario tener acceso a sus registros de consumo durante el periodo de facturación, esto trae como consecuencia que los usuarios repitan malas prácticas para el uso del agua o que no identifiquen con prontitud fugas ya que no cuentan con los recursos necesarios para identificar estos malos hábitos de consumo.

---

<sup>2</sup> AQUAE FUNDACIÓN. *¿Cuánta agua potable hay en la Tierra?* URL: <https://www.fundacionaquae.org/wiki/cantidad-de-agua-potable-fuente-de-vida/> (visitado 04-02-2023).

## 2.1. INTELIGENCIA ARTIFICIAL.

La inteligencia artificial es la base de todo el aprendizaje automático y el futuro de todos los procesos complejos de toma de decisiones, imita el proceso de la inteligencia humana al crear y aplicar algoritmos en un entorno informático dinámico<sup>3</sup>. En la actualidad, la inteligencia artificial se emplea para realizar tareas que suelen ser repetitivas y que tradicionalmente se han llevado a cabo de forma manual, tales como la interacción con el cliente a través de aplicativos, el reconocimiento de patrones y la resolución de problemas básicos repetitivos. De esta manera, se logra una mayor eficacia y eficiencia en dichas tareas y mejora la satisfacción del cliente en general<sup>4</sup>.

Las *RNA (Redes Neuronales Artificiales)* son un tipo de modelo computacional que se inspira en el funcionamiento del cerebro humano y su sistema nervioso. Estas redes están compuestas por un conjunto de unidades de procesamiento llamadas neuronales artificiales, que se organizan en capas y se interconectan mediante conexiones sinápticas artificiales como se muestra en la Figura 1, Las principales características de las *RNA* son: capacidad de aprendizaje y adaptación a partir de ejemplos, capacidad de generalización para realizar predicciones sobre datos no vistos previamente, su capacidad de procesamiento paralelo y distribuido. Las funciones principales de las *RNA* son la clasificación, regresión, predicción y optimización<sup>5</sup>.

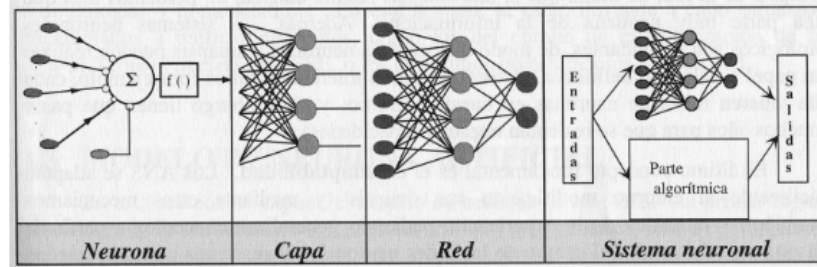
---

<sup>3</sup> NetApp. *¿Qué es la IA y por qué es importante?* | NetApp. Section: AI. URL: <https://www.netapp.com/es/artificial-intelligence/what-is-artificial-intelligence/> (visitado 04-02-2023).

<sup>4</sup> Oracle Colombia. *¿Qué es la inteligencia artificial (IA)?* URL: <https://www.oracle.com/co/artificial-intelligence/what-is-ai/> (visitado 04-02-2023).

<sup>5</sup> Abdelmalik Moujahid Pedro Larrañaga Iñaki Inza. *Tema 8. Redes Neuronales*. <http://www.sc.ehu.es/ccwbayes/docencia/mmcc/docs/t8neuronales.pdf>. (Accessed on 11/02/2023).

Figura 1. Sistema global de proceso de una red neuronal.



**2.1.1. Redes neurales convolucionales (CNN).** La Red Neuronal Convolucional (*CNN*) son un tipo de red neuronal artificial diseñada para procesar datos con estructura en forma de malla, como imágenes, vídeos o señales de audio. Estas redes utilizan las convoluciones para obtener características destacadas de los datos de entrada, de esta forma aprenden a identificar patrones en ellos. Sus capas se componen de un conjunto de filtros o *kernels* aplicados repetidamente a la imagen de entrada obteniendo así un mapa de características. A estos mapas se le aplican capas de submuestreo o capas de *pooling*, que reducen su tamaño para disminuir la complejidad computacional de la red, las *CNN* son capaces de aprender y reconocer estructuras o modelos cada vez más complejos y abstractos que pueden empezar con datos de píxeles sin procesar hasta aprender características muy específicas, como bordes, patrones, texturas, formas básicas y complejas.<sup>6</sup>

Las *CNN* emplea múltiples copias idénticas de la misma red, donde cada capa se especializa en una tarea específica. Posteriormente, la siguiente capa aprovecha el resultado de esta tarea para resolver problemas complejos. Esta característica ofrece la ventaja de permitir que una neurona en la red aprenda una vez y sea utilizada en diferentes partes lo cual simplifica el proceso de aprendizaje del modelo y disminuye errores. En comparación con otros algoritmos, las redes neuronales convolucionales requieren un preprocesamiento considerablemente

<sup>6</sup> Dan C Ciresan et al. *Flexible, High Performance Convolutional Neural Networks for Image Classification*. <https://people.idsia.ch/~juergen/ijcai2011.pdf>. (Accessed on 03/05/2023).

inferior, se componen de diversas capas, siendo la capa convolucional la más destacada, a la cual se debe el nombre de la red<sup>7</sup>.

## **2.2. APRENDIZAJE AUTOMÁTICO.**

El Aprendizaje Automático o *Machine Learning (ML)* es una rama de la inteligencia artificial que se enfoca en el desarrollo de algoritmos y modelos que permiten a las computadoras aprender de forma autónoma a partir de datos y mejorar su rendimiento en diversas tareas sin requerir una programación específica para cada una de ellas. Estos algoritmos utilizan el análisis de grandes volúmenes de datos para identificar patrones y relaciones, lo que les permite utilizar ese conocimiento para tomar decisiones y realizar tareas complejas.

El Aprendizaje Automático tiene la capacidad de automatizar tareas complejas que anteriormente requerían intervención humana, lo que resulta en una reducción de costos y un aumento de la eficiencia. Además, estos algoritmos son adaptables a nuevos datos y situaciones, lo que les permite seguir mejorando su rendimiento a medida que se les presenta nueva información. Una de las fortalezas del Aprendizaje Automático es su capacidad para hacer predicciones precisas sobre el futuro, basándose en patrones y tendencias identificados en los datos<sup>8</sup>.

**2.2.1. Aprendizaje automático supervisado.** El aprendizaje supervisado es una técnica de Aprendizaje Automático en la que se entrena un modelo utilizando un conjunto de datos etiquetados. En otras palabras, el modelo se entrena para aprender a predecir una salida a partir de una entrada utilizando ejemplos de entrada y salida previamente etiquetados. Por

---

<sup>7</sup> DATA SCIENCE. *Redes Neuronales Convolucionales – la manera ELI5*. 2020. URL: <https://datascience.eu/es/vision-artificial/redes-neuronales-convolucionales/> (visitado 20-02-2023).

<sup>8</sup> Aurélien Géron. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*. 3.<sup>a</sup> ed. O'Reilly Media, Inc, 2022. 1196 págs.

ejemplo, si se quiere entrenar un modelo para clasificar imágenes de perros y gatos se proporcionará al modelo un conjunto de imágenes etiquetadas como perros o gatos, y el modelo aprendería a clasificar nuevas imágenes en una de esas dos categorías. El aprendizaje supervisado se utiliza normalmente en tareas de clasificación y regresión, donde se busca predecir una salida discreta o continua a partir de una entrada <sup>8</sup>.

**2.2.2. Aprendizaje automático no supervisado.** El aprendizaje automático no supervisado es una técnica de Aprendizaje Automático en la que se entrena un modelo utilizando un conjunto de datos no etiquetados. Es decir, el modelo se entrena para encontrar patrones y estructuras en los datos sin la necesidad de tener una salida deseada. El objetivo del aprendizaje no supervisado es descubrir información oculta en los datos y agruparlos en categorías o *clusters*. Algunos ejemplos de tareas de aprendizaje no supervisado son la detección de anomalías, la reducción de dimensionalidad y la segmentación de imágenes, el aprendizaje no supervisado no requiere la presencia de etiquetas o salidas deseadas, lo que lo hace útil en situaciones donde no se dispone de datos etiquetados o donde la tarea es desconocida<sup>8</sup>.

**2.2.3. Aprendizaje profundo.** El Aprendizaje Profundo (*Deep Learning*) es una técnica de Aprendizaje Automático que se basa en el uso de redes neuronales artificiales con múltiples capas para aprender y extraer características de los datos de entrada. A diferencia del aprendizaje supervisado tradicional donde las características se definen previamente, el Aprendizaje Profundo permite que el modelo aprenda automáticamente las características relevantes directamente de los datos de entrada.

Las redes neuronales profundas se componen de múltiples capas de neuronas interconectadas, cada una procesa y transforma la información de entrada para generar una salida, lo que mejora la capacidad del modelo para realizar tareas de manera más precisa.

El Aprendizaje Profundo se utiliza en una amplia variedad de aplicaciones como el recono-

cimiento de voz, la identificación de imágenes, el procesamiento del lenguaje natural, entre otros, debido a su capacidad para aprender patrones y características en grandes conjuntos de datos, destaca en tareas complejas y ha impulsado avances significativos en el campo de la inteligencia artificial<sup>8</sup>.

### **2.3. TRANSFERENCIA DE APRENDIZAJE.**

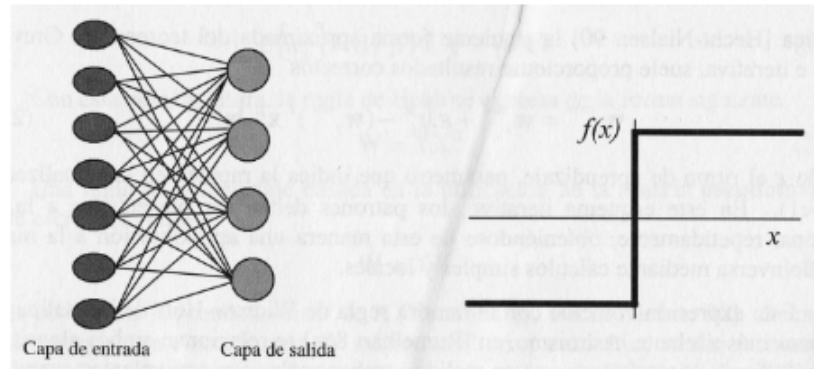
La Transferencia de Aprendizaje (*Transfer Learning*) es una técnica de *Machine Learning* que utiliza el conocimiento adquirido por un modelo en una tarea para mejorar el rendimiento en otra relacionada. En lugar de entrenar un modelo desde cero para cada tarea se aprovechan los conocimientos y habilidades previamente aprendidos para transferirse a una nueva actividad. Esta técnica es utilizada cuando se dispone de una cantidad limitada de datos para entrenar un modelo. La Transferencia de Aprendizaje se aplica en tareas como el reconocimiento de imágenes donde se usan modelos previamente entrenados, de esta forma se aprovechan los conocimientos aprendidos de características generales permitiendo una adaptación rápida y precisa a nuevas tareas<sup>8</sup>.

### **2.4. PERCEPTRÓN SIMPLE.**

El Perceptrón Simple es una arquitectura de red neuronal artificial que consta de una sola capa de neuronas de entrada, una capa de salida y una función de activación. El perceptrón simple se utiliza comúnmente para problemas de clasificación binaria. En esta arquitectura, cada nodo de entrada está conectado a un nodo de salida y cada conexión tiene un peso asociado que se utiliza para calcular la salida del nodo de salida. La función de activación del perceptrón simple es una función escalón que devuelve una salida binaria que indica si el perceptrón activa o no, como lo observamos en la figura 2 .

El aprendizaje del Perceptrón Simple se lleva a cabo mediante un proceso de entrenamiento supervisado. Durante el entrenamiento, se presentan a la red una serie de pares de vectores

Figura 2. Arquitectura y función de transferencia de un perceptrón simple.



de entrada y salida deseada. La red calcula su salida actual para cada vector de entrada y se compara con la salida deseada, si la salida actual no coincide con la salida deseada, se ajustan los pesos de las conexiones de entrada a la neurona de salida para reducir el error. Este ajuste se realiza mediante la regla de aprendizaje del Perceptrón, que actualiza los pesos de las conexiones en función del error cometido y del coeficiente de aprendizaje. El proceso de entrenamiento continúa hasta que la red converge y es capaz de producir la salida deseada para todos los vectores de entrada del conjunto de entrenamiento. Después del entrenamiento la red está lista para producir la salida adecuada para nuevas entradas que no se han visto antes<sup>9</sup>.

**Función de activación.** La función de activación del Perceptrón Simple es una función escalón que produce una salida binaria ( $1$  o  $0$ ) en función del valor de la suma ponderada de las entradas. El resultado de esta función se utiliza para determinar si el perceptrón activa o no, lo que significa si la salida del perceptrón es  $1$  o  $0$ , respectivamente.

En un perceptrón simple, la función de activación más común es la función escalón unitario

---

<sup>9</sup> Xabier Basogain Olabe. *REDES NEURONALES ARTIFICIALES Y SUS APLICACIONES*. Bilbao: Escuela Superior de Ingeniería de Bilbao, EHU.

(*step function*), que toma un valor de 1 si la entrada es mayor o igual que un cierto umbral (*threshold*) y 0 en caso contrario. Matemáticamente, la función de activación se define como:

$$\text{Si, } S_j > \theta; \text{ entonces } X_j = 1$$

$$\text{Si, } S_j \leq \theta; \text{ entonces } X_j = 0$$

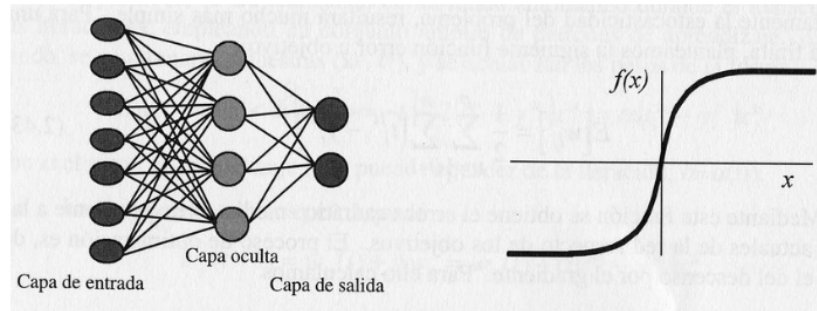
Donde  $x_j$  es la salida de la neurona  $j$ , y  $s_j$  es la suma ponderada de las entradas a la neurona  $j$  y el umbral se establece en  $\theta$ .

La función de activación de un perceptrón simple es importante porque es lo que le permite al perceptrón aprender a clasificar entradas en dos categorías diferentes. La regla de aprendizaje del perceptrón ajusta los pesos de las conexiones para minimizar el error en la tarea de clasificación y la función de activación determina cuál es la salida del perceptrón<sup>9</sup>.

**2.4.1. Perceptrón multicapa (MLP).** El Perceptrón Multicapa es una arquitectura de red neuronal artificial que consta de múltiples capas de neuronas incluyendo una capa de entrada, una o varias capas ocultas y una capa de salida. Cada neurona en una capa está conectada a todas las neuronas de la capa siguiente a través de conexiones ponderadas. La salida de la red se calcula mediante la propagación hacia adelante de las entradas a través de las capas ocultas hasta la capa de salida, donde se aplica una función de activación para producir la salida final. El Perceptrón Multicapa es capaz de resolver problemas no lineales y de clasificar patrones no linealmente separables lo que lo hace más poderoso que el Perceptrón simple. El entrenamiento del Perceptrón Multicapa se realiza mediante un proceso de retro propagación del error, que ajusta los pesos de las conexiones en función del error cometido en la salida de la red<sup>9</sup>, como lo indica la figura 3.

