UNIVERSITY OF THE PHILIPPINES

B.S Computer Engineering

Theodore J. Austria
Karl Vincent M. Espiritu

Computer Vision Based System for Automated Water Meter Reading

Undergraduate Project Adviser:

Nestor Michael Tiglao, PhD

Electrical and Electronics Engineering Institute

University of the Philippines Diliman

Undergraduate Project Reader:

Rhandley D. Cajote, PhD

Electrical and Electronics Engineering Institute

University of the Philippines Diliman

Date of Submission

11 March 2022

Permission is given for the following people to have access to this thesis:

Circle one or more concerns: I P C	
Available to the general public	Yes/No
Available only after consultation with author/thesis adviser	Yes/No
Available only to those bound by confidentiality agreement	Yes/No

Students' signature/s:

Signature/s of undergraduate project advisers:

University Permission Page

I hereby grant the University of the Philippines non-exclusive worldwide, royalty-free license to reproduce, publish, and public distribute copies of this work in any form subject to the provisions of applicable laws, the provisions of the UP IPR policy and any contractual obligations, as well as more specific permission marking on the Title Page.

Specifically I grant the following rights to the University:

- to upload a copy of the work in the theses database of the college/school/institute/department and in any other databases available on the public internet;
- to publish the work in the college/school/institute/department journal, both in print and electronic or digital format and online; and
- to give open access to above-mentioned work, thus allowing "fair-use" of the work in accordance with the provisions of the Intellectual Property Code of the Philippines (Republic Act No. 8293), especially for teaching, scholarly, and research purposes.

Theodore Austria — 11 March 2022

Karl Vincent Espiritu — 11 March 2022

Approval Sheet

In partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Engineering, this project entitled "Computer Vision Based System for Automated Water Meter Reading", prepared and submitted by Theodore Austria and Karl Vincent Espiritu, is hereby recommended for approval.

Nestor Michael C. Tiglao, PhD
Adviser

11 March 2022

Accepted in partial fulfillment of the requirements for the degree of [Course].

Rhandley D. Cajote, PhD
Panel Member

11 March 2022

Michael Angelo A. Pedrasa, PhD
Director, Electrical and Electronics Engineering Institute

11 March 2022

Abstract

Collecting readings from water meters is being done by sending out personnel from different water companies to different residential areas for them to manually record what they see on the water meter. Due to the nature of this being done manually, problems arise such as human errors, scheduling problems, or currently the COVID-19 pandemic. Thus, the researchers developed a model that utilizes computer vision to automate water meter readings remotely. The readings are displayed in an Internet of Things(IoT) dashboard and are seen by the consumers via a website dashboard. This way, consumers can easily monitor their meters through their devices and are more aware of their water consumption.

Contents

Chapter 1

Introduction

Water meters are devices which measure the amount of water passing through a pipe. It displays the amount in gallons or in cubic feet. In order for water companies to record the water consumption of residents, they send out their personnel to perform water meter readings in residential areas every month. This whole process, from reading water meters to recording them, is done manually.

Doing tasks manually introduces problems such as data inconsistencies, tedious work processes, and human error. In the case of water meter reading, these problems also occur and affect the consumers and the water service providers. Last June 2020, the Metropolitan Waterworks and Sewerage System (MWSS) Regulatory Office (RO) received more than a hundred complaints from social media regarding their bills for that month[1]. These complaints revolve around the situation in which there were alleged errors in the consumers' water bills. The MWSS provided explanations that they are not having extra charges to the consumers. This kind of problem comes from data inconsistencies because people who complained about their situation may have a firm estimate of their water usage. Another problem of the current and long been method of water meter reading is the issue with remote access. Since personnel from water companies personally go to a residential area for meter reading, they expose themselves to environmental hazards and currently, COVID-19. During the Enhanced Community Quarantine (ECQ) last March 2020, MWSS suspended the reading of water meters because of safety concerns for both the employees and the customers. During this time, the bill of consumers will purely be estimated based on the consumers' average monthly consumption and when meter readings resume, the actual consumption amount will be adjusted and corrected to their subsequent bills [2].

These problems of manual water meter reading can be addressed with various solutions. Having the ability to access these meters remotely allows consumers and water service providers to monitor readings whenever they want. An automated water meter reading is used in conjunction with an ESP32-CAM module. Integrating a computer vision based system can completely replace the process of having to personally go to residential areas to read the meter. Furthermore, the data and screenshots of the water meter captured from the system can be seen by the users through the Internet.

Chapter 2

Review of Related Work

There are numerous works regarding automated water meter reading. There are related works focused on just automating water meter reading without remote access to the water meter information, several studies include [3] - [4], while others also included a remote access for the automated readings using an Electronic Interface Module (EIM) [5], a Cloud based app [6], or with a Raspberry Pi [7].

2.1 Dataset

The dataset used by most of the related works were produced by themselves for their study/work. In [3], Peng and Chen constructed their own dataset by taking pictures of different water meter readings in different angles and in different environmental states. They named their dataset RWMID. Salomon and his team's work also created their own dataset called UPFR-ADMR [4]. These images inside the dataset were from employees of a Brazillian energy company. These images were taken by the employees in different environments and the device used to take these photos were very different. They then compressed the images into 640x480 or 480x640. In [7], Puttnies et al. created their own dataset, albeit only from one water meter model. Puttnies et al. then used the ICDAR-2003 dataset, which has 507 scene images that consists of various printed words on different backgrounds, to verify their character recognition systems.

The construction of the RWMID and UPFR-ADMR were photos taken manually of the subject by the researchers themselves, and to have other people do so. While these methods of data gathering are the standard, the ongoing pandemic made it difficult for this study to manually acquire photos of water meters. For this reason, this paper instead used a dataset which is publicly available through the website

Kaggle [8] for training the model.

2.2 Region of Interest

The values of a water meter is located in its Region of Interest (ROI). The importance of the ROI is seen heavily in [3] as Peng and Chen devised a system called the R-FCN structure that made use of the ROI. The ROI was found by creating masks to separate the foreground and the background. Finding the ROI was also heavily emphasized in the work of Puttnies et al. [7]. Puttnies et al. proposed a system where after pre-processing the images, they would then detect the ROI by finding an accumulation of vertical edges/lines. Puttnies et al. would then run an algorithm that would find these edges/lines and then create a mask as seen in Figure 2.1.

On the contrary, in [4], Salomon et al. were able to obtain promising results by foregoing ROI detection and image pre-processing. Salomon et al. were able to achieve this by having a straightforward approach to detecting the dial of the water meter. They just used the input images instead of creating masks and the likes to extract the ROI.

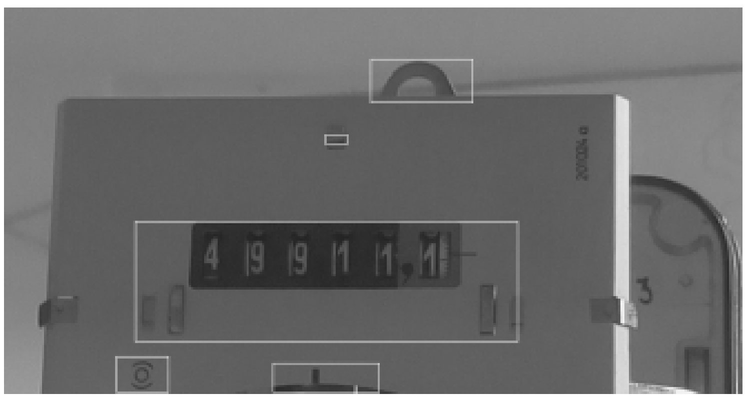
This paper made use of the ROI of the water meter. The researches made use of bounding boxes to identify the ROI on each water meter picture. This allowed the researchers to make use of the ROI and input it into the deep learning model for digit reading.

2.3 Deep Learning Model

In [3], Peng and Chen used two methods of a deep learning target detection in their *Object detection framework*. The first is a *Region proposal based*, which is a two-staged approach where it would have several frames used as samples that was inputted into a convolution neural network (CNN). The second is a *Regression method based*, which is a one-staged approach where they used the YOLO (You Only Look Once) algorithm. YOLO is a CNN based algorithm that predicts multiple box locations. The two methods have different results. The two-staged approach produces a more accurate algorithm, while its recognition rate is slower than a one-staged approach. The study had a model called R-FCN model where the FCN is primarily based on a 34 layer ResNet.

In [4], object detection was also explored by Salomon et al. Salomon et al. used and evaluated two deep networks, Faster R-CNN and YOLO. While both had good results, R-CNN was more accurate while YOLO had a higher frames per second.

Input image after vertical edge detection



ROIs found in the input image

Figure 2.1: Region of Interest taken from [7]

This paper evaluated Kucev's dataset [8] using Mask-RCNN, and ResNet's upgraded deep network called ResNeXt.

2.4 Remote Access

In [6], a large scale operation for automated water reading involves the use of Cloud. This server-less architecture contains all of the computations, analytic recordings and more. Cloud is used due to its reliability while also being able to handle multiple users.

In a smaller scale, Suresh, Muthukumar and Chandapillai's work [5] is preferred. Suresh et al. made use of an Electronic Interface Module connected to a meter with three methods to transmit its data remotely. The first method would be an EIM transmitting the water meter values to a smartphone via wifi or bluetooth however this would require frequent charging. This method is recommended for bulk water meters that consume large amounts. A second method would be an EIM connected via Ethernet to a router. Lastly, Suresh et al. have an EIM that interfaces to multiple meters remotely achieved via Ethernet Router / Gateway to the Internet. The methods included the features of being able to see meter reading information with a date-time stamped consumption, tamper flag and meter ID viewed through an app.

Lastly, in [7], a Raspberry Pi Model B together with a Raspberry Pi camera was used. This prototype costs $ 30. Puttnies et al. then plugged it with a C++ code together with either MatLab or OpenCV as they had used 2 models. The code was then executed with the Raspbian OS inside the Raspberry Pi.

For this paper, the researchers used the ESP32-CAM module due to it being fully featured with remote access potential and a camera. This full-featured microcontroller packed with a camera and an SD slot is priced at a price of less than $ 10.

2.5 Summary

In Table 2.1 is the summary of comparisons between the related works and the researchers' proposal in their dataset, region of interest, deep learning model and remote access.

	Related Work	**Researchers' Project**
Dataset	RWMID, UFPR-ADMIR, ICDAR-2003	Kucev Roman's '*Water Meters Dataset*'
Region of Interest (ROI)	R-FCN structure, Image pre-processing, Accumulation of vertical lines-edges	Bounding Box
Deep Learning Model	R-FCN model w/ a 34-layer ResNet, R-CNN, YOLO	Mask R-CNN, ResNeXt
Remote Access	Cloud, Electronic Interface Module, Raspberry Pi Model B	ESP32-CAM, Cloud – Google Drive

Table 2.1: Comparison between Related Works and Researchers' Proposed Project

Chapter 3

Problem Statement and Objectives

3.1 Problem Statement

3.1.1 Background

Reading water meters has long been a manual task for water companies as well as consumers to do. The manual process of water meter reading can cause problems such as data inconsistencies, human error, or health risks due to the COVID-19 pandemic. Although some residents know how to monitor their water consumption very well and water companies have maintained this state of water meter reading for a long time, new processes are being developed to remove certain kinds of problems that are brought about by the manual nature of water meter reading.

3.1.2 Problem

The researchers wanted to address the lack of an engineering system that looks at an accurate and real-time water consumption monitoring of a household. This is so the consumers can track their water bills in real-time, but also be able to cross reference their water bill.

3.1.3 Solution

This project created a system that allows water companies and residents to monitor their water meters remotely. This was done by installing ESP32-CAM modules in front of the water meters to capture a screenshot of the meter display. The said photo will then be sent into a cloud, in this case - Google Drive.

A deep learning model will then be used to take in the captured photo and process the photo to produce numeric results of the meter readings. The results will then be accessed through an IoT Dashboard seen in the website *things.ph*.

This will be useful for water companies to reduce the risk of their employees personally going to the site and exposing them with different environmental hazards, tedious reading and recording process, and COVID-19.

3.2 Objectives

The main objective of the project is to develop an engineering solution that automates water meter reading. Based on this, the objectives of the project are as follows:

- To use a module that captures images in intervals
- To input these images into a deep learning model that accurately reads the water meter
- To clearly visualize the results for the consumers to easily view

Chapter 4

Methodology

The system architecture for the project used a cloud server, in the form of Google Drive. It also used Google Colab to train the models for the project. The methodology was divided into several parts. These are the dataset, deep learning models, ESP32-CAM Module, and the IoT Dashboard.

As an overview, the researchers tackle what the dataset consists of, and the method of gathering the aforementioned dataset. For the deep learning models, the researchers talk about the different deep learning models they used - mainly used for ROI detection, digit detection, and digit reading. The ESP32-CAM Module is also discussed, which was used for the capturing of images of the water meter.

The process of the system architecture, seen in Figure 4.1, starts with the ESP32-CAM. This ESP32-CAM captures the images of the water meter. These images are then sent to the cloud server. This is then accessed by the deep learning model of ROI Detection then Digit Detection, and finally Digit Reading. After this, it is sent to the IoT Dashboard for the users to see.

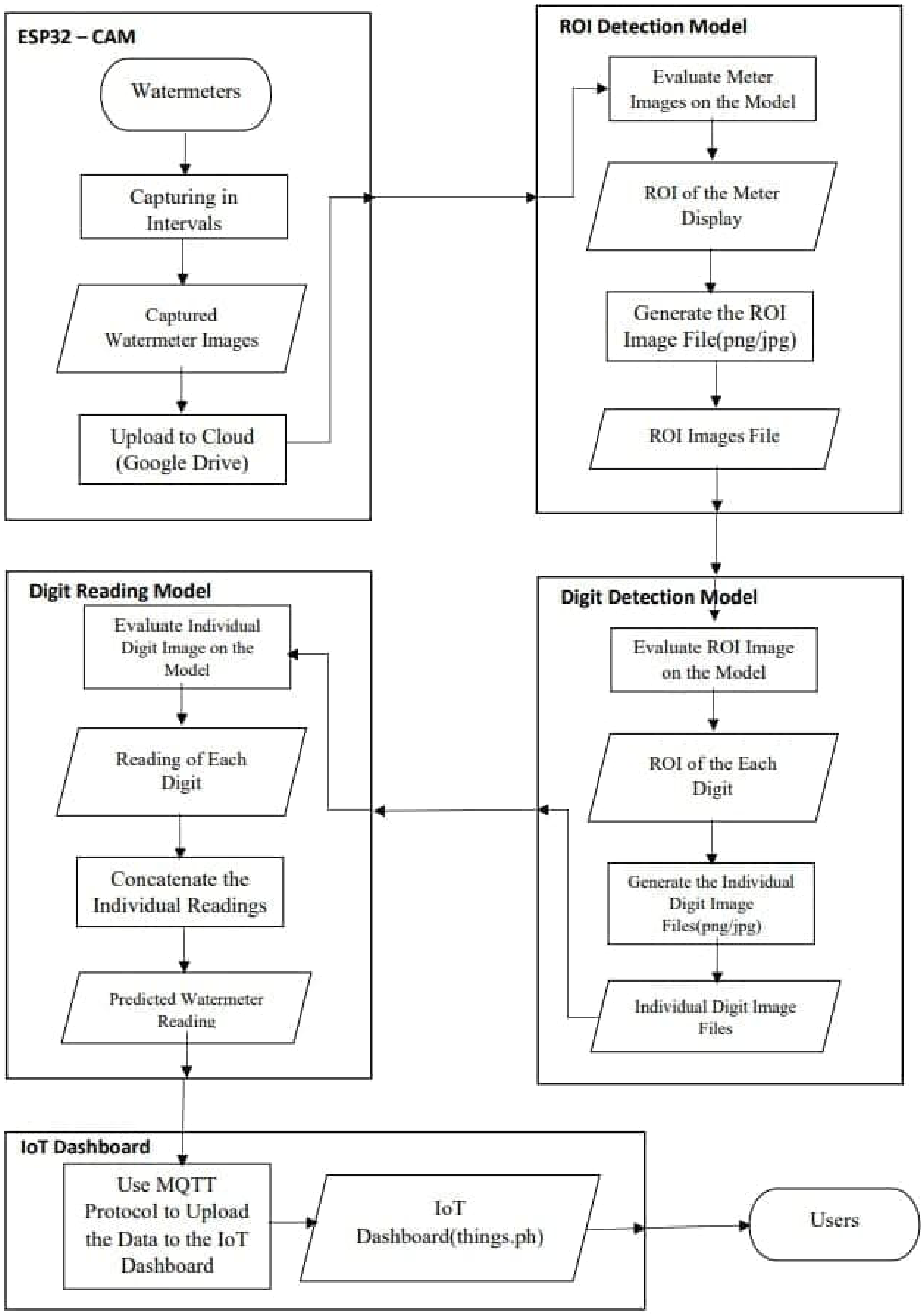


Figure 4.1: Block Diagram of the System Architecture

4.1 Dataset

The dataset is a collection of similar images that is used by a deep learning model to train on. In this paper, the researchers used water meter images to train the deep learning model so that it may be able to read the said water meter. While there was a dataset for the deep learning model to train on, there needs to be a dataset to test and validate the trained deep learning model. The researchers made use of both a training dataset taken from Kaggle, made by Roman Kucev [8]. While for the testing dataset, the researchers noted that the dataset by Roman Kucev contains the correct reading of the water meter in the filename. The predicted reading of each water meter was compared to the aforementioned correct reading. The researchers then validated the model by inputting their dataset created from the ESP32-CAM.

4.1.1 Gathering the Dataset

Initially, the researchers planned on making their own dataset with many different water meters for training but the creation of a new water meter dataset will take a long amount of time, while also being exposed to the dangers of the COVID-19 pandemic. Thus the dataset that was used, was from an open source dataset of water meter pictures taken from Kaggle. This dataset, as seen in Figure 4.2, contains 1244 images of different water meter pictures with their corresponding label. The pictures also have various different lighting, angles, glare and quality as seen in Figure 4.2.