Se puede pensar en un perceptrón multicapa como una red que consta de innumerables neuronas artificiales en diferentes capas. Un perceptrón multicapa tiene una estructura similar a

Figura 3. Arquitectura y función de activación de un perceptrón multicapa.



un perceptrón simple, pero con más capas ocultas. A diferencia de un perceptrón simple, la función de activación ya no es lineal, sino que se muestran funciones de activación no lineales para su implementación<sup>10</sup>.

El proceso de aprendizaje de una red Perceptrón multicapa comienza desde la capa de entrada, las salidas de cada neurona se propagan, estas salidas pasan por las neuronas de las capas ocultas y finalmente se genera una salida en la capa de salida de la red. Luego se compara la salida generada por la red con la salida deseada, se calcula el error que será el parámetro para modificar los pesos sinápticos<sup>11</sup>.

**2.4.2. Retropropagación.** La retropropagación o *Backpropagation* es un algoritmo de entrenamiento supervisado utilizado en redes neuronales artificiales, especialmente en redes multicapa como el Perceptrón Multicapa. El objetivo del algoritmo es ajustar los pesos de las conexiones entre las neuronas de la red para minimizar el error entre la salida deseada y la sa-

---

<sup>10</sup> Ligdi González. *¿Qué es el Perceptrón? Perceptrón Simple y Multicapa*. Aprende IA. 5 de oct. de 2021. URL: <https://aprendeia.com/que-es-el-perceptron-simple-y-multicapa/> (visitado 05-10-2021).

<sup>11</sup> *Redes Neuronales Artificiales: El Perceptrón Multicapa*. Vrelectroniq. (Accessed on 05/16/2023). 21 de jul. de 2020. URL: <https://vrelectroniq.wixsite.com/vrelectroniq/post/redes-neuronales-artificiales-el-perceptrn-multicapa> (visitado 03-06-2023).

lida real de la red. El algoritmo utiliza una técnica de propagación hacia atrás del error donde el error en la salida de la red se propaga hacia atrás a través de las capas ocultas de la red y se utiliza para ajustar los pesos de las conexiones. El proceso de ajuste de los pesos se realiza iterativamente hasta que se alcanza un nivel de precisión deseado. La retropropagación es uno de los algoritmos de entrenamiento más populares y efectivos para redes neuronales artificiales y ha sido utilizado en una amplia variedad de aplicaciones, incluyendo reconocimiento de patrones, procesamiento de imágenes, procesamiento de voz y control de robots, entre otros <sup>9</sup>.

## **2.5. SISTEMAS EMBEBIDOS.**

Los sistemas embebidos se utilizan para realizar una tarea específica en un entorno dedicado y controlado, se componen principalmente por un microcontrolador dedicado capaz de ejecutar instrucciones a una determinada velocidad controlado por una señal de reloj. Estos sistemas se emplean en una amplia variedad de aplicaciones desde electrodomésticos hasta sistemas de control de tráfico aéreo y se programan para realizar tareas específicas de manera eficiente y confiable. Algunos sistemas embebidos se pueden apreciar en sistemas de control de temperatura en aires acondicionados, electrodomésticos, sistemas de control de frenos en automóviles, celulares, sistemas de control de iluminación en edificios, entre otros. Los sistemas embebidos son esenciales para el funcionamiento de muchos dispositivos y sistemas que utilizamos en nuestra vida diaria.

Según el libro "Todo sobre sistemas embebidos: arquitectura, programación y diseño de aplicaciones prácticas con el *PIC18F*" de Sergio Salas Arriarán, la arquitectura de un sistema embebido se compone de tres elementos principales:

**Microcontrolador.** Actúa como el cerebro del sistema embebido, se encarga de procesar la información y controlar los dispositivos periféricos. El microcontrolador incluye una unidad

central de procesamiento (*CPU*), memoria, puertos de entrada y salida entre otros componentes.

**Dispositivos periféricos.** Estos dispositivos se pueden conectar al microcontrolador para proporcionar una funcionalidad adicional, como sensores, actuadores, pantallas, teclados, entre otros.

**Fuente de alimentación.** Este componente proporciona la energía necesaria para el correcto funcionamiento del sistema embebido. La arquitectura de un sistema embebido puede variar según la aplicación y los requisitos específicos del proyecto. Sin embargo, en general los sistemas embebidos se diseñan para ser compactos eficientes en energía y de bajo costo, los modos de bajo consumo de corriente (*modo sleep*) son sumamente importantes y se utilizan en una amplia variedad de aplicaciones que involucran dispositivos con baterías, como electrodomésticos, celulares, dispositivos de control entre otros<sup>12</sup>.

## 2.6. MICROCONTROLADOR.

Un microcontrolador es un tipo de circuito integrado que combina un procesador central, memoria y periféricos de entrada/salida en un solo chip. Estos dispositivos se utilizan para controlar sistemas electrónicos en una amplia variedad de aplicaciones, desde electrodomésticos hasta sistemas de control industrial.

En el caso del *ESP32 - Cam*, se trata de un microcontrolador de 32 bits que combina un procesador de doble núcleo, conectividad *Wi-Fi* y *Bluetooth*, y una cámara *OV2640* en un solo dispositivo. Esto ofrece varias ventajas en comparación con otros microcontroladores y

---

<sup>12</sup> Daniel Alexis Pérez-Aguilar, Redy Henry Risco-Ramos y Luis Casaverde-Pacherrez. *Sistemas embebidos y su aportación a la industrias*. Oasys. Section: Industria 4.0. URL: <https://oasys-sw.com/sistemas-embebidos-industria/> (visitado 02-04-2023).

cámaras por separado, como: Ahorro de espacio al tener todo integrado en un solo dispositivo, facilidad de programación ya que el *ESP32 - Cam* admite una amplia gama de lenguajes de programación de alto nivel, conectividad inalámbrica *Wi-Fi* y *Bluetooth*, captura de imágenes y vídeos de alta calidad con la cámara integrada, en la tabla 1<sup>13</sup> se presentan algunas de las principales características de la *ESP32-Cam*<sup>14</sup>.

Tabla 1. Especificaciones Técnicas *ESP32-Cam*.

Características	ESP32 Cam
Voltaje de alimentación	5 [VDC]
Voltaje E/S	3.3 [VDC]
RAM	520 KB SRAM + 4M PSRAM
Bluetooth	4.2 BR/EDR y BLE standards
Wi-Fi	802.11 b/g/n/
Interfaz de Soporte	UART, SPI, I2C, PWM
Tarjeta TF	Máximo 4G
Puertos E/S	9
Formato de Imagen	JPEG, BMP, GRAYSCALE
Rango de Espectro	2412 2484 [MHz]
Consumo de Energía	-Flash Apagado: 180 [mA]@ 5 [V] -Flash encendido y brillo al máximo: 310 [mA] @ 5 [V] -Modo Deep-sleep : 6 [mA] @ 5 [V] (disipación de potencia más baja) -Modo Modern-sleep : Mínimo alcanzado 20mA @ 5 [V] -Modo Light-sleep : Mínimo alcanzado 6.7 [mA] @ 5 [V]
Temperatura de Funcionamiento	-20°C 85°C
Entorno de almacenamiento:	-40°C 90°C, <90 % Humedad relativa
Dimensiones	40.5 [mm] x 27 [mm] x 4.5 [mm], (±0.2) [mm]

**2.6.1. Cámara.** La *ESP32 - Cam* es una placa de desarrollo que combina el microcontrolador *ESP32* con una cámara integrada. Esta placa admite dos tipos de cámara, que se detallan

<sup>13</sup> *ESP32-CAM Product Specification*. 2023. URL: <https://loboris.eu/ESP32/ESP32-CAM%20Product%20Specification.pdf> (visitado 03-04-2023).

<sup>14</sup> *ESP32 CAM introducción y primeros pasos*. Section: ESP32. 18 de ene. de 2022. URL: <https://programarfacil.com/esp32/esp32-cam/#:~:text=ESP32%2DCAM%2C%20es%20un%20dispositivo,podremos%20almacenar%20fotos%20o%20videos..>

en la tabla 2<sup>15, 16</sup>. La cámara incorporada por defecto es la *OV2640*, es una cámara *CMOS* de bajo consumo y costo. Es importante destacar que la placa también puede ser compatible con otras cámaras mediante la interfaz de cámara *SCCB* (*Serial Camera Control Bus*). Esto permite la conexión y utilización de cámaras externas, como la *OV7670* con la placa<sup>16</sup>.

Tabla 2. Modelos de Cámaras Compatibles con la Placa *ESP32-Cam*.

Características	OV2640	OV7670
Fuente de Alimentación	Núcleo: 1,3 [VDC] ± 5 % Analogico: 2,5 3,0 [VDC] E/S: 1,7 [V] a 3,3 [V]	Núcleo: 1,8 [VDC] ± 10 % Analogico: 2,45 3,0 [VDC] E/S: 1,7 [V] a 3,0 [V]
Consumo de Energía	Modo Activo: 125 [mW] (for 15 fps, UXGA YUV mode) 140 [mW] (for 15 fps, UXGA compressed mode) Modo Espera: 900 [µA]	Modo Activo: 60mW typical(for 15 fps, VGA YUV format) Modo Espera: <20 [µA]
Rango de Temperatura	Modo Img Estable: 0°C a 50°C	Modo Operación: - 30°C a 70°C Modo Img Estable: 0°C a 50°C
Formato de Salida (8bit)	YUV(422/420) / YCbCr422 RGB 565/555 8-bit compressed data 8/10bit Raw RGB data	YUV/YCbCr 4:2:2 RGB565/555/444 GRB 4:2:2 Raw RGB Data
Tamaño del Lente	1/4"	1/6"
Velocidad máxima de transferencia de imágenes	UXGA/SXGA : 15fps SVGA: 30 fps CIF: 60 fps	VGA: 30 fps
Relación señal/ruido	40 [dB]	46 [dB]
Sensibilidad	0.6 [V/Lux-sec]	1.3 [V/(Lux • sec)]
Rango dinámico	50 [dB]	52 [dB]
Tamaño del Pixel	2.2 [µm] x 2.2 [µm]	3.6 [µm] x 3.6 [µm]
Área de Imagen	3590 [µm] x 2684 [µm]	2.36 [mm] x 1.76 [mm]
Dimensiones del paquete	5725 [µm] x 6285 [µm]	3785 [µm] x 4235 [µm]
Resolución	UXGA: 1600 x 1200 SVGA: 800 x 600	Dimensiones del paquete
Ángulo de visión	25° non-linear	25°

Luego de realizar el análisis comparativo de las cámaras que admite la placa *ESP32 - Cam*,

<sup>15</sup> *OV2640/OV7171 CMOS UXGA: (1600 x 1200) SVGA: (800 x 600) CameraChip™ Implementation Guide - MIT*. 2023. URL: [https://www.uctronics.com/download/cam\\_module/OV2640DS.pdf](https://www.uctronics.com/download/cam_module/OV2640DS.pdf) (visitado 03-04-2023).

<sup>16</sup> *OV7670/OV7171 CMOS VGA (640x480) CameraChip™ Implementation Guide - MIT*. 2023. URL: <https://web.mit.edu/6.111/www/f2016/tools/OV7670app.pdf> (visitado 03-04-2023).

se concluye que la *OV2640* es la elección más adecuada para nuestro proyecto. Esta cámara viene incluida por defecto con la *ESP32 - Cam* y ofrece una mayor resolución de imagen, lo cual resulta fundamental para la captura de imágenes de nuestra base de datos<sup>15</sup>.

**2.6.2. Batería.** Las baterías desempeñan un papel crucial en los sistemas embebidos debido a su capacidad para proporcionar energía autónoma y portátil. Los sistemas embebidos son aquellos que están diseñados para realizar tareas específicas dentro de dispositivos o equipos más grandes y a menudo requieren una fuente de energía confiable y de larga duración<sup>17</sup>. En la tabla 3<sup>18</sup> se presentan algunos ejemplos de los tipos de baterías utilizadas comúnmente para alimentar estos sistemas.

Tabla 3. Tipos de baterías.

Características	Alcalina AA	LiFePo4	LiPo - Li-ion	NiMH	Alcalina de 9 V
Tensión descarga mínima	1 [V]	2.5 [V]	2.7 [V] - 3 [V]	0.8 [V]	6 [V]
Tensión de trabajo	1.5 [V]	3 [V] - 3.2 [V]	3.7 [V]	1.2[V]- 1.25[V]	9 [V]
Tensión de carga máxima	1.65 [V]	3.65 [V]	4.2 [V]	1.4 [V]	9.9 [V]
Número de recargas	500	500	500	1000	500
Densidad de energía	80 [Wh/kg]	90 - 160 [Wh/kg]	100 - 265 [Wh/kg]	60 - 120 [Wh/kg]	80 [Wh/kg]
Autodescarga / mes	0	0	<10 %	30 %	0

Después de observar las diversas opciones de baterías para alimentar nuestro sistema embebido, se determinó que las baterías recargables tipo *Li-ion* son la elección ideal. Estas baterías destacan por su durabilidad, tensión de carga máxima, bajo mantenimiento y mayor densidad de energía, lo cual las convierte en la opción más adecuada para su implementación en nuestro proyecto, se emplearon baterías *Li-ion 18650* con *2600 mA/h* y *3.7 [v]*<sup>18</sup>.

<sup>17</sup> *Diferentes niveles de voltaje de la placa del microcontrolador ESP32.* Section: Arduino. 29 de mayo de 2021. URL: <https://elosciloscopio.com/mejor-bateria-para-esp32/> (visitado 03-04-2023).

<sup>18</sup> *Comparación de batería de densidad de energía: celdas cilíndricas y prismáticas.* 2023. URL: <https://www.epectec.com/batteries/cell-comparison.html> (visitado 03-04-2023).

**2.6.3. Protocolo de comunicación inalámbrica.** La *ESP32 - Cam* ofrece una amplia compatibilidad con diversos protocolos de comunicación inalámbrica. Entre los protocolos admitidos se encuentran el *Wi-Fi (802.11 b/g/n)* y el *Bluetooth (BR/EDR y BLE)*<sup>19</sup>. Es importante destacar que la placa no cuenta con soporte integrado para protocolos inalámbricos como *LoRa* o *Zigbee*, Sin embargo, estos protocolos pueden habilitarse mediante el uso de módulos externos si los requerimientos del proyecto así lo demandan. Esto brinda flexibilidad para adaptar la *ESP32 - Cam* a distintas necesidades de conectividad inalámbrica. En este proyecto, nos centraremos en los protocolos de comunicación inalámbrica integrados en la *ESP32 - Cam*, tal como se muestra en la tabla 4<sup>19, 20, 21</sup>.

Tabla 4. Especificaciones técnicas *WiFi 802.11n* y *Bluetooth 4.2*

Características	WiFi 802.11n	Bluetooth 4.2 BR/EDR
Velocidad Transmisión máx	100 [Mbps] a 600 [Mbps]	3 [Mbps] a 24 [Mbps]
Banda de frecuencia	2,4 [GHz] y 5 [GHz]	2,4 [GHz]
Cobertura	30 - 45 [m] a 2,4 [GHz] 15 - 30 [m] a 5 [GHz]	hasta 10 [m]
Modulación	OFDM	GFSK
Ancho de Banda	20 [MHz] a 40 [MHz]	1 [MHz]
Técnica de transmisión	MIMO	BR/EDR
Seguridad	WPA	E0/SEGURIDAD+
Relación señal/ruido	20 [dB]	10 [dB]
topología de red	punto a punto y punto a red	Punto a punto
Latencia	1 a 15 [ms]	20 a 40 [ms]

<sup>19</sup> Daniel Moreno Martín. *Estudio y evaluación de comunicaciones Wi-Fi basadas en IEEE 802.11ac*. (Accessed on 03/03/2023). Oct. de 2017. URL: <https://upcommons.upc.edu/bitstream/handle/2117/110125/memoria.pdf?sequence=1&isAllowed=y>.