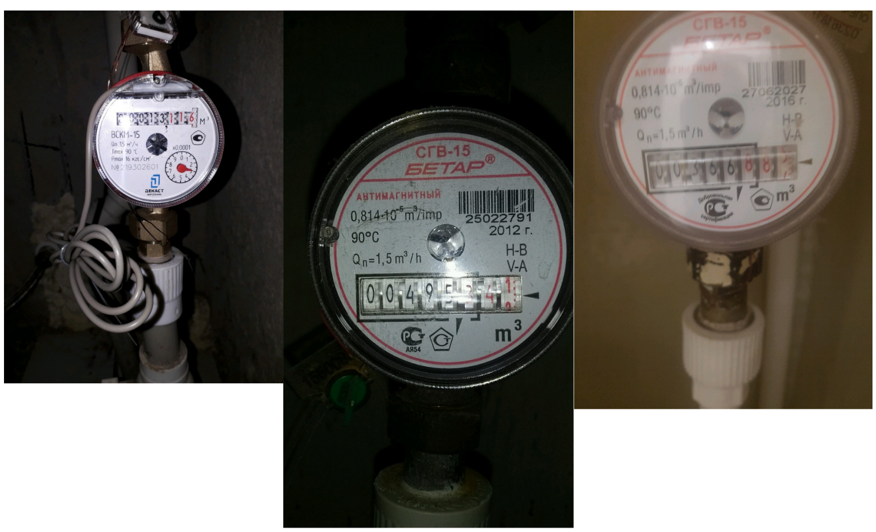


Figure 4.2: Sample water meter images taken from [8]

As for the validation dataset, the researchers created their own small dataset to test the deep learning model that is trained under Kucev's dataset. They captured the water meters of their respective household using the ESP32-CAM module. Though it is only a single water meter for each, the images are taken at differing times with differing environmental conditions.

4.2 ESP32-CAM Module

The camera that will be used to capture the displays of the water meter will be the ESP32-CAM. This module is a full-featured microcontroller that has a built camera for taking pictures and a micro SD card socket as well. The module has a built-in LED that can act as a flashlight for lighting the object in front. It also has WiFi, bluetooth, and multipurpose GPIO pins[9].

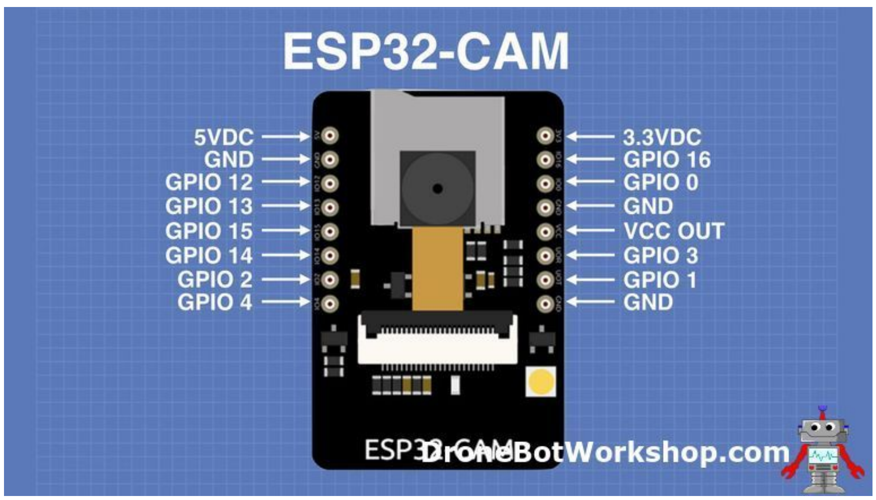


Figure 4.3: An ESP32-CAM module. Image taken from [9]

The ESP32-CAM module has the following specifications:

- 802.11b/g/n Wi-Fi
- Bluetooth 4.2 with BLE
- UART, SPI, I2C and PWM interfaces
- Clock speed up to 160 MHz
- Computing power up to 600 DMIPS
- 520 KB SRAM plus 4 MB PSRAM

- Supports WiFi Image Upload
- Multiple Sleep modes
- Firmware Over the Air (FOTA) upgrades possible
- 9 GPIO ports
- Built-in Flash LED

This module also supports and contains the OV7670 camera module which has the following specifications:

- 2 Megapixel sensor
- Array size is UXGA 1622x1200
- Output formats include YUV422, YUV420, RGB565, RGB555 and 8-bit compressed data
- Image transfer rate of 15 to 60 fps

4.2.1 ESP32-CAM Interfacing

In order to load programs into the ESP32-CAM module, an external component called the FTDI adapter will act as the connection between the ESP32-CAM and the computer. Since the ESP32-CAM has no USB ports, this extra step is needed. Figure 4.4 shows the connections between the two components (FTDI adapter on the right).

The schematic diagram of the circuit is also shown at Figure 4.4 but with an addition of a power source of 5V. It will be supplied at the 5VDC pin. The researchers used a power supply that outputs 5VDC. The two main options for this will be the use of a battery or through a power supply plugged into an outlet. The first option allows portability while the second option allows longer and constant power supply provided by the outlet.

To power the camera, 3.3V would be needed. The GPIO 0 pin must also be shorted with GND for the module's programming mode. With this setup, the module can now be loaded with programs.

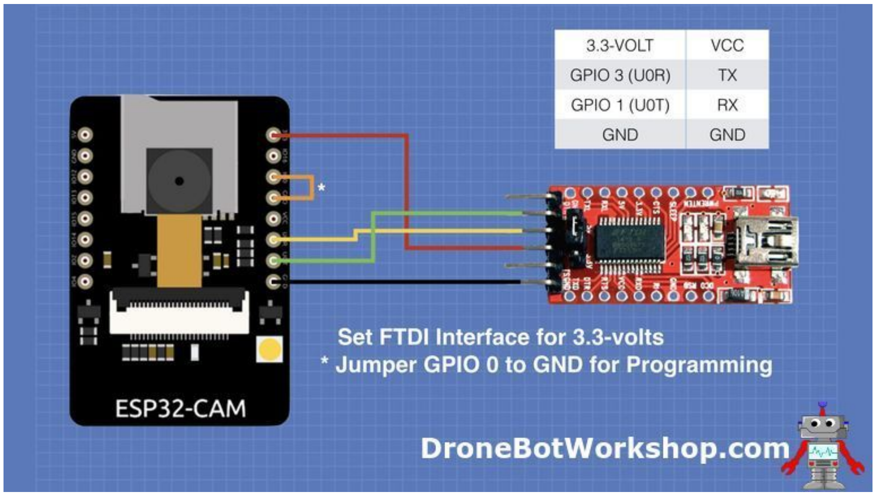


Figure 4.4: ESP32-CAM interfaced with FTDI adapter. Image taken from [9]

4.2.2 ESP32-CAM Placement

The ESP32-CAM module will be placed inside a plastic casing for protection against outside forces such as rain or dust. It will be placed in front of the water meter display to provide full readability as much as possible for image processing. A flashlight is already built-in the board itself to provide light whenever the image looks dark.

4.3 Deep Learning Models

The researchers used three deep learning models in order to obtain the reading of the water meter. The ROI Detection Model, the Digit Detection Model, and the Digit Reading Model. As mentioned before, the captured images from the ESP32-CAM are sent into the ROI Detection Model. Then the output of the ROI Detection Model is cascaded into the next deep learning model, which is the Digit Detection Model. Finally, the output of the Digit Detection Model is inputted into the Digit Reading Model.

4.3.1 ROI Detection Model

The ROI Detection Model creation was started by annotating the dataset for its bounding box, creating masks around the bounding box, and cropping the ROI Dataset. This overview is seen in Figure 4.5.

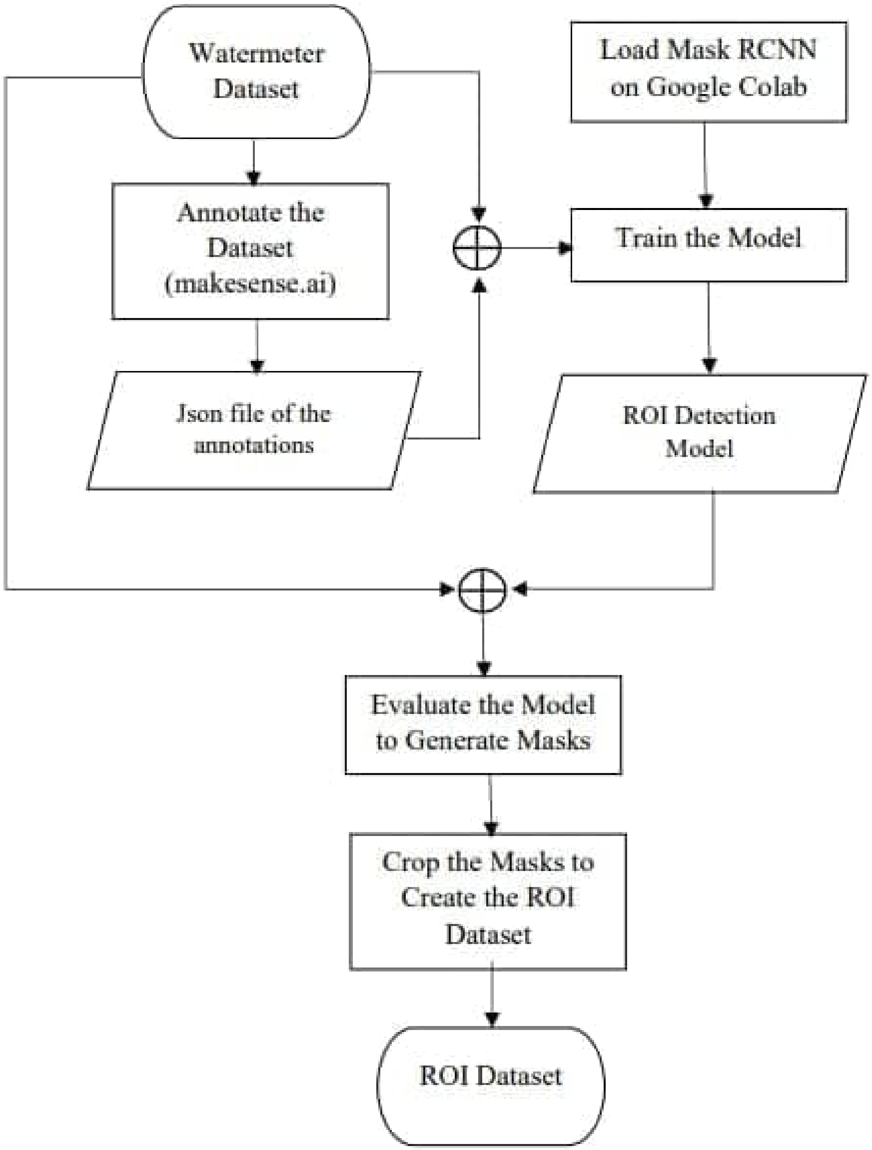


Figure 4.5: ROI Detection Model Block Diagram

This model generates bounding boxes, focused on the water meter reading of the water meter. The dataset was manually annotated using a polygon bounding box. A website called makesense.ai was used to create the annotations as seen in Figure 4.6. The annotations were stored in a .json format for the deep learning model to be trained with.

Figure 4.6: Sample image of water meter being annotated in makesense.ai website

After training, the deep learning model can automatically output a bounding box onto each water meter. The readings in the water meter is inside the generated bounding boxes in the water meter, as seen in Figure 4.7. This object detection was done using Mask R-CNN [10] which needed the dataset and the .json annotations as inputs.

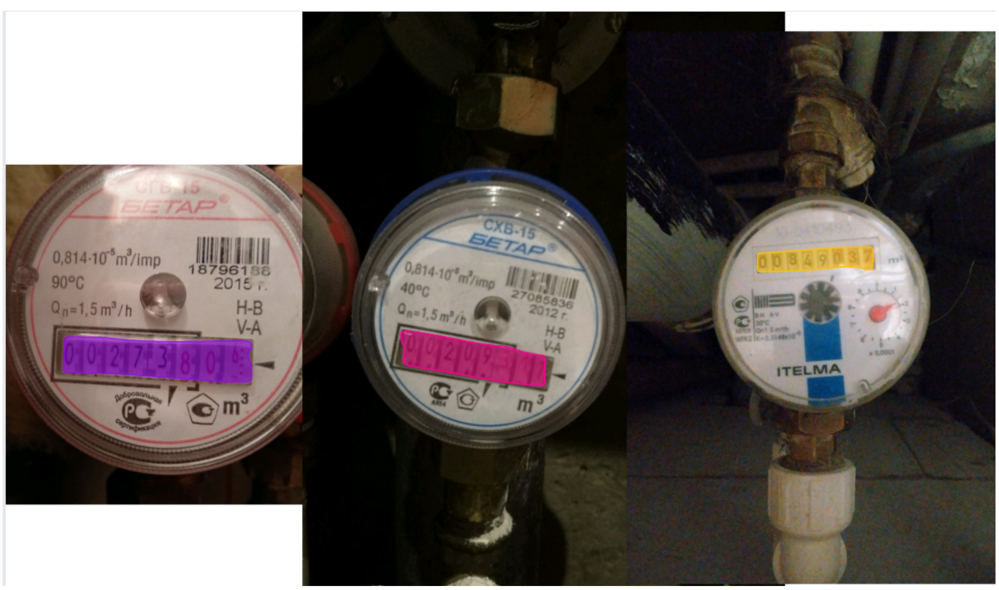


Figure 4.7: Detected ROI from randomly selected water meter images

The researchers then obtained masked images from the bounding box. Each image was generated with only the contents of the bounding box intersected with the water meter, as seen in Figure 4.8. Then, the images were created into individual image files for use in the next model. The researchers removed the black background, which resulted in showing only the ROI from the water meter as seen in Figure 4.9.

Figure 4.8: Masked images from Figure 4.7

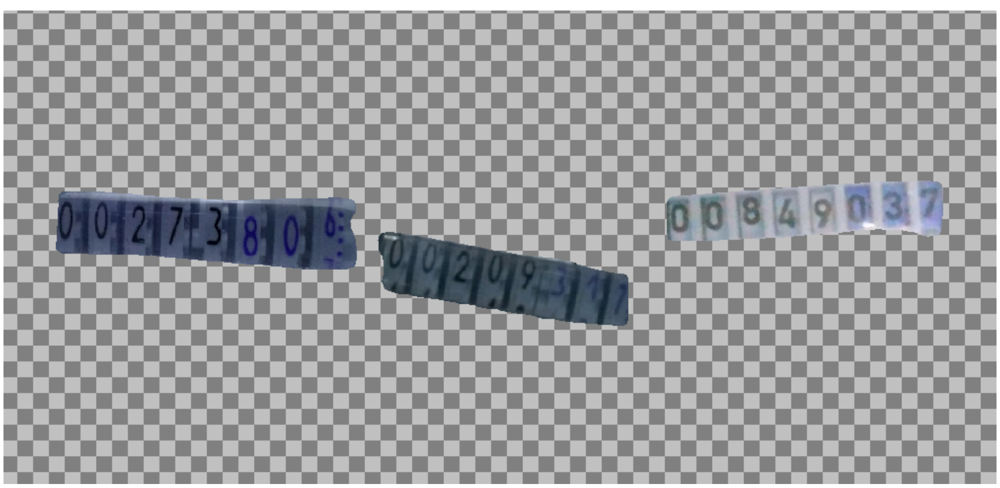


Figure 4.9: Cropped images from Figure 4.8

4.3.2 Digit Detection Model

The Digit Detection Model follows the same process as the ROI Detection Model, only this time the inputted images are the outputted images from the ROI Detection Model. To specify, the outputted images, from the ROI Detection Model, are individual images of each digit seen in the water meter. It was then ordered correctly from left to right to be able to read the water meter correctly. This is seen in the block diagram in Figure 4.10.

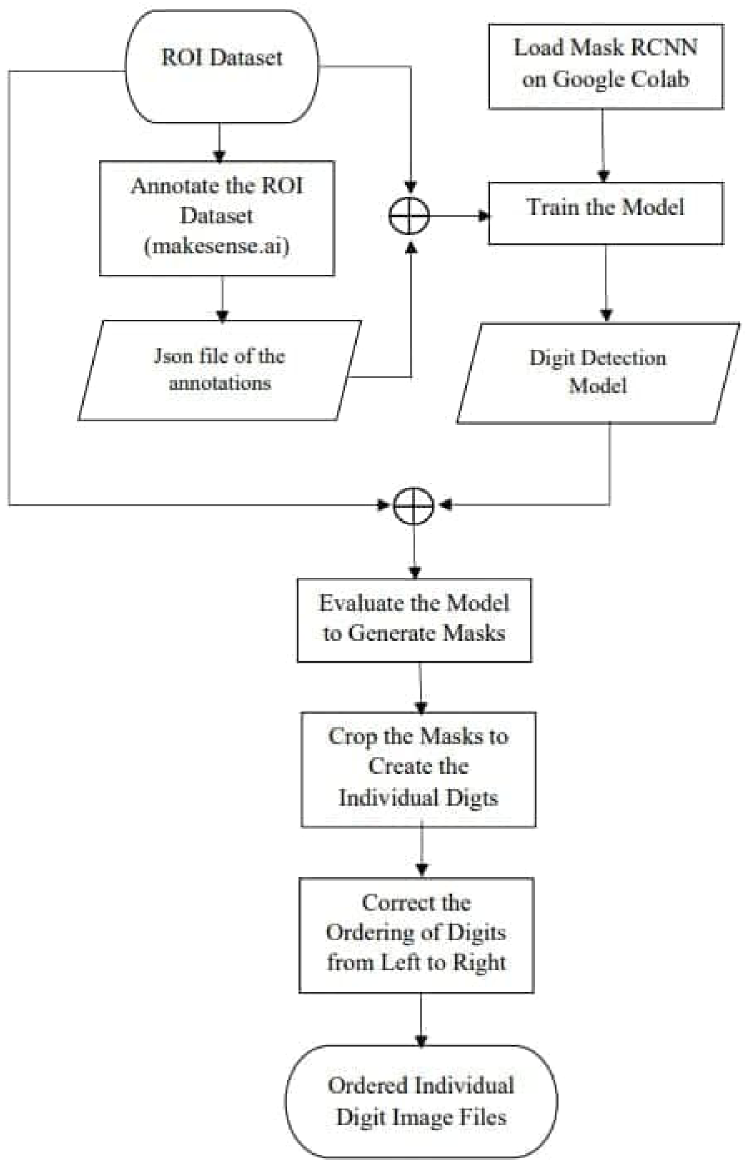


Figure 4.10: Digit Detection Model Block Diagram

The researchers put bounding boxes around the digits in the output of the ROI Detection Model, again using the website makesense.ai, as seen in Figure 4.11. This was inputted into the Digit Detection Model along with its annotation to help train the said deep learning model.