<sup>20</sup> Kinza Yasar. *What is 802.11n? | Definition from TechTarget*. Mobile Computing. (Accessed on 03/03/2023). URL: <https://www.techtarget.com/searchmobilecomputing/definition/80211n>.

<sup>21</sup> Neha T. *Difference Between Bluetooth and Wifi (with Comparison Chart)*. Tech Differences. 16 de ago. de 2017. URL: <https://techdifferences.com/difference-between-bluetooth-and-wifi.html>.

**2.6.4. Protocolo de transferencia de datos.** El protocolo *FTP (File Transfer Protocol)*, Un protocolo de comunicación es un conjunto de reglas y normas que permiten la comunicación entre dos o más dispositivos en una red. Estos protocolos definen cómo se establece mantiene y finaliza la comunicación, los protocolos establecen el intercambio de información, el formato de los mensajes y la secuencia de operaciones necesarias para una comunicación.

Dentro de una red se utilizan diversos protocolos de comunicación en diferentes capas y contextos que permiten la transferencia de datos en distintos formatos, algunos de los más comunes son: *Protocolos de Internet (TCP/IP)*, *Datagramas de Usuario (UDP)*, *transferencia de Hipertexto (HTTP)*, *Transferencia de Correo Simple (SMTP)*, *Resolución de Nombres de Dominio (DNS)* y *transferencia de archivos (FTP)*, se ha seleccionado el protocolo *FTP* debido a sus características, las cuales lo convierten en la opción ideal para cumplir los requerimientos solicitados en este proyecto.

***Transferencia de Archivos (FTP)***: Es un protocolo estándar de Internet utilizado para la transferencia de archivos entre un cliente y un servidor en una red. Este protocolo permite cargar, descargar y manipular archivos desde un servidor remoto a través del puerto 21, *FTP* admite varios modos de transferencia de archivos entre el cliente y el servidor como el modo de transferencia en bloque y el modo de transferencia en modo de texto.

El protocolo *FTP* fue seleccionado para este proyecto debido a sus características que se adaptan a las necesidades específicas del mismo. Algunos aspectos relevantes a destacar son los siguientes:

- La transferencia rápida y eficiente de imágenes especialmente en redes locales.
- Permite establecer un control de permisos y accesos para los archivos y directorios, proporcionando un mayor control sobre quién puede acceder y descargar las imágenes transferidas.
- La eficiencia para transferir múltiples imágenes, lo que ahorra tiempo y mejora la produc-

tividad, especialmente cuando se requiere transferir grandes volúmenes de imágenes en un corto período de tiempo.

## **2.7. PROGRAMA DE DISEÑO.**

**2.7.1. SolidWorks.** Es un software de diseño asistido por computadora (*CAD*) utilizado principalmente para el diseño en 3D de productos mecánicos, componentes industriales y es ampliamente utilizado en la industria de la ingeniería, el diseño industrial y la fabricación.

*SolidWorks* ofrece un conjunto de herramientas avanzadas para crear modelos tridimensionales y modificar objetos virtuales en 3D utilizando técnicas de modelado paramétrico, realizar simulaciones y pruebas de ingeniería, generar planos de fabricación y colaborar en proyectos de diseño. Esta herramienta ayuda a los ingenieros a optimizar sus diseños y tomar decisiones informadas antes de la fabricación. El *software* utiliza una interfaz intuitiva y amigable, lo que facilita su uso tanto para principiantes como para usuarios experimentados.<sup>22</sup>

## **2.8. IMPRESIÓN 3D.**

La producción de un objeto impreso en 3D se logra a través de un proceso aditivo de filamentos (*FDM*) Fused Deposition Modeling, que crea un objeto colocando capas sucesivas hasta que se crea el objeto por completo. Cada una de estas capas se puede ver como una sección transversal del objeto. La impresión 3D, a veces denominada fabricación aditiva, consiste en superponer materiales como plásticos, compuestos o biomateriales para crear objetos de varias formas, tamaños, rigidez y colores.<sup>23</sup>

---

<sup>22</sup> Sergio Gómez González. *SolidWorks®*. 2008. URL: <https://elibro-net.bibliotecavirtual.uis.edu.co/es/ereader/uis/45924?page=18> (visitado 03-06-2023).

<sup>23</sup> *Qué es la impresión 3D, importancia y cómo funciona la fabricación aditiva*. Section: BLOG INDUSTRIA 4.0. 12 de feb. de 2021. URL: <https://www.cursosaula21.com/que-es-la-impresion->

**2.8.1. Tipos de Materiales Plásticos para Impresión 3D.** Los plásticos de impresión *3D* son materiales livianos en forma de filamentos con una amplia gama de propiedades físicas adecuadas para proyectos de creación de prototipos y otras aplicaciones prácticas.

**Filamento Acido Poli-Lactico (PLA).** El plástico más común para la impresión *3D*. Alta rigidez, buen acabado final y precio razonable. El *PLA* es un termoplástico biodegradable para prototipos no funcionales de bajo costo. El acabado es mejor que el *ABS*, pero más quebradizo. No es apto para altas temperaturas.<sup>23</sup>.

---

3d/ (visitado 03-04-2023).

### **3. NORMATIVA DEL CONSUMO DE AGUA POTABLE**

El agua es un elemento vital para todos los seres vivos, es necesaria para la producción de alimentos, la higiene y el bienestar humano en general, por esta razón cuidar el agua es una tarea primordial que implica utilizarla de manera eficiente y responsable, evitando su desperdicio y contaminación al aplicar prácticas como el cierre adecuado de grifos, la reparación de fugas, reutilización del agua y el uso de tecnologías más eficientes en el consumo hídrico.

El acceso al agua potable es esencial para la vida y el bienestar de las comunidades en todo el mundo. En Colombia el suministro de agua y el manejo de los servicios de agua potable ha crecido rápidamente gracias al aumento de la población y al desarrollo económico del país. El acceso al agua potable es considerado un derecho fundamental por esta razón el gobierno ha implementado diferentes políticas y programas que varían según región, para mejorar la infraestructura y garantizar que el suministro de agua llegue a cada rincón del país.

En Bucaramanga el *AMB S.A. E.S.P* administra el recurso a través de la facturación por consumo medido, donde los usuarios pagan en función de la cantidad de agua que utilizan. Este dato es controlado a través de medidores de agua instalados en los hogares y establecimientos comerciales, lo que permite registrar mes a mes el consumo real de cada usuario.

La ciudad ha implementado sistemas de tarifas progresivas, donde el costo por metro cúbico de agua aumenta a medida que se incrementa el consumo. Esta estrategia fomenta el uso responsable del agua, además se utiliza la forma de facturación con beneficios de tarifa social que busca garantizar el acceso al agua potable a los sectores con menos recursos económicos, otorgando subsidios a los usuarios de bajos ingresos que no superen el consumo básico, lo que les permite acceder al servicio a un costo asequible e incentiva al uso responsable del preciado líquido.

### **3.1. TARIFAS DE CONSUMO DE AGUA POTABLE.**

El *Acueducto Metropolitano de Bucaramanga S.A. E.S.P* es responsable de establecer y aplicar las normas relacionadas con el suministro de agua potable en los municipios de Bucaramanga, Floridablanca y Girón. Donde se instalan medidores de agua que registran la cantidad de agua consumida por los usuarios en metros cúbicos ( $m^3$ ).

El *AMB S.A. E.S.P* utiliza un modelo de tarifas progresivas para el consumo residencial, lo que implica que el costo por metro cúbico aumenta a medida que se incrementa el consumo. Además, se implementa un sistema de categorías basado en la estratificación socioeconómica de los usuarios, beneficiándose a los estratos de menores ingresos. Este sistema aplica porcentajes de subsidio a los costos de cargo fijo y consumo básico. Sin embargo, el consumo complementario y el consumo suntuario no reciben dicho beneficio.

En el caso del uso no residencial, las tarifas se determinan en función del tipo de servicio prestado, ya sea comercial, industrial, oficial, especial, temporal o pilas públicas. Únicamente las pilas públicas cuentan con un porcentaje de subsidio en el cargo por consumo fijo y el cargo por consumo básico, como se detalla en las siguientes tablas 5 y 6 <sup>24</sup>.

**3.1.1. Consumo básico.** El consumo básico de agua es la cantidad mínima necesaria para satisfacer las necesidades básicas de una persona o un hogar, como beber, cocinar, lavar platos y utensilios de cocina, higiene personal y limpieza básica del hogar. Esta cantidad puede

---

<sup>24</sup> *Acto de gerencia tarifas 2023*. <https://www.amb.com.co:8444/DocumentoInfo/actotarifas2023.pdf>. (Accessed on 28/04/2023). Abr. de 2023.

Tabla 5. Tarifa de servicio público de agua potable para uso residencial.

USO RESIDENCIAL	ESTRATO 1			ESTRATO 2		ESTRATO 3			ESTRATO 4	ESTRATO 5	ESTRATO 6
	B/MANGA	FLORIDA	GIRÓN	B/GA-F/DA	GIRÓN	B/MANGA	FLORIDA	GIRÓN			
SUBSIDIO / CONTRIBUCIÓN	-50 %	-45 %	-25 %	-30 %	-20 %	-10 %	-15 %	-5 %	0 %	50 %	60 %
CARGO FIJO (\$/Sus.Mes)	4959,31	5455,24	7438,97	6943,03	7934,90	8926,76	8430,83	9422,69	9918,62	14877,93	15869,79
CONSUMO BÁSICO (\$/m <sup>3</sup> )	1478,47	1626,32	2217,71	2069,86	2365,55	2661,25	2513,40	2809,09	2956,94	4435,41	4731,10
CONS. COMPLEMENTARIO (\$/m <sup>3</sup> )	2956,94	2956,94	2956,94	2956,94	2956,94	2956,94	2956,94	2956,94	2956,94	4435,42	4731,11
CONS. Suntuario (\$/m <sup>3</sup> )	2956,94	2956,94	2956,94	2956,94	2956,94	2956,94	2956,94	2956,94	2956,94	4435,43	4731,12

Tabla 6. Tarifa de servicio público de agua potable para uso no residencial.

USO NO RESIDENCIAL	COMERCIAL	INDUSTRIAL	OFICIAL	ESPECIAL	TEMPORAL			PILAS PUBLICAS		
					BGA	FLORIDA	GIRON	BGA	FLORIDA	GIRÓN
SUBSIDIO / CONTRIBUCIÓN	50 %	30 %	0 %	0 %	50 %	60 %	50 %	-70 %	-70 %	-50 %
CARGO FIJO (\$/Sus.Mes)	14877,93	12894,21	9918,62	9918,62	14877,93	15869,79	14877,93	2975,93	2975,93	4959,31
CARGO CONSUMO (\$/m <sup>3</sup> )	4435,41	3844,02	2956,94	2956,94	4435,41	4731,10	4435,41	887,08	887,08	1478,47

variar dependiendo del tamaño del hogar, la ubicación geográfica y las condiciones climáticas, según lo menciona el artículo “Consumo Básico de Agua Potable en Colombia” escrito por Gustavo Chacon M., Imelda Lizcano y Yefer Aspilla Lara<sup>25</sup>.

En Bucaramanga se determinó que el consumo básico de agua para uso residencial es de 16 m<sup>3</sup>. Es importante destacar que los estratos 1, 2 y 3 cuentan con un porcentaje de subsidio en el cargo por consumo básico, mientras que los estratos 5 y 6 se les aplica un porcentaje de contribución como se detalla en la tabla 5.

**3.1.2. Consumo complementario y suntuario.** El consumo complementario hace referencia al uso adicional de agua que excede el consumo básico establecido para satisfacer las necesidades de una persona o un hogar. En la ciudad se considera consumo complementario hasta los 32 m<sup>3</sup>, que representa el doble del consumo básico habitual. Es importante destacar que este tipo de consumo no cuenta con porcentajes de subsidio para ningún estrato socio-

<sup>25</sup> *Consumo básico de agua potable en Colombia, author = M, Gustavo Chacón and Lizcano, Imelda and Lara, Yefer Aspilla,| Tecnogestión: Una mirada al ambiente. es-ES. Feb. de 2016. URL: <https://revistas.udistrital.edu.co/index.php/tecges/article/view/4379> (visitado 07-06-2023).*

económico, dado que es un consumo adicional al necesario para cubrir las necesidades.

El consumo suntuario hace referencia al consumo de agua que excede al complementario superando los  $32 m^3$ , se caracteriza por ser un uso excesivo, innecesario o exagerado de agua en actividades consideradas lujosas o para fines que no son esenciales para la vida cotidiana, lo que puede tener un impacto negativo en la disponibilidad y la conservación del recurso hídrico. Por tanto, es importante promover un uso responsable del agua, impidiendo prácticas de consumo suntuario y fomentar la conservación del recurso para garantizar su disponibilidad para toda la comunidad a largo plazo.

**3.1.3. Cálculo del servicio de acueducto.** A continuación, se presenta el desglose del cobro de agua de un hogar estrato tres en la ciudad de Bucaramanga, donde se observa el cobro mensual por el servicio de acueducto generado por *AMB S.A. E.S.P.* Se incluyen los conceptos de cargo fijo, consumo básico, donde se aplica el porcentaje de subsidio del *10 % (estrato 3)*, también se observan el cargo de consumo complementario y suntuario sin beneficio de subsidio, figura 4.


Es fundamental comprender los conceptos de cobro aplicados en la facturación para obtener una estimación mensual cercana a la real. Al entender el significado de cada uno de los valores registrados, podremos comprender por qué se aplican y por qué varían en los precios, estos conceptos son fundamentales en el desarrollo de este proyecto ya que nos centraremos en estimar el cobro por concepto de acueducto, donde el precio por consumo total se puede hallar siguiendo la ecuación (1).

$$CT = (CF - \%SE) + [(CB * RC) - \%SE] + (RC * CC) + (RC * CS) \quad (1)$$

Donde :

Referencia de Consumo: *RC*

Figura 4. Datos de facturación para el servicio de acueducto.

ACUEDUCTO	
REFERENCIA CONSUMO (\$/m ) <sup>3</sup>	\$ 2,957
REFERENCIA CARGO FIJO (\$/mes)	\$ 9,919
TASAS AMBIENTALES (\$/m ) <sup>3</sup>	5,35
	
CONCEPTOS	VALOR
CARGO FIJO ACUEDUCTO	\$9,919
Consumo BASICO (1-16)	\$47,311
Consumo COMPL. (17-32)	\$47,311
Consumo SUNTU. (33-37)	\$14,785
AJUSTE DECENA FACTURA	\$5
SUBSIDIO CARGO FIJO(-10%)	\$-992
SUBSID. CONS BASICO(-10%)	\$-4,731
<b>SUBTOTAL MES</b>	<b>\$113,608</b>
<b>SALDO EN MORA</b>	<b>\$0</b>
<b>TOTAL ACUEDUCTO</b>	<b>\$113,608</b>

Cargo Fijo :  $CF$

Consumo Básico:  $CB$

Consumo Complementario:  $CC$

Consumo Suntuario:  $CS$

Subsidio por Estrato:  $\%SE$

### 3.2. NORMA TÉCNICA PARA MEDIDORES DE AGUA.

El *acueducto metropolitano de Bucaramanga S.A. E.S.P.* Establece en la norma técnica interna los requisitos mínimos que deben cumplir los medidores de agua potable para ser instalados en establecimientos residenciales y no residenciales en los municipios de Floridablanca, Bucaramanga y Girón.