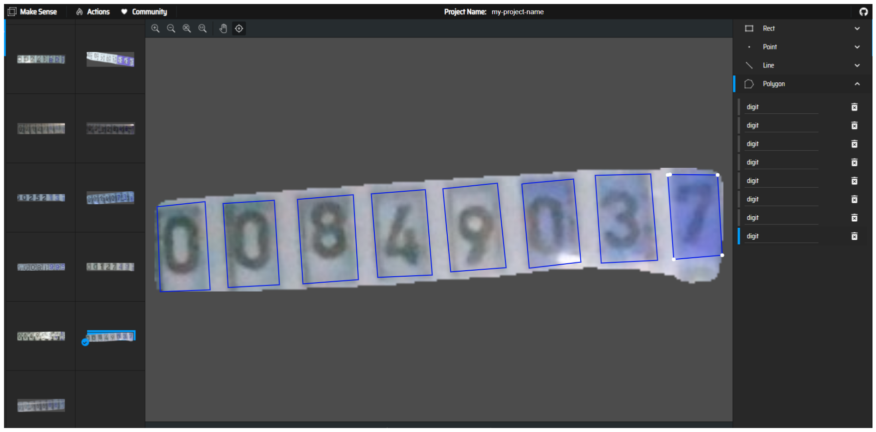


Figure 4.11: Manually annotated bounding boxes around the digits using makesense.ai website

Once finished training, bounding boxes will now appear around the digits of the cropped water meter. As seen in Figure 4.12, each bounding box contains a single digit from the cropped water meter reading. Just like in the ROI Detection Model, the researchers created a mask around each bounding box.

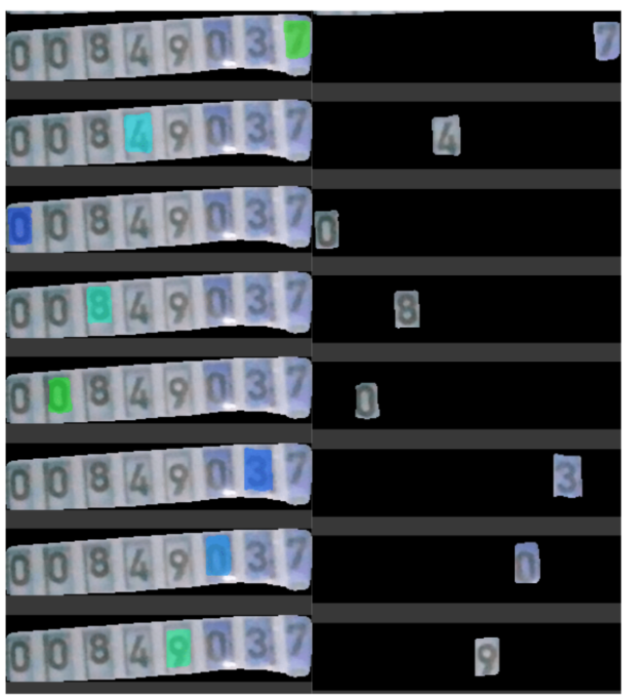


Figure 4.12: Detected digits and their corresponding masks

However, this method runs into the problem of having digits that are not in order. With this in mind, the researchers used an algorithm which ordered the digits. The algorithm takes into account the x-coordinates of each individual image. The researchers then rearranged the produced digits in ascending order, as seen in Figure 4.13.

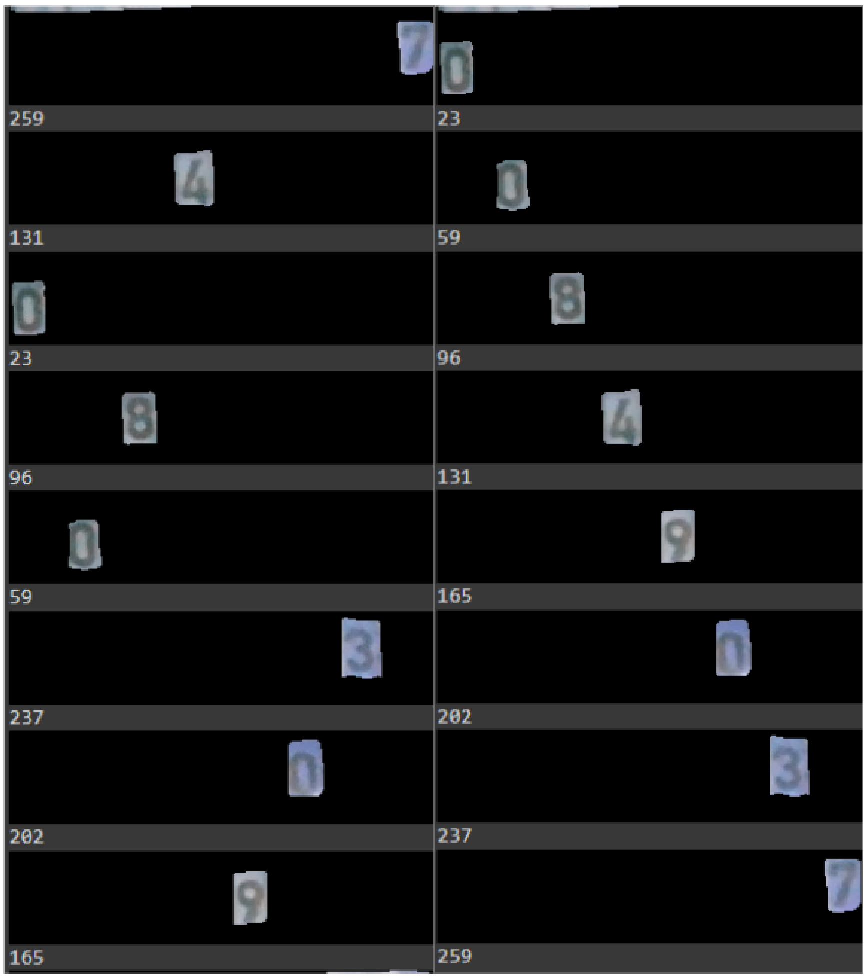


Figure 4.13: Ordering of digits

To finish the Digit Detection Model, the researchers cropped the images from its mask and created individual image files for the next deep learning model to use. For visualization, in Figure 4.14 the upper half is the individual files while the lower half is the individual images put together.

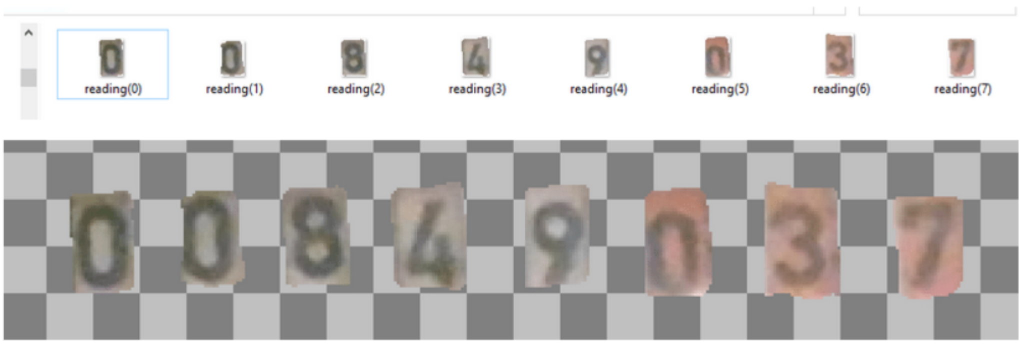


Figure 4.14: Individual images

4.3.3 Digit Reading Model

The Digit Reading Model evaluates each inputted image outputted by the Digit Detection Model. The researchers made use of their previously created Deep Learning model for SVHN digit reading. Particularly, this Deep Learning model was a ResNeXt model created for their CoE 197Z - Deep Learning course. The researchers sorted each individual image, outputted by the Digit Detection Model, into separate folders with their respective digit.

Each individual image, starting from the leftmost of the water meter, is then read. This is later concatenated to the other individual images fro the same water meter until all images are analyzed. Once all digits are analyzed and concatenated, the resulting number is the predicted water meter reading. This is explained by the block diagram as seen in Figure 4.15

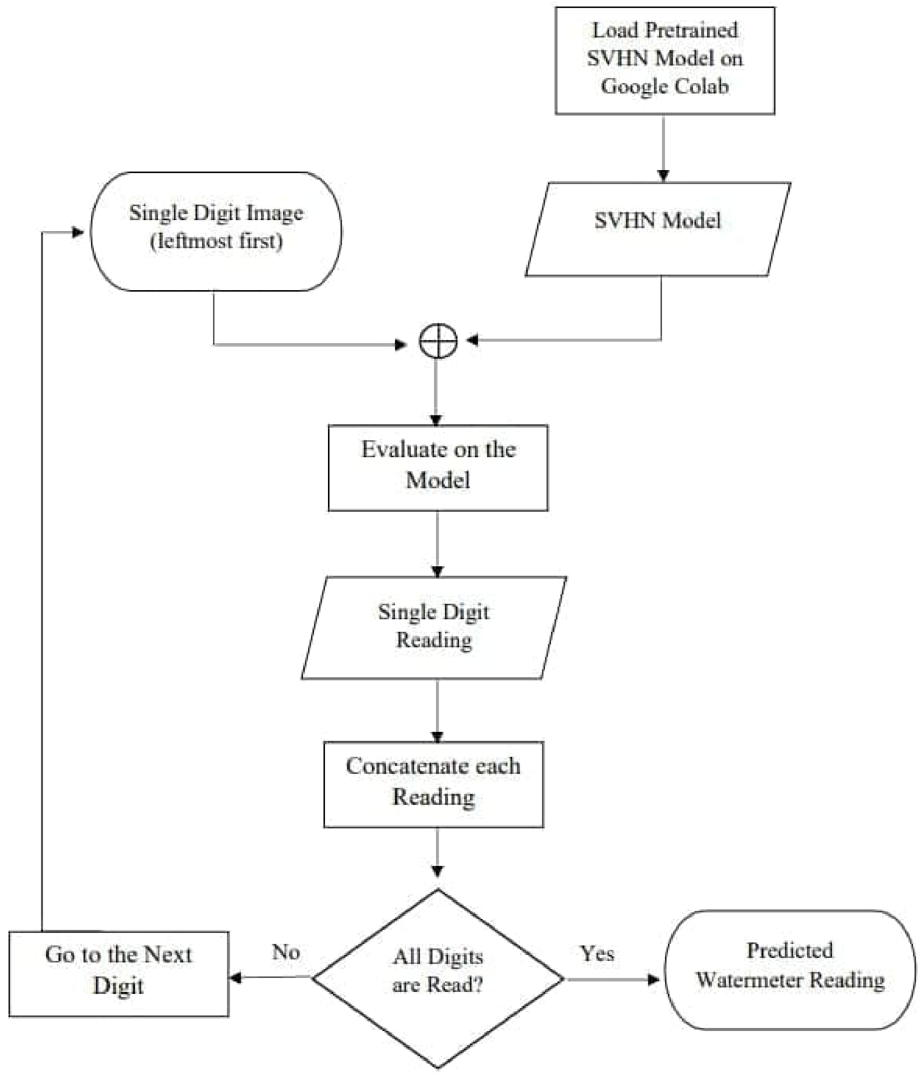


Figure 4.15: Digit Reading Model Block Diagram

4.4 IoT Dashboard

The predicted water meter reading is sent to the researchers' IoT Dashboard. This IoT Dashboard is found in the website *things.ph*. The predicted text is uploaded via the MQTT Protocol to the researchers' account in the website *things.ph*.

Once the predicted water meter is sent, the results may be viewed in the things.ph website as seen in Figure 4.16. In the IoT Dashboard, at the top left is the current reading of the water meter. Beside it is a graph of the reading vs time diagram and below it is its history. At the bottom left, the estimated total bill can be seen. Beside the total bill is the breakdown of the billing.

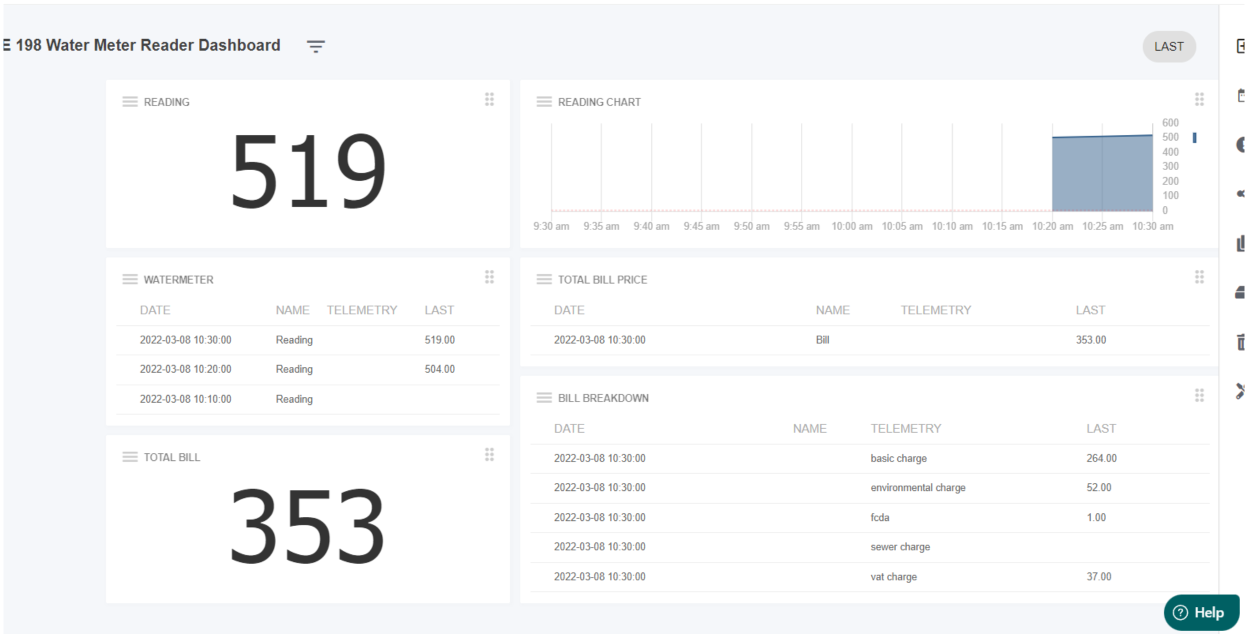


Figure 4.16: IoT Dashboard - things.ph website showing results

4.5 Water Bill Calculator

The predicted reading was then used as an input to create an estimated calculation for the water bill. First, the researchers assumed that the water meter is a residential household. This is to ease the calculation needed for the water bill. Next, the Maintenance Service Charge (MSC) was not factored in the calculations because its influence is negligible. In addition, there are too many meter sizes that would need to be taken into account for. Finally, the researchers were only able to calculate for Maynilad and Manila Water because their calculations for the water bill is available online. This can be seen in Figure 4.17 and Figure 4.18.

A sample of this can be seen when 2 images of the same water meter, taken from the validation dataset, is run through the system. These water meters can be seen in Figure 4.19. The output on the IoT Dashboard, complete with the estimated water bill is shown below, in Fig 4.20.

MWSS

NOTICE TO MAYNILAD CUSTOMERS AND THE PUBLIC
NEW WATER RATES FOR THE WEST ZONE

Maynilad

Effective 15 days after publication, Maynilad Water Services, Inc. (Maynilad) will implement a **Rate Adjustment Limit (RAL) of 5.70% which is the change in Consumer Price Index** to be applied to the 2018 Basic Charge, as recommended by MWSS Regulatory Office (RO) in its Resolution No. 2018-13-CA dated 6 December 2018 and as approved/confirmed by the MWSS Board of Trustees' (BOT) Resolution No. 2018-191-RO dated 13 December 2018.

Maynilad is an agent and contractor of the Metropolitan Waterworks and Sewerage System (MWSS) for the West Zone of the Greater Metro Manila Area, which consists of the following areas: The cities of **Manila** (all but portions of San Andres & Sta. Ana), **Quezon City** (west of San Juan River, West Avenue, EDSA, Congressional, Mindanao Ave.; the northern part starting from Districts of the Holy Spirit & Batasan Hills), **Makati** (west of South Super Hi-way), **Caloocan**, **Las Piñas**, **Malabon, Muntinlupa, Navotas, Parañaque, Pasay**, and **Valenzuela -** all in **Metro Manila**; the cities of **Cavite**, **Bacoor** and **Imus**, and the towns of **Kawit**, **Noveleta** and **Rosario -** all in Cavite Province. Maynilad has 12 Business Area offices across the West Zone that serve its customers.