El *Acueducto Metropolitano de Bucaramanga S.A. E.S.P.* establece en su norma técnica los diámetros nominales y especificaciones técnicas requeridas para instalar un contador de agua en los municipios de Bucaramanga , Floridablanca y Girón, se aceptan medidores de velocidad con diámetro nominal *DN15 (1/2")*, *DN20 (3/4")* y *DN25 (1")*, las especificaciones técnicas detalladas de la norma se encuentran disponibles en el anexo 1. El contador de agua modelo *R200* utilizado en este proyecto, posee un diámetro nominal *DN15 (1/2")* que cumple con las especificaciones técnicas requeridas por el *AMB S.A. E.S.P* <sup>26</sup>.

### **3.3. CONTADORES DE AGUA.**

Los contadores son elementos que se usan para obtener información sobre la cantidad o volumen de agua que fluye a través de la tubería o sistema de suministro de agua de forma más precisa y se instalan en hogares, edificios comerciales, industrias y otros lugares donde se requiere medir y facturar el consumo de agua.

El medidor de agua consta de varios componentes, incluyendo una carcasa, un mecanismo de medición y una serie de ruedas giratorias como se muestra en la figura 5. Cuando el agua fluye a través del medidor, el mecanismo de medición registra la cantidad de agua que pasa y las ruedas giratorias se mueven proporcionalmente al caudal de agua<sup>27</sup>.

**3.3.1. Tipos de contadores de agua.** La elección del tipo de medidor depende de diversos factores, como el caudal esperado, la precisión requerida y las características específicas

---

<sup>26</sup> *Norma Técnica Interna de Medidores - amb - Acueducto Metropolitano de Bucaramanga.* es. <https://www.amb.com.co/amb/conoce-amb/nuestros-procesos/norma-tecnica-interna-de-medidores/>. Feb. de 2022.

<sup>27</sup> redaccion iAgua. *¿Qué es un contador de agua y cuántos tipos hay?* iAgua. Publisher: iAgua. 11 de mayo de 2021. URL: <https://www.iagua.es/respuestas/que-es-contador-agua-y-cuantos-tipos-hay> (visitado 06-07-2023).

Figura 5. Despiece del contador de agua modelo R200.



de la instalación. Existen medidores para agua fría o caliente, los contadores se dividen en dos grupos, correspondientes a los contadores mecánicos y no mecánicos que a su vez se subdividen<sup>27</sup>.

**Contadores mecánicos.** Este tipo de contadores operan por medio de componentes mecánicos presentes en su interior y por la interacción de turbinas, paletas o ruedas accionadas por el flujo de agua que atraviesa el contador. Por ejemplo, en los contadores de turbina interna, esta gira de acuerdo al caudal de agua que pasa a través del contador. A medida que la turbina gira, se registran el número de vueltas para calcular el volumen de agua consumida. Por otro lado, los contadores de paletas en su interior hacen que estas se desplacen gracias al flujo de agua, este movimiento permite medir el consumo. Los contadores mecánicos se subdividen en: contadores de velocidad, chorro único, chorro múltiple, tipo woltman, tangenciales y proporcionales. En este proyecto se emplea un contador mecánico de chorro único modelo *R200*.

**Contador de Chorro Único** Este tipo de contadores se caracteriza por tener una única entrada de agua que impacta directamente sobre una turbina diseñada para girar, se encuentra ubicada de manera que el agua impacta tangencialmente sobre sus aspas para producir una rotación a medida que el agua fluye a través de ella. La turbina gira proporcionalmente a la velocidad y al volumen del agua que pasa, la rotación generada se transmite a un mecanismo conectado a una serie de engranajes y medidores que permiten visualizar y registrar el consumo de agua a través de una esfera o dispositivo numérico digital en el que se muestra el volumen acumulado<sup>27</sup>.

**3.3.2. Contador de agua modelo R200.** El medidor de agua modelo *R200*, es un tipo de contador mecánico utilizado para medir el consumo de agua en instalaciones residenciales y comerciales, se basa en tecnología mecánica para medir el caudal de agua, utiliza el principio del chorro único, su tamaño más común es el *DN15* (diámetro nominal de 15 mm o aproximadamente 1/2 pulgada), tiene una pantalla o dial para mostrar la lectura del consumo de agua acumulado y cuenta con protección contra manipulaciones o sistemas de sellado para garantizar la precisión y la integridad del medidor.

Este proyecto utiliza el medidor de agua modelo *R200*, con un diámetro nominal *DN 15* de 1/2" de transmisión mecánica, que cumple con las normas *MID (Directiva de Instrumentos de Medición) OIML R49, en 14154 e ISO 4064, NTC - 1063-1 : V- 2007* y la *Resolución 0330 de Junio de 2017* mostrados en la tabla 7 . Este modelo de medidor de agua está diseñado con características especiales para garantizar su precisión y durabilidad. Por ejemplo, se encuentra sellado herméticamente con un líquido especial que mantiene clara su lectura, cuenta con un sello de seguridad que evita el desmontaje del contador por el usuario o terceros, la transmisión mecánica es inmune a campos magnéticos, también cuenta con un filtro que protege las partes internas del contador de partículas no deseadas evitando obstrucciones,

además cuenta con una válvula que evita el retorno del agua y su estructura es fabricada en bronce<sup>28</sup>.

Tabla 7. Caudales e Incertidumbre para contador modelo R200, según NTC ISO 4064-1 : v2017 .

CERTIFICADO DE CALIBRACIÓN VIGENTE			
Tipo de caudal.	Nombre de Caudal	ERROR RELATIVO INDICACIÓN E( %)	INCERTIDUMBRE (U %)
Q1	caudal mínimo	1.89 %	0.33 %
Q2	caudal transición	-0.70 %	0.34 %
Q3	caudal permanente	1.39 %	0.11 %

El anexo 2, se presenta una tabla comparativa de las especificaciones técnicas requeridas por el *Acueducto Metropolitano de Bucaramanga S.A. E.S.P* para los contadores de agua Vs las especificaciones de calibración del modelo de contador de agua utilizado en este proyecto <sup>26</sup>.

**3.3.3. Lectura de contador de agua.** La lectura de un contador de agua digital puede ser compleja, gracias a la cantidad de números que se observan en el dial y al constante movimiento de las ruedas medidoras, estas características pueden generar confusión al intentar identificar correctamente los números durante la transición entre dos valores, por esta razón es importante tener en cuenta la lectura anterior para realizar una correcta interpretación de los datos en el dial del medidor de agua.

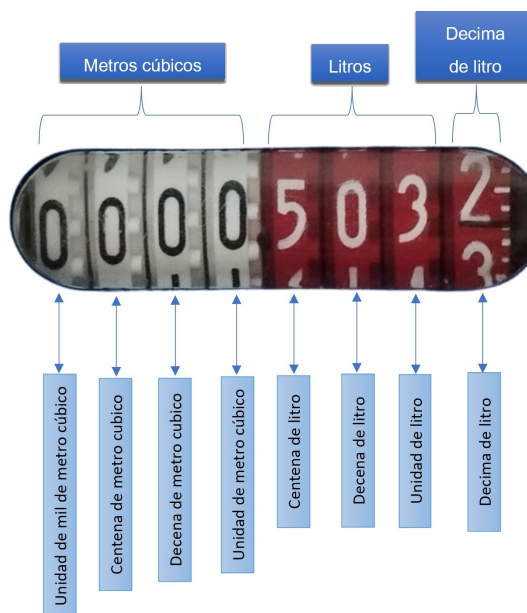
La forma correcta de leer los contadores de agua es de izquierda a derecha, de la misma forma que se lee el contador de kilómetros de un automóvil. Los cuatro primeros números indican la cantidad de agua en metros cúbicos que han pasado por el contador, estas cifras se identifican en color negro y fondo blanco, mientras que los tres primeros números en color blanco y fondo rojo indican la cantidad de litros de agua consumido en fracción de metros

---

<sup>28</sup> *Medidor volumétrico metálico preequipado R200 calibrado 1/2"*. es. URL: <https://controlagua.com/producto/medidor-volumetrico-metalico-preequipado-r200-calibrado-1-2/> (visitado 12-06-2023).

cúbicos y el ultimo indicador de color blanco y fondo rojo muestra la proporción de agua consumida en un litro, esta identificación por colores es valida para el modelo de contador de agua elegido en este proyecto como se muestra en la figura 6.

Figura 6. Lectura dial del contador de agua.



#### 4. PROTOCOLO DE TRANSMISIÓN DE DATOS

La captura de imágenes juega un papel fundamental en el desarrollo de este proyecto ya que a través de ellas podemos identificar con precisión el consumo de agua registrado en los contadores. Por esta razón se ha seleccionado la *ESP32-Cam* con una cámara *OV2640* de 2 *Mpx* que permite capturar imágenes nítidas y detalladas, además es posible ajustar la configuración de la cámara a diferentes condiciones de iluminación y requisitos de calidad de imagen y posee soporte para diferentes formatos de imagen, lo que permite obtener imágenes nítidas y detalladas para su posterior procesamiento.

La *ESP32-Cam* también cuenta con capacidad de conectividad inalámbrica a través de *Wi-Fi*, lo que facilita la transferencia de imágenes a un servidor local sin la necesidad de cables adicionales. Se creó un servidor local utilizando *Filezilla* que es un software de cliente de *FTP* de código abierto que proporciona una interfaz gráfica fácil de usar para la transferencia de archivos entre sistemas locales y servidores remotos. Un protocolo *FTP* (*File Transfer Protocol*), es un protocolo de red para transferencia de archivos entre sistemas conectados a una red *TCP/IP*. Permite a los usuarios cargar archivos desde su sistema local a un servidor remoto, y descargar archivos desde el servidor remoto a su sistema local.

La cámara *OV2640* captura una imagen con resolución *UXGA 1600 x 1200 pixeles* del contador de agua cada 3 horas y verifica la disponibilidad del servidor local *FTP*, si este se encuentra disponible las imágenes capturadas son enviadas a un directorio del equipo para su posterior procesamiento y elimina las imágenes de la tarjeta de memoria *SD* de 4 *Gbytes*, en caso contrario cuando el servidor no se encuentra disponible se guardan las imágenes en la tarjeta de memoria hasta que se detecte nuevamente disponibilidad en el servidor, luego de realizar este proceso la *ESP32-Cam* entra en sueño profundo hasta completar nuevamente el tiempo estipulado para tomar una nueva imagen con el fin de preservar la carga de la batería,

El código para este proceso se encuentra en *GitHub Código\_ESP32-Cam*, anexo4.

#### 4.1. TRANSFERENCIA DE DATOS AL SERVIDOR LOCAL.

Se captura una imagen cada tres horas con una resolución de imagen de *UXGA de 1600 x 1200 píxeles*. Sin embargo, antes de realizar la captura, se verifica la disponibilidad de la memoria *PSRAM*. En caso de que la memoria *PSRAM* no esté disponible, se captura una imagen con una resolución menor, *XGA de 1024 x 768 píxeles*, asegurando que se obtenga una imagen dentro del intervalo de tiempo establecido, permitiendo continuidad en la toma de datos para su posterior análisis, código 4.1.

Código 4.1. Escrito en C para establecer la resolución de la imagen.

```
1  if (psramFound()) { // Verificar si se encuentra una memoria PSRAM
2      config.frame_size = FRAMESIZE_UXGA; // UXGA (1600x1200)
3      config.jpeg_quality = 10; // Calidad de compresion JPEG: 10
4      config.fb_count = 2; // Contador de buffers de frame: 2
5  } else {
6      config.frame_size = FRAMESIZE_XGA; // XGA (1024x768)
7      config.jpeg_quality = 12; // Calidad de compresion JPEG: 12
8      config.fb_count = 1; // Contador de buffers de frame: 1
9  }
10 esp_deep_sleep(3 * 3600 * 1000000); // toma foto cada 3 horas
```

Se realiza una espera de 10 segundos al inicializar la cámara con el fin de asegurar que esta se estabilice adecuadamente antes de tomar la foto. Esta espera es necesaria para garantizar que la cámara esté lista y pueda capturar la imagen correctamente, preservando así los colores originales de la escena. Código 4.2.

Código 4.2. Inicialización de cámara *OV2640*.

```
1  esp_err_t err = esp_camera_init(&config);
2  if (err != ESP_OK) {
```

```

3     Serial.printf("Error al inicializar la camara: 0x%x", err);
4     return; }
5     delay(10000);

```

Las imágenes se capturan y guardan con un formato de nombre que sigue la estructura "día\_mes\_año\_hora", por medio de un cliente *NTP* se obtiene el los datos de fecha y hora, los cuales se ajustan la hora del meridiano de Colombia. Código4.3.

Código 4.3. Obtención de fecha por cliente *NTP*.

```

1 NTPClient timeClient(ntpUDP, "pool.ntp.org", (-3600*5), 60000);
2 // Obtener fecha y hora actual
3 unsigned long epochTime = timeClient.getEpochTime();
4 struct tm* ptm = gmtime((time_t*)&epochTime);
5 int year = ptm->tm_year + 1900;
6 int month = ptm->tm_mon + 1;
7 int day = ptm->tm_mday;
8 int hour = ptm->tm_hour;
9 int minute = ptm->tm_min;
10 String dateTime;
11 dateTime = String(day)+"_"+String(month)+"_"+String(year)+"_"+
    String(hour);
12 String path = "/" + String(dateTime) + ".jpg";

```

Se han creado tres funciones. La primera función se encarga de verificar la disponibilidad del servidor y asignar el *puerto 21* para la transmisión de archivos, se realiza una verificación para comprobar la disponibilidad del servidor local *FTP*. Si el servidor está disponible, la función devuelve un valor booleano "*true*". En caso contrario, devuelve un valor "*false*". Código 4.4 .

Código 4.4. Comprobación de disponibilidad del servidor.

```

1 bool ftpAvailable() {
2     Serial.println("Comprobando disponibilidad servidor FTP...");

```

```

3   WiFiClient client;
4   if (client.connect(ftp_server, 21)) {
5       client.stop();
6       return true;
7   } else {
8       return false; }

```

Si el servidor local *FTP* está disponible, se procede al envío de las imágenes capturadas y guardadas en la tarjeta *SD*. Cada imagen se envía en *buffers de 256 bits*, una vez completado el envío de una imagen, el archivo correspondiente se cierra y la imagen se elimina de la tarjeta *SD* para dar paso al siguiente envío. Este proceso se repite hasta que todas las imágenes hayan sido enviadas al servidor *FTP* y eliminadas de la tarjeta de memoria. Código 4.5.

Código 4.5. Envío de imágenes al servidor y liberación de espacio en la *SD*.

```

1 void sendImagesToServerAndDelete() {
2     File root = SD_MMC.open("/"); // Abrir carpeta raiz tarjeta SD
3     File file = root.openNextFile();
4     while (file) {
5         if (!file.isDirectory()) {
6             String filename = file.name(); // Nombre del archivo
7             Serial.println(filename);
8             if (filename.endsWith(".jpg")) {
9                 Serial.println("Subiendo " + filename);
10                ftp.InitFile("Type I");
11                ftp.NewFile(filename.c_str());
12                if (file.size() > 0) {
13                    uint8_t buf[256];
14                    size_t len;
15                    while ((len = file.read(buf, sizeof(buf))) > 0) {
16                        ftp.WriteData(buf, len); } }

```

```

17     ftp.CloseFile(); // Cerrar el archivo
18     file.close();
19     SD_MMC.remove(filename.c_str()); } }

```

Si el servidor local no está disponible, se procede a tomar la imagen y guardarla en la tarjeta *SD*. En lugar de enviar la imagen directamente al servidor, se almacena localmente para transferirla en el momento en que el servidor esté nuevamente disponible. Código 4.6.

**Código 4.6.** Captura y almacenamiento local en la tarjeta *SD*.

```

1     fs::FS& fs = SD_MMC;
2     Serial.printf("Nombre archivo de imagen: %s\n", path.c_str());
3     File file = fs.open(path.c_str(), FILE_WRITE);
4     if (!file) {
5         Serial.println("Error al abrir archivo modo de escritura");
6     } else {
7         file.write(fb->buf, fb->len);
8         Serial.printf("Guardado en la ruta: %s\n", path.c_str()); }
9     file.close(); // Cerrar el archivo
10    esp_camera_fb_return(fb); } //Liberar buffer de frame de la camara

```

## 5. BASE DE DATOS.

Una buena base de datos es fundamental ya que con esta información se realiza el entrenamiento de las redes neuronales. Estos conjuntos de datos proporcionan los elementos necesarios para que los modelos aprendan y logren generalizar de manera precisa y efectiva.

Los datos de entrenamiento, validación y testeo para este proyecto se generaron tomando imágenes directas de un contador de agua modelo *R200*. Este contador cuenta con 8 ruedas que indican el consumo de agua. La obtención de la base de datos se llevó a cabo haciendo girar las ruedas para obtener todas las combinaciones posibles, esta metodología garantiza una cobertura completa de los diferentes valores y configuraciones que pueden presentarse en el contador, permitiendo así un entrenamiento más completo y preciso de las redes neuronales, estas se alimentan de datos de entrada y utiliza algoritmos para procesarlos y ajustar los pesos de sus conexiones internas. Estos pesos determinan la capacidad de la red para reconocer patrones, hacer predicciones y tomar decisiones, una base de datos completa y bien etiquetada permite que la red neuronal tenga acceso a información relevante y estructurada, facilitando el aprendizaje de patrones y relaciones complejas.

Se crearon dos bases de datos para el entrenamiento de las redes neuronales. La primera base de datos se utilizó para entrenar la red neuronal de segmentación, mientras que la segunda base de datos se utilizó para entrenar la red neuronal de detección. Las imágenes correspondientes a estas bases de datos fueron capturadas a través de una *ESP32 - Cam*, ubicada en la parte superior de una estructura de fácil acople diseñada para este proyecto, la estructura cuenta con un orificio que permite empotrar la cámara *OV2640* de la *ESP32 - Cam*, permitiendo capturar las imágenes del contador de agua. Además, cuenta con tres tiras de *LED* 's estratégicamente ubicadas dentro de la estructura, garantizando un ambiente de iluminación controlada para una mejor toma de datos.

## 5.1. BASE DE DATOS DE SEGMENTACIÓN.

La base de datos está compuesta por un total de 578 imágenes *RGB* de  $224 \times 224$  píxeles, se divide aproximadamente en un 70 % para entrenamiento, un 20 % para validación y un 10 % para testeo, obteniendo la siguiente distribución inicial tabla 8.

Tabla 8. Base de datos inicial para segmentación.

Clase.	Total imágenes.
Datos Entrenamiento	408
Datos Validación	114
Datos test	58

El aumento de datos de entrenamiento es fundamental para obtener mejores resultados en el rendimiento del modelo de aprendizaje profundo a entrenar. En este caso, se aplicó un aumento de datos y etiquetado a las 408 imágenes de entrenamiento utilizando la plataforma *Roboflow*. Las etiquetas se asignan en formato *YOLOv8*, esta etiqueta consiste en asignar puntos alrededor de la sección de interés que se va a segmentar. Las transformaciones aplicadas a las imágenes originales incluyeron una *rotación aleatoria* en un rango de  $-6^\circ$  a  $+6^\circ$ , un *desenfoque Gaussiano aleatorio* entre 0 y 1 píxel, y un *ruido equivalente al 1 % de los píxeles*. Es importante destacar que se realizaron estas transformaciones de forma leve, teniendo en cuenta que se trabaja en un entorno controlado. El objetivo fue simular diversas condiciones o escenarios que podrían encontrarse en la vida real, con el propósito de fortalecer la capacidad del modelo para adaptarse a situaciones diversas y mejorar su capacidad de segmentación de los contadores de agua, después de realizar el aumento de datos se obtuvo una [Base\\_de\\_datos\\_de\\_segmentación](#) con su respectiva etiqueta, el detalle de la cantidad de imágenes obtenidas se presenta en la Tabla 9. Puede descargar la base de datos en el Anexo4.

## 5.2. BASE DE DATOS DE DETECCIÓN.

Se crearon dos bases de datos para entrenar la redes neuronales de detección, ya que se realizará una comparativa entre los modelos *YOLOv8* y *Detectron2*. Cada uno de los modelos

Tabla 9. Base de datos aumentada para segmentación *YOLOv8*.

<b>Clase.</b>	<b>Total imágenes etiquetadas.</b>
Datos Entrenamiento	1224
Datos Validación	114
Datos test	58
<b>Total datos</b>	<b>1396</b>

requiere imágenes con dimensiones  $640 \times 640$  píxeles y etiquetado en un formato específico, por lo que las bases de datos se adaptaron a los requisitos de cada modelo. Las imágenes utilizadas en ambas bases de datos corresponden al resultado de la salida de la red neuronal de segmentación. Esto significa que las imágenes vienen con el área de interés (dial del contador) segmentada.

**5.2.1. Base de datos de detección para modelo *YOLOv8*.** La base de datos utilizada para entrenar el modelo de detección con *YOLOv8* se compone por el conjunto de imágenes de salida generadas por la red neuronal de segmentación. Estas imágenes están segmentadas en el área de interés específicamente en el dial del contador. En total, se cuenta con 578 imágenes segmentadas en formato *RGB* de  $640 \times 640$  píxeles, se divide aproximadamente en un 75 % para entrenamiento, un 15 % para validación y un 10 % para testeo, obteniendo la siguiente distribución inicial tabla 10.

Tabla 10. Base de datos inicial para detección

<b>Clase.</b>	<b>Total imágenes.</b>
Datos Entrenamiento	433
Datos Validación	86
Datos test	59

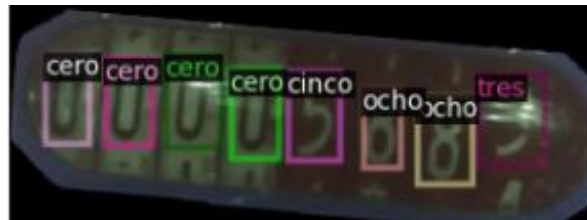
Se aplicó un aumento de datos y etiquetado a las 433 imágenes de entrenamiento utilizando la plataforma *Roboflow*. Las etiquetas se asignan en formato *YOLOv8*, este proceso consiste en asignar la etiqueta correspondiente a cada uno de los números del dial, figura 7. Las transformaciones aplicadas a las imágenes originales incluyeron una *rotación aleato-*

ria en un rango de  $-6^\circ$  a  $+6^\circ$ , un *desenfoque Gaussiano aleatorio* entre 0 y 1 píxel, y un *ruido equivalente al 1 % de los píxeles*. Después de realizar este aumento obtenemos una [Base de datos de detección YOLOv8](#) con su respectiva etiqueta para entrenar la red neuronal de detección con YOLOv8, la cantidad de imágenes obtenidas se muestran en la Tabla 11. Puede descargar la base de datos en el anexo 4.

Tabla 11. Base de datos aumentada para detección con YOLOv8 y Detectron2.

Clase.	Total imágenes etiquetadas.
Datos Entrenamiento	1300
Datos Validación	86
Datos test	59
<b>Total datos</b>	<b>1445</b>

Figura 7. Etiquetado para detección con Roboflow.



**5.2.2. Base de datos de detección para modelo Detectron2.** La base de datos utilizada para entrenar el modelo de detección con Detectron 2 está compuesta por un conjunto de imágenes que han pasado por el procedimiento de aumento de datos y la distribución de porcentaje mencionados en la sección 5.2.1. Estas imágenes tienen un formato RGB de  $640 \times 640$  píxeles y están etiquetadas en formato *coco*, obteniendo una [Base de datos de detección Detectron2](#) con su respectiva etiqueta para entrenar la red neuronal de detección con Detectron2. Tabla 11. Puede descargar la base de datos en el anexo 4.

### **5.3. ESTRUCTURA EN 3D.**

Para el desarrollo de este proyecto, se diseñó una estructura de fácil acople que no afecta el funcionamiento del contador de agua. Esta estructura cumple con la función de mantener los elementos electrónicos a una distancia óptima de 11 cm, lo cual es crucial para lograr un enfoque adecuado de la cámara y capturar todos los números del dial del contador. De esta manera, se asegura una toma precisa de los datos. Además, la estructura está diseñada para proteger los equipos electrónicos de las condiciones climáticas y del entorno en el que se encuentre el medidor de agua, brindando así una protección adecuada a los componentes del sistema.

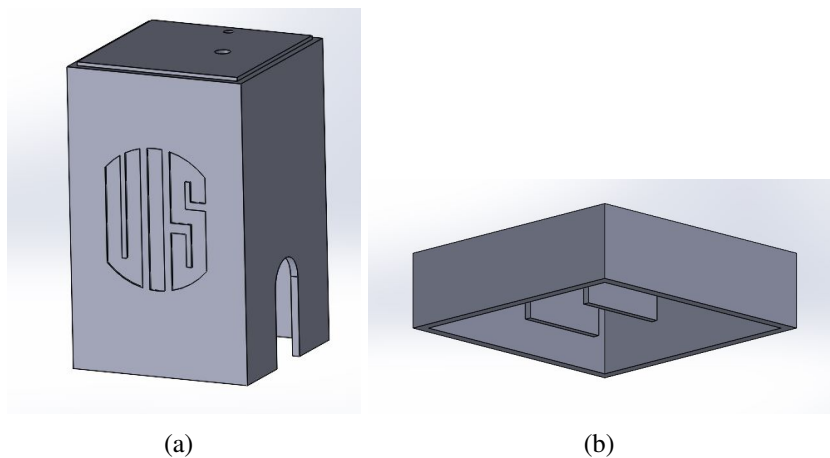
El modelado de la estructura se llevó a cabo utilizando el software de diseño asistido por computadora (*CAD*) *SolidWorks*, este software es ampliamente utilizado en la industria para el diseño en *3D*, permite optimizar los diseños de ingeniería, diseño industrial y fabricación, la licencia de este software fue concedida por la Universidad Industrial de Santander, a quien agradecemos por facilitarnos dicha herramienta ofimática para el desarrollo del modelado, durante el proceso se crearon varios modelos que se fueron modificando en apariencia y dimensiones permitiéndonos llegar a un producto final, este diseño final consta de cuatro piezas modeladas en *3D*, un cuerpo principal, un cuerpo secundario con su tapa para alojar los dispositivos de suministro de energía y una tapa de protección superior para proteger cables y microcontrolador. La impresión final de la estructura en *3D* se realizó a través de filamentos (*FDM*), colocando capas sucesivas de *PLA* con un factor de llenado del 80 % y cuyas dimensiones finales y planos a detalle se pueden observar en el repositorio de *GitHub* [Modelado\\_en\\_3D](#), anexo 3.

**5.3.1. Cuerpo principal.** Esta estructura se acopla de forma directa al contador de agua sin afectar su funcionamiento, su principal función es sostener los elementos electrónicos a distancias específicas que nos permitan la recopilación precisa de datos y asegura el soporte

de los componentes electrónicos en una sola estructura compacta. Al estar diseñada para encajar perfectamente con el contador de agua, garantiza la integridad del sistema y facilita la toma de datos de manera eficiente. Además, proporciona un soporte para los elementos electrónicos ayudando a mantener la estabilidad y protección de estos evitando posibles daños o movimientos no deseados.

El cuerpo principal de la estructura es una caja rectangular con un espesor de  $4\text{ mm}$  y dimensiones  $159\text{ mm} \times 98.2\text{ mm} \times 98.2\text{ mm}$ . En las caras laterales parte inferior, se encuentran dos orificios en forma de "U" con un diámetro de  $\frac{1}{2}$ ", los cuales coinciden con el diámetro de la tubería de entrada y salida del contador de agua. En la cara posterior de la caja, hay cuatro orificios de  $3\text{ mm}$  que se utilizan para la sujeción de la estructura secundaria, así como un orificio para el paso de los cables necesarios para las conexiones correspondientes. En el interior de la estructura, se ha instalado una tira de *LED* que proporciona una iluminación constante. En la cara frontal de la estructura, se ha agregado el logo institucional de la *UIS* en bajo relieve y en la parte superior de la estructura, se encuentran dos orificios, uno diseñado para el lente de la cámara *OV2640* de la *ESP32-Cam* y el otro destinado al paso de los cables de las diferentes instalaciones. Esto se puede ver en la figura 8a.

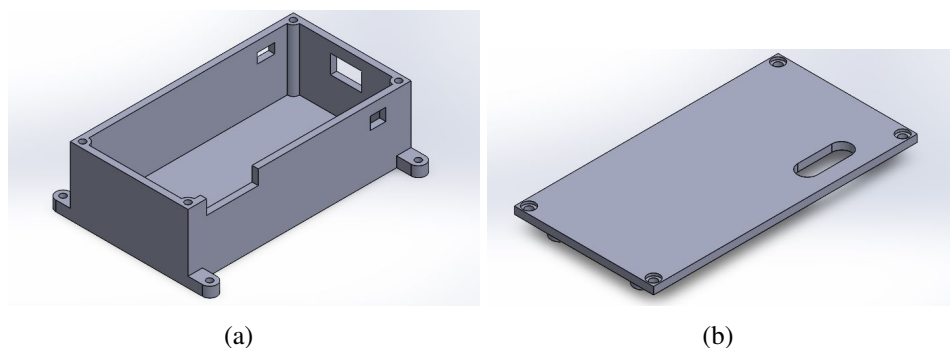
Figura 8. (a) Cuerpo principal. (b) Tapa superior.



**5.3.2. Tapa superior.** Esta estructura tiene como función principal proteger el microcontrolador y las conexiones ubicadas en la parte superior. Está compuesto por un cuadrado de medidas  $98.2\text{ mm} \times 98.2\text{ mm} \times 28\text{ mm}$ , el cual ha sido vaciado en su interior dejando un espesor de  $4\text{ mm}$  en las paredes. Además, se han diseñado dos paletas en la parte interna de la tapa para asegurar que el microcontrolador *ESP32-Cam* se mantenga en su lugar. De esta manera, se garantiza la seguridad y estabilidad del microcontrolador en la estructura para su correcto funcionamiento. El diseño de esta estructura se observa en la figura 8b.

**5.3.3. Cuerpo secundario.** Esta estructura se encarga de mantener la *Power Bank* y sus dos baterías *18650* de *Li-ion*, es una caja rectangular con dimensiones  $108\text{ mm} \times 78\text{ mm} \times 33\text{ mm}$ , esta estructura cuenta con orificios que permiten el acceso a los puertos de carga, así como la visualización de los *LED's* indicadores de carga, proporciona acceso al switch *on/off* y al botón de *reset* del sistema de alimentación. Esta estructura va adherida a la estructura principal por medio de cuatro tornillos de  $3\text{ mm}$  y está protegida con una tapa de medidas  $108\text{ mm} \times 57\text{ mm} \times 3\text{ mm}$ . La tapa cuenta con topes que evitan el contacto directo de las *power bank* con la estructura permitiendo una mejor refrigeración del sistema, como se observan en la figura 9.

Figura 9. (a) Cuerpo secundario. (b) Tapa cuerpo secundario.