The new schedule of water and sewer rates for all Maynilad customers is as follows:

1. WATER CHARGES
A. BASIC CHARGE

	Old Rate	New Rate		Old Rate	New Rate
RESIDENTIAL			**SEMI-BUSINESS**		
I. Those consuming 10 cu. m. or less (lifeline consumers)					
(Net of lifeline discount)	₱ 91.13 /conn.	₱ 96.32 /conn.			
II. Those consuming more than 10 cu. m.					
First 10 cu.m.	₱ 155.31 /conn.	₱ 164.16 /conn.	First 10 cu.m.	₱ 155.31 /conn.	₱ 164.16 /conn. *
Next 10 cu.m.	18.95 /cu.m.	20.03 /cu.m.	Next 10 cu.m.	31.81 /cu.m.	33.62 /cu.m.
Next 20 cu.m.	36.04 /cu.m.	38.09 /cu.m.	Next 20 cu.m.	39.22 /cu.m.	41.45 /cu.m.
Next 20 cu.m.	47.34 /cu.m.	50.03 /cu.m.	Next 20 cu.m.	49.74 /cu.m.	52.57 /cu.m.
Next 20 cu.m.	55.30 /cu.m.	58.45 /cu.m.	Next 20 cu.m.	57.84 /cu.m.	61.13 /cu.m.
Next 20 cu.m.	57.84 /cu.m.	61.13 /cu.m.	Next 20 cu.m.	60.52 /cu.m.	63.96 /cu.m.
Next 50 cu.m.	60.49 /cu.m.	63.93 /cu.m.	Next 50 cu.m.	63.18 /cu.m.	66.78 /cu.m.
Next 50 cu.m.	63.18 /cu.m.	66.78 /cu.m.	Next 50 cu.m.	65.85 /cu.m.	69.60 /cu.m.
Over 200 cu.m.	65.85 /cu.m.	69.60 /cu.m.	Over 200 cu.m.	68.44 /cu.m.	72.34 /cu.m.
BUSINESS GROUP I			**BUSINESS GROUP II**		
First 10 cu.m.	₱ 705.82 /conn.	₱ 746.05 /conn.	₱ First 10 cu.m.	₱ 763.74 /conn.	₱ 807.27 /conn.
Next 90 cu.m.	70.91 /cu.m.	74.95 /cu.m.	Next 90 cu.m.	76.87 /cu.m.	81.25 /cu.m.
Next 100 cu.m.	71.10 /cu.m.	75.15 /cu.m.	Next 100 cu.m.	77.34 /cu.m.	81.74 /cu.m.
Next 100 cu.m.	71.32 /cu.m.	75.38 /cu.m.	Next 100 cu.m.	77.96 /cu.m.	82.40 /cu.m.
Next 100 cu.m.	71.56 /cu.m.	75.63 /cu.m.	Next 100 cu.m.	78.45 /cu.m.	82.92 /cu.m.
Next 100 cu.m.	71.79 /cu.m.	75.88 /cu.m.	Next 100 cu.m.	78.96 /cu.m.	83.46 /cu.m.
Next 100 cu.m.	72.12 /cu.m.	76.23 /cu.m.	Next 100 cu.m.	79.54 /cu.m.	84.07 /cu.m.
Next 100 cu.m.	72.44 /cu.m.	76.56 /cu.m.	Next 100 cu.m.	80.04 /cu.m.	84.60 /cu.m.
Next 100 cu.m.	72.68 /cu.m.	76.82 /cu.m.	Next 100 cu.m.	80.52 /cu.m.	85.10 /cu.m.
Next 100 cu.m.	72.92 /cu.m.	77.07 /cu.m.	Next 100 cu.m.	81.12 /cu.m.	85.74 /cu.m.
Next 100 cu.m.	73.15 /cu.m.	77.31 /cu.m.	Next 100 cu.m.	81.58 /cu.m.	86.23 /cu.m.
Next 200 cu.m.	73.48 /cu.m.	77.66 /cu.m.	Next 200 cu.m.	82.14 /cu.m.	86.82 /cu.m.
Next 200 cu.m.	73.69 /cu.m.	77.89 /cu.m.	Next 200 cu.m.	82.63 /cu.m.	87.33 /cu.m.
Next 200 cu.m.	74.08 /cu.m.	78.30 /cu.m.	Next 200 cu.m.	83.24 /cu.m.	87.98 /cu.m.
Next 200 cu.m.	74.28 /cu.m.	78.51 /cu.m.	Next 200 cu.m.	83.67 /cu.m.	88.43 /cu.m.
Next 200 cu.m.	74.46 /cu.m.	78.70 /cu.m.	Next 200 cu.m.	84.25 /cu.m.	89.05 /cu.m.
Next 500 cu.m.	74.68 /cu.m.	78.93 /cu.m.	Next 500 cu.m.	84.78 /cu.m.	89.61 /cu.m.
Next 500 cu.m.	75.08 /cu.m.	79.35 /cu.m.	Next 500 cu.m.	85.26 /cu.m.	90.11 /cu.m.
Next 500 cu.m.	75.33 /cu.m.	79.62 /cu.m.	Next 500 cu.m.	85.75 /cu.m.	90.63 /cu.m.
Next 500 cu.m.	75.60 /cu.m.	79.90 /cu.m.	Next 500 cu.m.	86.38 /cu.m.	91.30 /cu.m.
Next 500 cu.m.	75.74 /cu.m.	80.05 /cu.m.	Next 500 cu.m.	86.82 /cu.m.	91.76 /cu.m.
Next 500 cu.m.	76.06 /cu.m.	80.39 /cu.m.	Next 500 cu.m.	87.45 /cu.m.	92.43 /cu.m.
Next 500 cu.m.	76.38 /cu.m.	80.73 /cu.m.	Next 500 cu.m.	87.91 /cu.m.	92.92 /cu.m.
Next 500 cu.m.	76.56 /cu.m.	80.92 /cu.m.	Next 500 cu.m.	88.38 /cu.m.	93.41 /cu.m.
Next 500 cu.m.	76.87 /cu.m.	81.25 /cu.m.	Next 500 cu.m.	88.99 /cu.m.	94.06 /cu.m.
Next 500 cu.m.	77.14 /cu.m.	81.53 /cu.m.	Next 500 cu.m.	89.52 /cu.m.	94.62 /cu.m.
Next 500 cu.m.	77.34 /cu.m.	81.74 /cu.m.	Next 500 cu.m.	90.02 /cu.m.	95.15 /cu.m.
Next 500 cu.m.	77.64 /cu.m.	82.06 /cu.m.	Next 500 cu.m.	90.64 /cu.m.	95.80 /cu.m.
Next 500 cu.m.	77.96 /cu.m.	82.40 /cu.m.	Next 500 cu.m.	91.03 /cu.m.	96.21 /cu.m.
Next 500 cu.m.	78.18 /cu.m.	82.63 /cu.m.	Next 500 cu.m.	91.67 /cu.m.	96.89 /cu.m.
Next 500 cu.m.	78.45 /cu.m.	82.92 /cu.m.	Next 500 cu.m.	92.13 /cu.m.	97.38 /cu.m.
Next 500 cu.m.	78.73 /cu.m.	83.21 /cu.m.	Next 500 cu.m.	92.71 /cu.m.	97.99 /cu.m.
Over 10000 cu.m.	78.97 /cu.m.	83.46 /cu.m.	Over 10000 cu.m.	93.19 /cu.m.	98.50 /cu.m.

**Based on IRR-2008-03 dated 31 March 2008 and confirmed by MWSS BOT Res. No. 2008-064 dated 24 April 2008, the first 10 cubic meter of water consumed in Semi-Business customers shall be billed at Residential Rate.*

B. FOREIGN CURRENCY DIFFERENTIAL ADJUSTMENT (FCDA) - A percentage of the Basic Charge subject to periodic review and adjustment.
The FCDA for the 1st Quarter of 2019 is negative 0.27% of the new Basic Charge.

2. A. ENVIRONMENTAL CHARGE (EC) - 20% of Water Charge
Applicable to all customers of Maynilad

B. SEWERAGE CHARGE (SC) for Customers Connected to Sewerlines
0% of Water Charge for Residential and Semi-Business Customers
20% for Water Charge for Business Group I and II customers

3. MAINTENANCE SERVICE CHARGE (MSC)

METER SIZE	AMOUNT (per conn.)
1/2" or 13mm	₱ 1.50
3/4" or 20mm	2.00
1" or 25mm	3.00

METER SIZE	AMOUNT (per conn.)
1 1/4" or 40mm	₱ 4.00
2" or 50mm	6.00
3" or 75mm	10.00

METER SIZE	AMOUNT (per conn.)
4" or 100mm	₱ 20.00
6" or 150mm	35.00
8" or 200mm	50.00

4. VALUE-ADDED TAX (VAT) - 12% of Charges 1, 2 and 3

THE MONTHLY BILL IS THE SUM OF 1, 2, 3 and 4

Approved by:

ATTY. PATRICK LESTER NG TY
Chief Regulator, MWSS-RO

PDDG. REYNALDO V. VELASCO (Ret.)
Administrator, MWSS

RAMONCITO S. FERNANDEZ
President and CEO, Maynilad

For further inquiries you may call Maynilad Water Services Hotline at 1626 or visit www.mayniladwater.com.ph

Maynilad

What Maynilad customers should know

Beginning 1 January 2019, residents in the West Zone will have adjusted water rates as Maynilad implements the Consumer Price Index (CPI) adjustment of 5.70% equivalent to an average basic rate adjustment of ₱1.95/cu.m. from the ₱34.29/cu.m. average Basic Charge in 2018.

Maynilad was also granted a Foreign Currency Differential Adjustment (FCDA) equivalent to a negative 0.27% as applied to the newly approved 2019 Average Basic Charge of ₱36.24/cu.m., beginning 1st quarter of 2019.