## 6. IMPLEMENTACIÓN DE LAS REDES NEURONALES

En el presente Capítulo se describe el diseño de la propuesta de implementación de las redes neuronales usadas en este proyecto, está dividida en dos partes que corresponde a la segmentación y detección de los números en el dial de un contador de agua modelo *R200*. Además, se realiza una evaluación de los entrenamientos realizados para cada una de las etapas del proyecto. La propuesta se basa en un marco de dos etapas, en la primera etapa se segmenta el dial del contador dentro de la imagen utilizando el modelo *YOLOv8m*. En la segunda etapa se realiza la detección de los números presentes en el dial y para esta etapa se realiza una comparativa entre los modelos *Detectron2* y *YOLOv8* para la selección del modelo a implementar. El objetivo de esta propuesta es lograr una segmentación precisa del dial y una detección precisa de los números, lo cual es fundamental para el funcionamiento del sistema embebido.

Para entrenar y evaluar los modelos de detección y segmentación para la lectura del contador de agua, se utilizan conjuntos de datos etiquetados manualmente en *Roboflow* con imágenes del modelo de contador de agua *R200* generadas específicamente para el desarrollo de este proyecto donde se miden diferentes métricas de rendimiento para cuantificar el desempeño de los modelos, lo que contribuye a determinar su eficacia en la tarea de lectura del contador de agua.

La implementación de las redes neuronales se desarrolló utilizando *Google Colab*, un entorno de desarrollo en línea que permite ejecutar código de *Python* en la nube sin necesidad de tener una configuración local en el equipo. Basado en *Jupyter Notebook*, *Google Colab* proporciona una plataforma interactiva para crear y ejecutar código de manera eficiente. Es especialmente útil en proyectos de aprendizaje automático, ya que ofrece acceso gratuito a recursos potentes como unidades de procesamiento gráfico (*GPU*) y unidades de procesa-

miento tensorial (*TPU*). Además, cuenta con integración con *Google Drive*, lo que facilita el almacenamiento de datos de entrenamiento, validación y prueba en la nube. por estas razones se hace uso en esta investigación.<sup>29</sup>.

## 6.1. MODELO DE SEGMENTACIÓN.

La segmentación es un proceso que consiste en dividir una imagen en partes con el fin de separar las sección de interés del resto de la imagen, Su importancia radica en que permite extraer únicamente la información relevante para utilizarla como entrada en el entrenamiento de otras redes neuronales, esta técnica es útil en otras tareas de análisis y reconocimiento de imágenes ya que al dividir el conjunto de datos en segmentos más pequeños, se pueden procesar y cargar en memoria de forma más eficiente durante el entrenamiento de la red neuronal, esto contribuye en mejorar el rendimiento y la capacidad de generalización de los modelos al enfocarse en las regiones de interés y reducir la influencia de las áreas irrelevantes de la imagen <sup>30</sup>. Las *CNN* aprenden a segmentar imágenes observando muchos ejemplos de imágenes y sus correspondientes etiquetas.

**6.1.1. Entrenamiento red *YOLOv8*.** *YOLOv8*<sup>31</sup> es la última versión de la serie *YOLO* (*You Only Look Once*) introduce varias mejoras que permiten obtener un rendimiento mejorado en comparación con las versiones anteriores, emplea un mecanismo de detección sin anclaje, una red piramidal de funciones mejoradas y una función de pérdida modificada. Una

---

<sup>29</sup> Ekaba Bisong. “Google Colaboratory”. en. En: *Building Machine Learning and Deep Learning Models on Google Cloud Platform: A Comprehensive Guide for Beginners*. Ed. por Ekaba Bisong. Berkeley, CA: Apress, 2019, págs. 59-64. DOI: 10.1007/978-1-4842-4470-8\_7.

<sup>30</sup> Nora La Serna Palomino y Ulises Norberto Román Concha. “Técnicas de Segmentación en Procesamiento Digital de Imágenes”. En: *Revista de investigación de Sistemas e Informática* 6.2 (30 de dic. de 2009). Number: 2, págs. 9-16.

<sup>31</sup> Glenn Jocher, Ayush Chaurasia y Jing Qiu. *YOLO by Ultralytics*. Ver. 8.0.0. original-date: 2022-09-11T16:39:45Z. Ene. de 2023.

de las principales características de *YOLOv8* es su capacidad de realizar detección en una sola pasada por la imagen y calcula una puntuación de confianza para cada cuadro delimitador que contenga un objeto con la medida de similitud entre la realidad y el cuadro delimitador previsto <sup>32,33</sup>Para garantizar una detección más precisa, se utiliza un mecanismo de supresión no máxima. Este proceso descarta los cuadros delimitadores que se superponen y selecciona el cuadro con la mayor confianza, lo que ayuda a evitar duplicaciones y a obtener una salida más precisa.

Para realizar el proceso de segmentación se utilizó *YOLOv8*<sup>31</sup>. Se encontró que *YOLOv8m-seg* presentaba un equilibrio óptimo entre precisión y velocidad de inferencia<sup>34</sup> lo que lo convierte en el modelo adecuado para la tarea de segmentación. Para la configuración de los parámetros de entrenamiento de *YOLOv8m-seg* se realizaron pruebas previas para encontrar los valores óptimos que permitieran obtener los mejores resultados para la segmentación con los datos tomados en el ambiente controlado en que se desarrolló este proyecto. La Tabla 12 muestra los parámetros definidos para realizar el entrenamiento de la red neuronal en la tarea de segmentación, el código implementado se puede encontrar en *GitHub* [Entrenamiento\\_segmentación\\_YOLOv8](#), anexo 5.

***epochs***: Es el número de épocas o iteraciones completas que se utilizarán durante el entrenamiento.

***imgsz***: Es el tamaño de las imágenes de entrada durante el entrenamiento.

---

<sup>32</sup> Muhammad Hussain. “YOLO-v1 to YOLO-v8, the Rise of YOLO and Its Complementary Nature toward Digital Manufacturing and Industrial Defect Detection”. en. En: *Machines* 11.7 (jul. de 2023). Number: 7 Publisher: Multidisciplinary Digital Publishing Institute, pág. 677. DOI: 10 . 3390 / machines11070677.

<sup>33</sup> Andrei Dumitriu et al. “Rip Current Segmentation: A Novel Benchmark and YOLOv8 Baseline Results”. es. En: 2023, págs. 1261-1271.

<sup>34</sup> Ultralytics. *Segment*. es.

Tabla 12. Hiperparámetros para entrenamiento en segmentación modelo *YOLOv8m*.

Modelo	Clase	Parámetros	Valor
YOLOv8m	segmentación	epochs	300
		patience	50
		batch size	16
		imgsz	224
		optimizer	SGD
		weight_decay	0.0005
		lr0	0.01

***patience***: Es el número de épocas en las que se puede tolerar una falta de mejora en el rendimiento antes de detener el entrenamiento.

***optimizer***: Algoritmo de optimización utilizado durante el entrenamiento.

***weight decay***: Ayuda a prevenir el sobreajuste al penalizar los pesos más grandes en la función de pérdida.

***batch size***: Define el tamaño del lote utilizado durante el entrenamiento.

***lr0***: Establece la tasa de aprendizaje inicial utilizada por el optimizador durante el entrenamiento, la cantidad en la que los pesos del modelo se actualizan en cada paso durante el entrenamiento.

## 6.2. MODELOS DE DETECCIÓN.

Son algoritmos o arquitecturas de redes neuronales diseñadas para identificar y localizar objetos específicos en imágenes o vídeos. Estos modelos utilizan técnicas de aprendizaje profundo y visión por computadora para analizar y comprender el contenido de una imagen o vídeo, mediante aprendizaje profundo se pueden utilizar una variedad de técnicas para realizar la detección de objetos. Los enfoques populares basados en el aprendizaje profundo utilizan redes neuronales convolucionales (*CNN*), como los usados en este proyecto *YOLOv8* y *Detectron2* que aprenden automáticamente a detectar objetos dentro de las imágenes<sup>35</sup>.

<sup>35</sup> *What Is Object Detection?* URL: <https://la.mathworks.com/discovery/object-detection.html> (visitado 05-07-2023).

**6.2.1. Entrenamiento *Detectron2*.** *Detectron2* es un framework de código abierto desarrollado por *Facebook AI Research* que se utiliza para tareas de detección y segmentación de objetos en imágenes y vídeos. El parámetro de evaluación que se utiliza se denomina *Mean Average Precision (mAP)*. *Detectron2* se basa en una estructura modular y escalable que utiliza redes neuronales convolucionales (*CNN*), *Feature Pyramid Networks (FPN)* y *Region Proposal Networks (RPN)* para extraer características, generar propuestas de regiones y realizar inferencias de detección. Además, ofrece una amplia variedad de modelos pre entrenados que permite ajustar los parámetros según las necesidades de la tarea<sup>36,37</sup>.

Para *Detectron2* se utilizó el modelo *Faster R-CNN* con la arquitectura *X-101* y *FPN (Feature Pyramid Network)* entrenado con el conjunto de datos *COCO*. se entreno el modelo con el conjunto de datos de salida de segmentación. *Detectron2*<sup>38</sup> fue seleccionado para este proyecto debido a su flexibilidad, rendimiento y capacidad de personalización, lo que lo hace adecuado para la tarea de detección en la inferencia. Para la configuración de los parámetros se realizaron pruebas para determinar los valores óptimos que permitieran obtener los mejores resultados en la detección de los números del dial. La Tabla 13 muestra los valores utilizados para el entrenamiento. El código implementado se puede encontrar en *GitHub Entrenamiento\_detección\_Detectron2*, anexo 5.

**ITER:** Número máximo de iteraciones que se realizan durante el entrenamiento.

**LR:** Tasa de aprendizaje base para el entrenamiento.

**BATCH:** Es el número de imágenes por lote utilizado durante el entrenamiento.

**STEPS:** Establece los pasos en los que se reducirá la tasa de aprendizaje durante el entrena-

---

<sup>36</sup> Ritik Singh et al. “Helmet Detection Using Detectron2 and EfficientDet”. En: *2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT)*. 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT). Jul. de 2021, págs. 1-5. DOI: 10.1109/ICCCNT51525.2021.9579953.

<sup>37</sup> ¡Bienvenido a la documentación de detectron2! — documentación de detectron2 0.6. URL: <https://detectron2.readthedocs.io/en/latest/> (visitado 05-07-2023).

<sup>38</sup> Yuxin Wu et al. *Detectron2*. <https://github.com/facebookresearch/detectron2>. 2019.

Tabla 13. Hiperparámetros para entrenamiento en detección modelo *Detectron2*.

Modelo	Clase	Parámetros	Valor
<b>Detectron2</b>	Detección	ITER	3000
		BATCH	4
		LR	0.001
		STEPS	(1000, 3001)
		BATCH SIZE	64
		CLASSES	11
		PERIOD	500

miento.

**BATCH SIZE:** Tamaño del lote por imagen interés del modelo.

**CLASSES:** Número de categorías que el modelo debe detectar.

**PERIOD:** Cada cuantas iteraciones se muestran los resultados.

**6.2.2. Entrenamiento YOLOv8.** *YOLOv8*<sup>31</sup>. Se eligió *YOLOv8m* ya que este presentaba un equilibrio entre precisión y velocidad de inferencia<sup>31</sup> lo que lo convierte en el modelo adecuado para la tarea de detección de los números del dial del contador de agua modelo *R200*. Para el entrenamiento se utilizaron los parámetros que se muestran en la Tabla 14. El código implementado se puede encontrar en *GitHub* [Entrenamiento\\_detección\\_YOLOv8](#), anexo 5.

Tabla 14. Hiperparámetros para entrenamiento en detección modelo *YOLOv8*.

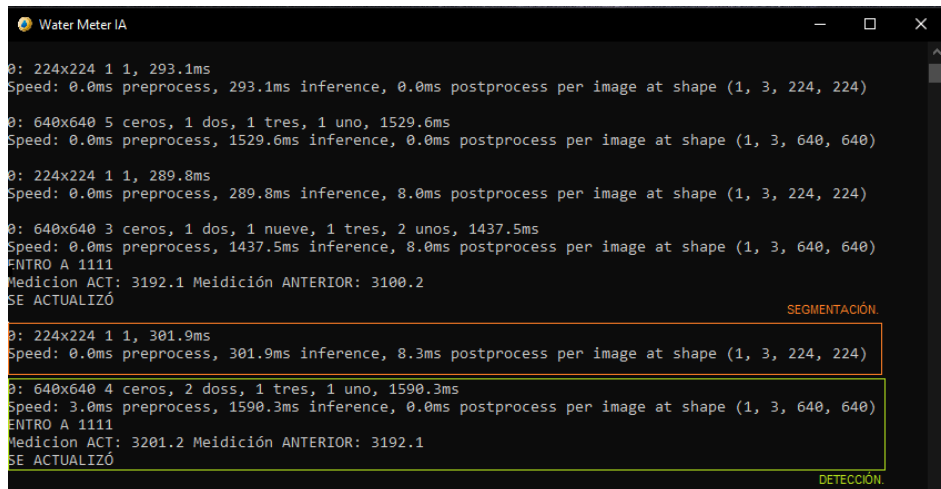
Modelo	Clase	Parámetros	Valor
<b>YOLOv8</b>	Detección	epochs	300
		patience	50
		batch size	16
		imgsz	640
		optimizer	SGD
		weight_decay	0.0005
		lr0	0.01

## 7. INTERFAZ GRÁFICA

El proyecto cuenta con una interfaz gráfica desarrollada utilizando *PyQt5* en conjunto con *QT Designer*. *PyQt5* es una biblioteca de *Python* que permite crear de manera sencilla y eficiente aplicaciones de escritorio con interfaces gráficas de usuario (*GUI*). Proporciona una amplia gama de funcionalidades para desarrollar aplicaciones interactivas con interfaces gráficas. Durante el desarrollo de la aplicación, se utilizaron diversas bibliotecas, entre las cuales se destaca *Pandas*, *openpyxl*, *datetime*, *sys*, entre otras librerías usadas para el desarrollo de la aplicación. Los códigos correspondientes a la interfaz se encuentran disponible en el [GitHub Código\\_Interfaz](#), anexo 5.

Cuando se inicia aplicación, se realiza el procesamiento de las imágenes ubicadas en la carpeta del servidor local FTP. Este proceso puede ser observado en una ventana emergente de la línea de comandos (cmd), como se muestra en la figura 10. En esta ventana, se visualiza la información de compilación del proceso de segmentación y posteriormente el proceso de detección de cada una de las imágenes presentes en la carpeta del servidor local. Durante el procesamiento, se realiza una verificación para asegurar que los números identificados sean iguales o mayores a la inferencia anterior. En caso de cumplir esta condición, los datos son procesados y guardados en la base de datos para su posterior presentación en la interfaz. Si no se cumple esta condición, se descartan el valor y se vuelve a procesar la misma inferencia hasta encontrar un valor coherente.

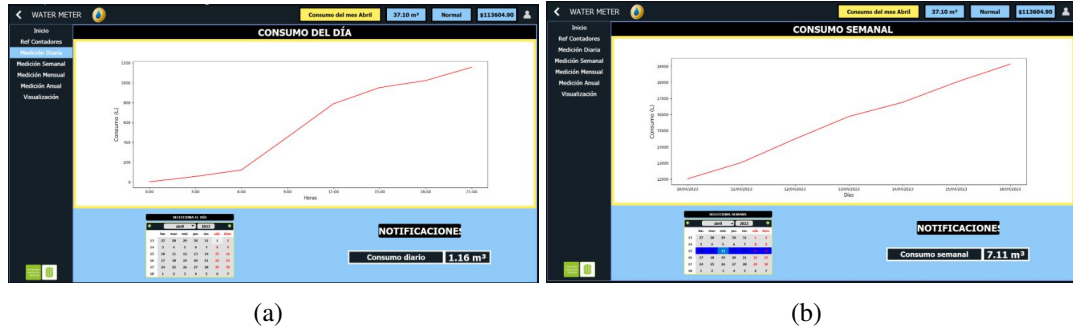
Figura 10. Procesamiento de las imágenes guardadas en el servidor local.



Para brindar una experiencia agradable, cómoda e intuitiva al usuario, la interfaz cuenta con un sistema de búsqueda de fechas por calendario y un menú lateral animado que ofrece diversas funcionalidades. Al desplegar este menú se puede acceder a la sección de medición diaria se presenta una gráfica lineal del consumo del día seleccionado, la gráfica se genera con los datos capturados cada tres horas y el consumo se muestra en litros de agua. Además, se muestra una notificación que indica el consumo total del día seleccionado en metros cúbicos, Figura 11a. En la sección de medición semanal se observa una gráfica lineal generada con los datos acumulados de los días correspondientes a la semana y el consumo se indica en litros. Se muestra una notificación que indica el consumo total de la semana en metros cúbicos. Figura 11b.

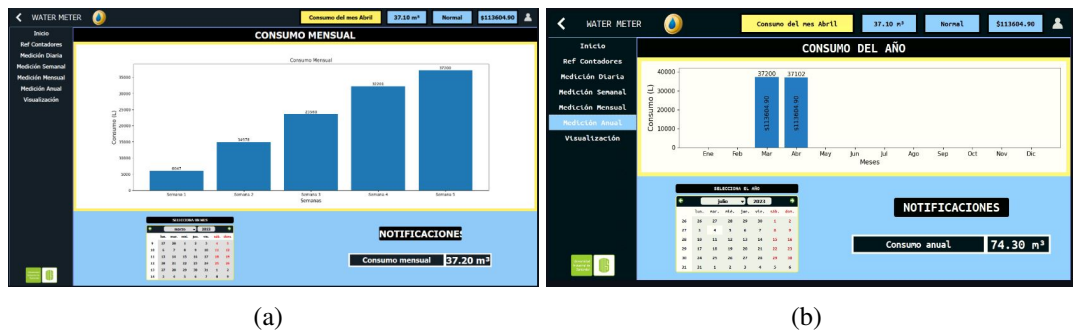
La sección de medición mensual presenta una gráfica de barras que indica el consumo acumulado de todas las semanas correspondientes al mes seleccionado. Además, se muestra una notificación que indica el consumo total del mes en metros cúbicos brindando una referencia de la cantidad de agua utilizada durante ese periodo, en la parte superior de la interfaz, se visualiza el consumo en metros cúbicos del mes actual, también se incluye un indicador que presenta si el consumo se encuentra dentro de los límites promedios del usuario y un valor

Figura 11. (a) Consumo Diario. (b) Consumo Semanal.



estimado a cancelar por el servicio de acueducto calculado en base a la tabla de tarifas del servicio establecido por el *AMB S.A. E.S.P*<sup>24</sup>. Figura 12a. En la sección de medición anual, se encuentra una gráfica de barras que presenta el consumo en litros para cada uno de los meses del año junto al valor aproximado a pagar por el servicio de acueducto. Además, se muestra una notificación que indica el consumo total del año en metros cúbicos, mediante esta gráfica el usuario puede identificar patrones y tendencias en el consumo de agua y tener una visión global del comportamiento de consumo a lo largo del año. Figura 12b.

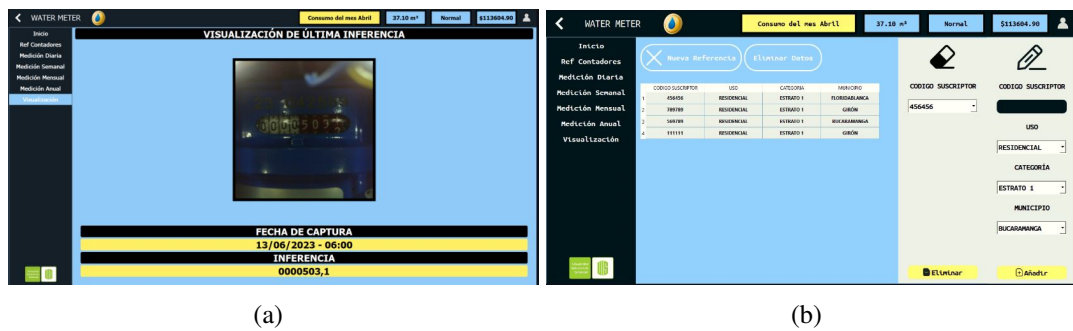
Figura 12. (a) Consumo Mensual. (b) Consumo Anual.



En la sección visualización se observa la última imagen capturada, junto con la fecha y hora en que fue tomada, además, presenta el valor identificado por los modelos de inteligencia artificial entrenados en este proyecto. Figura 13a. Esta sección ofrece al usuario una visuali-

zación instantánea de los resultados obtenidos a partir de la inferencia de la última imagen, brindando al usuario una referencia rápida y actualizada sobre el estado del sistema. En la sección Referencia de Contadores se presenta una tabla que permite agregar o eliminar usuarios al sistema. Los datos que se pueden ingresar corresponden al código del suscriptor, tipo de uso, estrato socioeconómico y municipio. Figura 13b. Esta funcionalidad se propone como una herramienta para llevar un control de los usuarios que utilizan el aplicativo en futuros trabajos.

Figura 13. (a) Ultima Inferencia. (b) Datos del Usuario.



## 8. RESULTADOS

Este capítulo presenta los resultados obtenidos durante el desarrollo del proyecto, se expone en dos etapas correspondientes a la segmentación y detección.

### 8.1. DESEMPEÑO RED NEURONAL *YOLOv8m-seg* SEGMENTACIÓN.

En la evaluación del modelo de segmentación con *Deep Learning*, se utilizó *YOLOv8m-seg* para evaluar su desempeño se presentan las métricas obtenidas. Entre estas destacan el *AP* (*Average Precision*) con umbral del 50 % y el *mAP* (*Mean Average Precision*) en un rango de umbrales del 50 % al 95 %. Estos resultados revelan la capacidad del modelo para realizar una segmentación precisa del dial del contador en las imágenes brindando información detallada sobre su forma y contorno. Esto resulta útil para eliminar áreas no importantes del contador y mejorar la eficiencia de la detección. En la tabla 15 se presentan las métricas obtenidas con los datos de validación compuestos por 116 imágenes. Para el proceso de entrenamiento se establecieron 300 épocas, pero se detuvo en la época 177 debido a la falta de mejoría en las últimas 50 épocas, siguiendo los parámetros de entrenamiento establecidos para evitar el sobreajuste. El tiempo total de entrenamiento requerido fue de 1,057 horas.

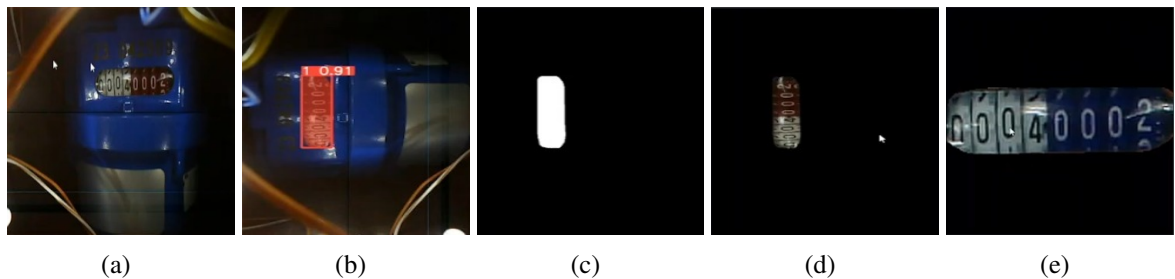
Tabla 15. Resultados de segmentación *YOLOv8*.

métrica	mAP 50	mAP 50-95
<i>YOLOv8m-seg</i>	0.995	0.925

La figura 14 presenta el proceso de segmentación realizado a las imágenes, inicialmente se indica el área de interés por medio de una etiqueta figura (b), luego se genera la máscara teniendo en cuenta la etiqueta del área de interés figura (c), se realiza la resta de la imagen original y la máscara generada obteniendo el área de interés (dial del contador) como se observa en la figura (d), por último se realizan unas transformaciones como recorte, rotación

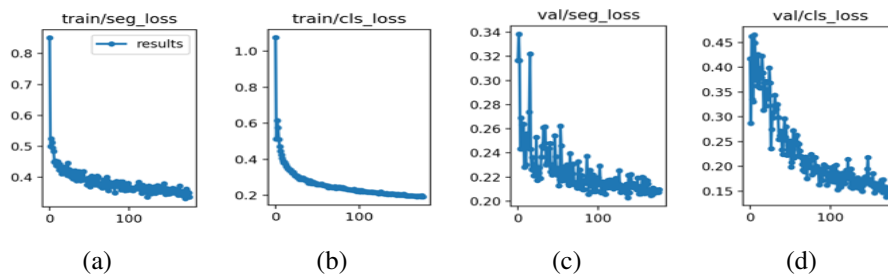
y redimensionamiento para obtener la figura (e) que corresponde a la imagen entrada para el proceso de detección.

Figura 14. Proceso de segmentación.: (a) Imagen tomada del contador. (b) Etiquetado para segmentación. (c) Mascara para segmentación. (d) Imagen segmentada. (e) Procesamiento de la imagen segmentada.



Para evaluar el desempeño del modelo se generaron curvas de pérdidas, las cuales permiten visualizar la evolución de la pérdida, durante el entrenamiento del modelo a lo largo de las diferentes épocas. La pérdida es una medida que indica que tan lejos están las predicciones del modelo de los valores reales o esperados. En la Figura 15 se presentan las curvas de pérdida obtenidas por *YOLOv8* en la tarea de segmentación, en las etapas de entrenamiento y validación respectivamente.

Figura 15. Gráficas de la función de pérdida modelo *YOLOv8* para segmentación.: (a) Gráfica de la función de pérdida de clasificación para entrenamiento. (b) Gráfica de la función de pérdida de localización para entrenamiento. (c) Gráfica de la función de pérdida de clasificación para validación. (d) Gráfica de la función de pérdida de localización para validación.



## 8.2. COMPARATIVA ENTRE YOLOv8 Y DETECTRON2 EN DETECCIÓN.

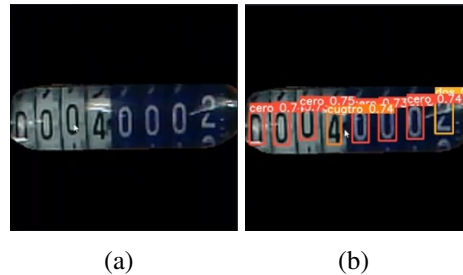
En este proyecto se realiza la comparación del desempeño y precisión de los modelos *YOLOv8* Y *detectron2* en la tarea de detección de los números en un contador de agua. La tabla 16 muestra la comparación entre las métricas de estudio para el desempeño de los modelos descritos y permite determinar el porcentaje de acierto con cada una de las 10 clases de entrenamiento (*números del 0 al 9*), junto con el valor del *mAP* (*Mean Average Precision*) en un rango de umbrales del 50 % al 95 % que nos indica el rendimiento del sistema en términos de precisión, de dicha tabla podemos observar que el modelo *YOLOv8* presenta mayor porcentaje de acierto en comparación con *detectron2*, obteniendo mejores resultados en siete de las diez (10) clases y un *mAP(50-95)* superior, por esta razón se eligió *YOLOv8* para su implementación.

Tabla 16. Comparación entre *YOLOv8* y *Detectron2*.

Clase	métricas	Modelo	
		YOLOv8	Detectron2
	Cero	62.10 %	57.62 %
	Uno	39.70 %	40.94 %
	Dos	56.60 %	60.11 %
	Tres	67.20 %	53.37 %
	Cuatro	59.10 %	54.47 %
Detección	Cinco	61.30 %	52.63 %
	Seis	50.70 %	54.46 %
	Siete	53.80 %	43.71 %
	Ocho	60.40 %	55.66 %
	Nueve	59.80 %	53.14 %
	mAP	57.10 %	52.61 %

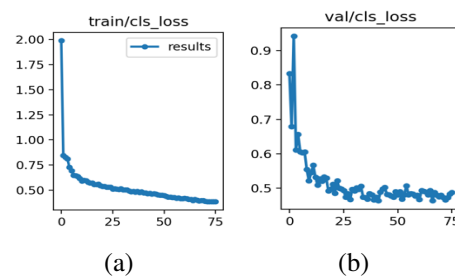
La figura 16 presenta el proceso de detección realizado a las imágenes. Se puede observar que la imagen de entrada al proceso de detección corresponde a la imagen segmentada obtenida en el proceso anterior, como se muestra en la figura (a). Para iniciar con el proceso de detección se identifican las áreas de interés que corresponden a cada uno de los números del dial y se les asignan etiquetas, posteriormente se identifican los números dentro de estas áreas como se observa en la figura (b). Los datos identificados se guardan en la base de datos para su posterior presentación en la interfaz gráfica.

Figura 16. Proceso de detección.: (a) Imagen tomada del contador. (b) Etiquetado para detección.



La figura 17 presenta los resultados de las curvas de pérdida para el modelo *YOLOv8* en la tarea de detección para las etapas de entrenamiento y validación respectivamente. Para el proceso de entrenamiento de *YOLOv8* en detección se establecieron 300 épocas, sin embargo se detuvo en la época 76 para evitar el sobreajuste, el entrenamiento demoró 1,406 horas. La figura 17 muestra el resultado del modelo de detección en las imágenes.

Figura 17. Gráficas de la función de pérdida modelo *YOLOv8* para detección.: (a) Gráfica de la función de pérdida de clasificación para entrenamiento. (b) Gráfica de la función de pérdida de clasificación para validación.



***traincls\_loss***: (*pérdida de clasificación*) representa la pérdida o error asociado con la clasificación de los objetos detectados por el modelo.

***train/df\_loss***: "*Detection, Localization, and False positives*", se refiere a los aspectos relacionados con la detección, localización y la cantidad de falsos positivos en la predicción de objetos.

## 9. RECOMENDACIONES

A partir de la experiencia adquirida en el desarrollo de este trabajo, se presentan algunas recomendaciones con el objetivo de mejorar la calidad del trabajo y su aplicabilidad en futuros proyectos.

- En la interfaz gráfica, se recomienda incorporar opciones de ajuste de parámetros-variables que permitan al usuario personalizar la visualización de la información según sus necesidades. Algunos de los parámetros recomendados son: la frecuencia de captura de datos, los límites considerados normales de consumo de agua y una alarma para indicar niveles bajos de la batería. Estos ajustes brindarán a los usuarios la capacidad de maximizar el uso de la información presentada y realizar un seguimiento más efectivo de su consumo de agua. Además, se sugiere implementar estos cambios en la interfaz de manera fácil e intuitiva para que los usuarios puedan realizar los ajustes de forma sencilla.
- Se sugiere incorporar el sistema *OTA (Over-the-Air)* en futuras actualizaciones del sistema, esto permitirá realizar actualizaciones de forma inalámbrica y sin intervenciones en la estructura existente. Mediante el uso del sistema *OTA*, se pueden realizar actualizaciones frecuentes en los parámetros relevantes del sistema, como la frecuencia de captura de datos, el rango para identificar consumos anormales de agua, entre otros parámetros relevantes. Esto brinda la posibilidad de mantener el sistema actualizado de manera ágil y eficiente, sin la necesidad de intervenciones físicas en los dispositivos.
- Para lograr un entrenamiento más efectivo de las redes neuronales implementadas, se recomienda aumentar el tamaño de la base de datos utilizada para el entrenamiento. Al disponer de un conjunto de datos más amplio, se espera mejorar las métricas y el rendimiento de cada una de las redes neuronales.

## 10. CONCLUSIONES

En este proyecto, se lograron alcanzar los objetivos propuestos utilizando técnicas de aprendizaje profundo y procesamiento digital de imágenes por medio de una cámara e inteligencia artificial.

- Se generaron dos bases de datos con imágenes etiquetadas de un contador de agua modelo *R200*. Estos conjuntos de datos con sus respectivas etiquetas se utilizaron para entrenar y evaluar los modelos de aprendizaje profundo con arquitectura de red neuronal convolucional propuestos en este proyecto.
- Durante el desarrollo del proyecto, se diseñó una estructura que asegura la colocación adecuada del sistema embebido y la cámara *OV2640*, garantizando una captura óptima de imágenes del contador de agua. Esta estructura se diseñó considerando la facilidad de instalación y desmontaje. Además, se tuvo en cuenta la seguridad y estabilidad de la estructura para garantizar un posicionamiento estable de la cámara durante el proceso de toma de datos.
- Realizar la segmentación previamente es una buena práctica que ayuda a mejorar la eficiencia y precisión en la detección de objetos. Al segmentar primero la imagen se eliminan las regiones no relevantes reduciendo la cantidad de datos a procesar, lo que lleva a una detección rápida y precisa debido a que se enfoca en las áreas de interés reduciendo el ruido y las interferencias, mejorando la calidad de los resultados de detección de los modelos de aprendizaje profundo.
- En la primera fase de este proyecto, se logró implementar con éxito la segmentación mediante el modelo *YOLOv8*. Durante las pruebas y evaluaciones realizadas el modelo demostró su rendimiento siendo capaz de identificar y segmentar de manera precisa el dial del contador, eliminando eficazmente las regiones no relevantes en la imagen. Se

obtuvo una precisión del 96.5 % estos resultados demuestran la eficiencia y efectividad del modelo en la tarea de segmentación de las imágenes.

- En la segunda fase de este proyecto, se logró implementar con éxito la detección mediante dos modelos *YOLOv8* y *detectron2*. Se creó un conjunto de imágenes etiquetadas para el entrenamiento de cada uno de ellos, el resultado del test obtuvo una precisión del 57.1 % y 52.61 % respectivamente, mostrando mejores resultados de desempeño en la tarea de detección con *YOLOv8*, por esta razón es el modelo elegido y aplicado en este proyecto.
- Se desarrolló una aplicación de escritorio práctica y de fácil uso que integra las técnicas de detección y segmentación. Esta herramienta permite realizar dichas tareas de manera automatizada y eficiente. Los datos capturados por la *ESP32-Cam* son enviados a un servidor local a través del protocolo *FTP* para su posterior análisis y presentación brindando al usuario una visualización fácil y didáctica del consumo de agua.

En conclusión, a través de este proyecto se logró desarrollar un primer prototipo para el monitoreo del consumo de agua dedicado al usuario, aplicando técnicas de segmentación y detección de objetos por medio del análisis de imágenes. Los resultados obtenidos demuestran una buena precisión en las técnicas implementadas, lo que puede contribuir a trabajos futuros en las áreas de monitoreo inteligente del consumo de servicios básicos en contadores.

## BIBLIOGRAFÍA

- Acto de gerencia tarifas 2023*. <https://www.amb.com.co:8444/DocumentoInfo/actotarifas2023.pdf>. (Accessed on 28/04/2023). Abr. de 2023 (vid. págs. 36, 65).
- Bisong, Ekaba. “Google Colaboratory”. en. En: *Building Machine Learning and Deep Learning Models on Google Cloud Platform: A Comprehensive Guide for Beginners*. Ed. por Ekaba Bisong. Berkeley, CA: Apress, 2019, págs. 59-64. DOI: 10.1007/978-1-4842-4470-8\_7 (vid. pág. 58).
- Ciresan, Dan C et al. *Flexible, High Performance Convolutional Neural Networks for Image Classification*. <https://people.idsia.ch/~juergen/ijcai2011.pdf>. (Accessed on 03/05/2023) (vid. pág. 19).
- Colombia, Oracle. *¿Qué es la inteligencia artificial (IA)?* URL: <https://www.oracle.com/co/artificial-intelligence/what-is-ai/> (visitado 04-02-2023) (vid. pág. 18).
- Comparación de batería de densidad de energía: celdas cilíndricas y prismáticas*. 2023. URL: <https://www.epectec.com/batteries/cell-comparison.html> (visitado 03-04-2023) (vid. pág. 30).
- Consumo básico de agua potable en Colombia, author = M, Gustavo Chacón and Lizcano, Imelda and Lara, Yefer Asprilla,* | *Tecnogestión: Una mirada al ambiente*. es-ES. Feb. de 2016. URL: <https://revistas.udistrital.edu.co/index.php/tecges/article/view/4379> (visitado 07-06-2023) (vid. pág. 37).

*Diferentes niveles de voltaje de la placa del microcontrolador ESP32.* Section: Arduino. 29 de mayo de 2021. URL: <https://elosciloscopio.com/mejor-bateria-para-esp32/> (visitado 03-04-2023) (vid. pág. 30).

Dumitriu, Andrei et al. “Rip Current Segmentation: A Novel Benchmark and YOLOv8 Baseline Results”. es. En: 2023, págs. 1261-1271 (vid. pág. 59).

*ESP32 CAM introducción y primeros pasos.* Section: ESP32. 18 de ene. de 2022. URL: <https://programarfácil.com/esp32/esp32-cam/#:~:text=ESP32%2DCAM%2C%20es%20un%20dispositivo,podremos%20almacenar%20fotos%20o%20videos.> (vid. pág. 28).

*ESP32-CAM Product Specification.* 2023. URL: <https://loboris.eu/ESP32/ESP32-CAM%20Product%20Specification.pdf> (visitado 03-04-2023) (vid. pág. 28).

FUNDACIÓN, AQUAE. *¿Cuánta agua potable hay en la Tierra?* URL: <https://www.fundacionaquae.org/wiki/cantidad-de-agua-potable-fuente-de-vida/> (visitado 04-02-2023) (vid. pág. 17).

González, Ligdi. *¿Qué es el Perceptrón? Perceptrón Simple y Multicapa.* Aprende IA. 5 de oct. de 2021. URL: <https://aprendeia.com/que-es-el-perceptron-simple-y-multicapa/> (visitado 05-10-2021) (vid. pág. 25).

Géron, Aurélien. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow.* 3.<sup>a</sup> ed. O'Reilly Media, Inc, 2022. 1196 págs. (vid. págs. 20-22).

Gómez González, Sergio. *SolidWorks®.* 2008. URL: <https://elibro-net.bibliotecavirtuales.edu.co/es/ereader/uis/45924?page=18> (visitado 03-06-2023) (vid. pág. 33).

Hussain, Muhammad. “YOLO-v1 to YOLO-v8, the Rise of YOLO and Its Complementary Nature toward Digital Manufacturing and Industrial Defect Detection”. en. En: *Machines* 11.7 (jul. de 2023). Number: 7 Publisher: Multidisciplinary Digital Publishing Institute, pág. 677. DOI: 10.3390/machines11070677 (vid. pág. 59).

iAgua, redaccion. *¿Qué es un contador de agua y cuántos tipos hay?* iAgua. Publisher: iAgua. 11 de mayo de 2021. URL: <https://www.iagua.es/respuestas/que-es-contador-agua-y-cuantos-tipos-hay> (visitado 06-07-2023) (vid. págs. 40-42).

Jocher, Glenn, Ayush Chaurasia y Jing Qiu. *YOLO by Ultralytics*. Ver. 8.0.0. original-date: 2022-09-11T16:39:45Z. Ene. de 2023 (vid. págs. 58, 59, 62).

Martín, Daniel Moreno. *Estudio y evaluación de comunicaciones Wi-Fi basadas en IEEE 802.11ac*. (Accessed on 03/03/2023). Oct. de 2017. URL: <https://upcommons.upc.edu/bitstream/handle/2117/110125/memoria.pdf?sequence=1&isAllowed=y> (vid. pág. 31).

*Medidor volumétrico metálico preequipado R200 calibrado 1/2"*. es. URL: <https://controlagua.com/producto/medidor-volumetrico-metalico-preequipado-r200-calibrado-1-2/> (visitado 12-06-2023) (vid. pág. 43).

NetApp. *¿Qué es la IA y por qué es importante?* | NetApp. Section: AI. URL: <https://www.netapp.com/es/artificial-intelligence/what-is-artificial-intelligence/> (visitado 04-02-2023) (vid. pág. 18).

Ng, Kiat Siong, Pei-Yin Chen y Yuan-Chi Tseng. “A design of automatic water leak detection device”. En: *2017 IEEE 2nd International Conference on Opto-Electronic Information Processing (ICOIP)*. 2017 IEEE 2nd International Conference on Opto-Electronic Infor-

mation Processing (ICOIP). Jul. de 2017, págs. 70-73. DOI: 10.1109/OPTIP.2017.8030701 (vid. pág. 13).

*Norma Técnica Interna de Medidores - amb - Acueducto Metropolitano de Bucaramanga.* es. <https://www.amb.com.co/amb/conoce-amb/nuestros-procesos/norma-tecnica-interna-de-medidores/>. Feb. de 2022 (vid. págs. 40, 43).

Olabe, Xabier Basogain. *REDES NEURONALES ARTIFICIALES Y SUS APLICACIONES.* Bilbao: Escuela Superior de Ingeniería de Bilbao, EHU (vid. págs. 23, 24, 26).

*OV2640/OV7171 CMOS UXGA: (1600 x 1200) SVGA: (800 x 600) CameraChip™ Implementation Guide - MIT.* 2023. URL: [https://www.uctronics.com/download/cam\\_module/OV2640DS.pdf](https://www.uctronics.com/download/cam_module/OV2640DS.pdf) (visitado 03-04-2023) (vid. págs. 29, 30).

*OV7670/OV7171 CMOS VGA (640x480) CameraChip™ Implementation Guide - MIT.* 2023. URL: <https://web.mit.edu/6.111/www/f2016/tools/OV7670app.pdf> (visitado 03-04-2023) (vid. pág. 29).

Palomino, Nora La Serna y Ulises Norberto Román Concha. “Técnicas de Segmentación en Procesamiento Digital de Imágenes”. En: *Revista de investigación de Sistemas e Informática* 6.2 (30 de dic. de 2009). Number: 2, págs. 9-16 (vid. pág. 58).

Pedro Larrañaga Iñaki Inza, Abdelmalik Moujahid. *Tema 8. Redes Neuronales.* <http://www.sc.ehu.es/ccwbayes/docencia/mmcc/docs/t8neuronales.pdf>. (Accessed on 11/02/2023) (vid. pág. 18).

Pérez-Aguilar, Daniel Alexis, Redy Henry Risco-Ramos y Luis Casaverde-Pacherrez. *Sistemas embebidos y su aportación a la industrias.* Oasys. Section: Industria 4.0. URL: <https://oasys-sw.com/sistemas-embebidos-industria/> (visitado 02-04-2023) (vid. pág. 27).

*Qué es la impresión 3D, importancia y cómo funciona la fabricación aditiva.* Section: BLOG INDUSTRIA 4.0. 12 de feb. de 2021. URL: <https://www.cursosaula21.com/que-es-la-impresion-3d/> (visitado 03-04-2023) (vid. págs. 33, 34).

*Redes Neuronales Artificiales: El Perceptrón Multicapa.* Vrelectroniq. (Accessed on 05/16/2023). 21 de jul. de 2020. URL: <https://vrelectroniq.wixsite.com/vrelectroniq/post/redes-neuronales-artificiales-el-perceptron-multicapa> (visitado 03-06-2023) (vid. pág. 25).

SCIENCE, DATA. *Redes Neuronales Convolucionales – la manera ELI5.* 2020. URL: <https://datascience.eu/es/vision-artificial/redes-neuronales-convolucionales/> (visitado 20-02-2023) (vid. pág. 20).

Singh, Ritik et al. “Helmet Detection Using Detectron2 and EfficientDet”. En: *2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT)*. 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT). Jul. de 2021, págs. 1-5. DOI: 10.1109/ICCCNT51525.2021.9579953 (vid. pág. 61).

T, Neha. *Difference Between Bluetooth and Wifi (with Comparison Chart)*. Tech Differences. 16 de ago. de 2017. URL: <https://techdifferences.com/difference-between-bluetooth-and-wifi.html> (vid. pág. 31).

Ultralytics. *Segment*. es (vid. pág. 59).

*What Is Object Detection?* URL: <https://la.mathworks.com/discovery/object-detection.html> (visitado 05-07-2023) (vid. pág. 60).

Wu, Yuxin et al. *Detectron2*. <https://github.com/facebookresearch/detectron2>. 2019 (vid. pág. 61).

Yasar, Kinza. *What is 802.11n? | Definition from TechTarget*. Mobile Computing. (Accessed on 03/03/2023). URL: <https://www.techtarget.com/searchmobilecomputing/definition/80211n> (vid. pág. 31).

*¡Bienvenido a la documentación de detectron2! — documentación de detectron2 0.6*. URL: <https://detectron2.readthedocs.io/en/latest/> (visitado 05-07-2023) (vid. pág. 61).

## ANEXOS

### **Anexo A. Norma Técnica Interna de Medidores de Agua.**

El presente anexo describe a detalle la Norma Técnica Interna de Medidores de Agua de el *Acueducto Metropolitano de Bucaramanga S.A. E.S.P* , la cual nos ofrece un informe detallado de los requerimientos básicos necesarios para la selección, instalación y /o cambio de contadores de agua en los municipios de Bucaramanga, Floridablanca y Girón.

Las especificaciones expuestas en esta norma técnica se tomaron en consideración para la selección del modelo de contador de agua elegido para el desarrollo de este proyecto.

## **Anexo B. Especificaciones Técnicas y de Calibración de Contador de Agua.**

En este anexo se presentan las especificaciones de calibración del contador de agua utilizado en este proyecto y se compara con las especificaciones técnicas requeridas por el *Acueducto Metropolitano de Bucaramanga S.A. E.S.P.* Además, se adjunta el certificado de calibración de medidor emitido por el laboratorio *Hidrométrica S.A.* acreditado por la *ONAC ISO/IEC 17025:2017 12-LAC-031*, esta información valida el cumplimiento de las normativas establecidas por el *AMB S.A. E.S.P.* y demuestra la confiabilidad de las mediciones realizadas por el contador de agua utilizado en el proyecto.

### **Anexo C. Planos de la Estructura de Fácil Acople.**

En este anexo se encuentran los planos correspondientes a la estructura de fácil acople, diseñada para facilitar la toma de datos y proporcionar un soporte adecuado a los elementos electrónicos utilizados en este proyecto. Estos planos ofrecen una representación gráfica precisa de la estructura y de cada una de las piezas que la componen, mostrando las dimensiones, detalles y distribución de cada uno de sus componentes. Estos planos son una herramienta útil para comprender claramente el propósito de cada una de las piezas y realizar un correcto ensamble de la estructura.

#### **Anexo D. Bases de datos.**

En este anexo se encuentran las bases de datos creadas para el desarrollo de este proyecto, se incluyen los conjuntos de datos etiquetados para el entrenamiento, validación y prueba de cada una de las arquitecturas de redes neuronales implementadas, este recurso se comparte con el objetivo de verificar la transparencia de los resultados obtenidos en este proyecto. Además, de compartir la información recopilada para aquellos interesados en utilizarla para futuros estudios.

### **Anexo E. Repositorio de GitHub.**

Para recopilar toda la información generada durante el desarrollo de este proyecto, se creó un repositorio en *GitHub* donde se encuentra disponible toda la información adicional y recursos relevantes como documentos, imágenes, códigos y diseños que respaldan y complementan el desarrollo del proyecto. Además, se han incluido comentarios y notas explicativas para mejorar la comprensión del contenido en las carpetas principales. Este recurso se realizó para aquellos interesados en explorar más a fondo el trabajo realizado, los archivos disponibles en el repositorio de *GitHub* se pueden visualizar en línea y se pueden descargar, lo que facilita su consulta y acceso por parte de cualquier persona interesada en profundizar el proyecto.