FCDA is a tariff mechanism granted to the concessionaires to allow them to recover losses or give back gains arising from the fluctuating movements of the peso against other currencies. This is because Maynilad pays foreign currency denominated Concession Fees to the MWSS, as well as loans to fund service improvement projects.

Impact on monthly water bills

Depending on their water consumption, Maynilad residential customers can expect the following adjustment in their monthly water bills:

Monthly consumption (cu.m.)	Monthly bill adjustment (₱)
10	5.30
20	20.08
30	41.02

Even as we continue to invest in infrastructure that ensures water security and environmental sustainability, we continue to provide discounted rates to senior citizens and to low-income residential households consuming 10 cu.m. or less per month.

Investments in service improvement programs will be sustained

√ **Water Security** – To support the development of additional water sources and construction of more pump stations and reservoirs.

√ **Disaster Resiliency** – To retrofit existing facilities and thus mitigate the impact of natural calamities on operations, as well as adapt to climate change.

√ **Environmental Sustainability** – To accelerate the construction of wastewater treatment facilities that will prevent harmful effluents from flowing into water bodies, thus protecting the environment and promoting community health.

√ **Expansion** – To lay new pipelines and other facilities that will bring potable water to under-served and unserved areas in the West Zone.

Our commitment to provide excellent customer service stays strong

Since re-privatization in 2007, Maynilad has spent over ₱91.2 billion to improve and expand water and wastewater operations. Over 9 million people now have reliable and affordable potable water because of Maynilad's aggressive investments. Majority of its customers now have 24-hour water supply, and 100% receive their water supply at an average pressure of 7 psi.

Maynilad strives to continue its programs to enhance operational efficiency, build more wastewater facilities, and lay pipes that will connect more people to its network. This is all in line with the company's mission to improve the lives of the communities it serves.

For more information, please visit www.mayniladwater.com.ph.

Figure 4.17: Maynilad Water Rates

MWSS

NOTICE TO MANILA WATER CUSTOMERS AND THE PUBLIC
NEW WATER RATES FOR THE EAST ZONE

MANILA WATER
CARE IN EVERY DROP

Effective January 1, 2019, Manila Water Company, Inc. (MWCI) will implement a **5.70% Consumer Price Index** adjustment on the existing Basic Charge, based on MWSS Regulatory Office Resolution No. 2018-12-CA dated December 6, 2018 and as approved by the MWSS Board Resolution No. 2018-190-RO dated December 13, 2018.

Manila Water Company, the East Zone concessionaire, covers the following areas: **Manila** (San Andres and Sta. Ana only), **Quezon City** (east of San Juan River, West Avenue, EDSA, Congressional and Mindanao Ave., Districts of Tandang Sora, Pasong Tamo and Matandang Balara), **Makati City** (east of South Super Highway), **Mandaluyong City, San Juan City, Marikina City, Pasig City, Pateros, Taguig City - all in Metro Manila; Rizal Province.**

The new schedule of water and sewer rates for all MWCI customers is as follows:

1. WATER CHARGE

A. BASIC CHARGE

RESIDENTIAL			OLD RATE	NEW RATE	SEMI-BUSINESS			OLD RATE	NEW RATE
I. Low-income household consuming 10 cu. m. or less			₱ 59.75 /conn.**	₱ 63.16 /conn.**					
II. Consuming more than 10 cu. m.									
First	10	cu.m.	₱ 105.27 /conn.	₱ 111.27 /conn.	First	10	cu.m.	₱ 105.27 /conn.	₱ 111.27 /conn. *
Next	10	cu.m.	12.83 /cu.m.	13.56 /cu.m.	Next	10	cu.m.	21.49 /cu.m.	22.71 /cu.m.
Next	20	cu.m.	24.33 /cu.m.	25.71 /cu.m.	Next	20	cu.m.	26.50 /cu.m.	28.01 /cu.m.
Next	20	cu.m.	32.06 /cu.m.	33.89 /cu.m.	Next	20	cu.m.	33.68 /cu.m.	35.60 /cu.m.
Next	20	cu.m.	37.45 /cu.m.	39.58 /cu.m.	Next	20	cu.m.	39.25 /cu.m.	41.49 /cu.m.
Next	20	cu.m.	39.25 /cu.m.	41.49 /cu.m.	Next	20	cu.m.	41.00 /cu.m.	43.34 /cu.m.
Next	50	cu.m.	41.00 /cu.m.	43.34 /cu.m.	Next	50	cu.m.	42.76 /cu.m.	45.20 /cu.m.
Next	50	cu.m.	42.76 /cu.m.	45.20 /cu.m.	Next	50	cu.m.	44.52 /cu.m.	47.06 /cu.m.
Over	200	cu.m.	44.52 /cu.m.	47.06 /cu.m.	Over	200	cu.m.	46.39 /cu.m.	49.03 /cu.m.

BUSINESS GROUP I			OLD RATE	NEW RATE	BUSINESS GROUP II			OLD RATE	NEW RATE
First	10	cu.m.	₱ 478.39 /conn.	₱ 505.66 /conn.	₱ First	10	cu.m.	₱ 517.63 /conn.	₱ 547.13 /conn.
Next	90	cu.m.	47.89 /cu.m.	50.62 /cu.m.	Next	90	cu.m.	52.08 /cu.m.	55.05 /cu.m.
Next	100	cu.m.	48.16 /cu.m.	50.91 /cu.m.	Next	100	cu.m.	52.36 /cu.m.	55.34 /cu.m.
Next	100	cu.m.	48.30 /cu.m.	51.05 /cu.m.	Next	100	cu.m.	52.77 /cu.m.	55.78 /cu.m.
Next	100	cu.m.	48.43 /cu.m.	51.19 /cu.m.	Next	100	cu.m.	53.18 /cu.m.	56.21 /cu.m.
Next	100	cu.m.	48.69 /cu.m.	51.47 /cu.m.	Next	100	cu.m.	53.43 /cu.m.	56.48 /cu.m.
Next	100	cu.m.	48.83 /cu.m.	51.61 /cu.m.	Next	100	cu.m.	53.85 /cu.m.	56.92 /cu.m.
Next	100	cu.m.	48.99 /cu.m.	51.78 /cu.m.	Next	100	cu.m.	54.25 /cu.m.	57.34 /cu.m.
Next	100	cu.m.	49.28 /cu.m.	52.09 /cu.m.	Next	100	cu.m.	54.50 /cu.m.	57.61 /cu.m.
Next	100	cu.m.	49.38 /cu.m.	52.19 /cu.m.	Next	100	cu.m.	54.90 /cu.m.	58.03 /cu.m.
Next	100	cu.m.	49.53 /cu.m.	52.35 /cu.m.	Next	100	cu.m.	55.36 /cu.m.	58.52 /cu.m.
Next	200	cu.m.	49.77 /cu.m.	52.61 /cu.m.	Next	200	cu.m.	55.63 /cu.m.	58.80 /cu.m.
Next	200	cu.m.	49.92 /cu.m.	52.77 /cu.m.	Next	200	cu.m.	56.01 /cu.m.	59.20 /cu.m.
Next	200	cu.m.	50.06 /cu.m.	52.91 /cu.m.	Next	200	cu.m.	56.27 /cu.m.	59.48 /cu.m.
Next	200	cu.m.	50.34 /cu.m.	53.21 /cu.m.	Next	200	cu.m.	56.70 /cu.m.	59.93 /cu.m.
Next	200	cu.m.	50.47 /cu.m.	53.35 /cu.m.	Next	200	cu.m.	57.09 /cu.m.	60.34 /cu.m.
Next	500	cu.m.	50.61 /cu.m.	53.49 /cu.m.	Next	500	cu.m.	57.36 /cu.m.	60.63 /cu.m.
Next	500	cu.m.	50.86 /cu.m.	53.76 /cu.m.	Next	500	cu.m.	57.78 /cu.m.	61.07 /cu.m.
Next	500	cu.m.	51.00 /cu.m.	53.91 /cu.m.	Next	500	cu.m.	58.17 /cu.m.	61.49 /cu.m.
Next	500	cu.m.	51.13 /cu.m.	54.04 /cu.m.	Next	500	cu.m.	58.44 /cu.m.	61.77 /cu.m.
Next	500	cu.m.	51.40 /cu.m.	54.33 /cu.m.	Next	500	cu.m.	58.84 /cu.m.	62.19 /cu.m.
Next	500	cu.m.	51.55 /cu.m.	54.49 /cu.m.	Next	500	cu.m.	59.27 /cu.m.	62.65 /cu.m.
Next	500	cu.m.	51.69 /cu.m.	54.64 /cu.m.	Next	500	cu.m.	59.52 /cu.m.	62.91 /cu.m.
Next	500	cu.m.	51.96 /cu.m.	54.92 /cu.m.	Next	500	cu.m.	59.93 /cu.m.	63.35 /cu.m.
Next	500	cu.m.	52.08 /cu.m.	55.05 /cu.m.	Next	500	cu.m.	60.37 /cu.m.	63.81 /cu.m.
Next	500	cu.m.	52.23 /cu.m.	55.21 /cu.m.	Next	500	cu.m.	60.60 /cu.m.	64.05 /cu.m.
Next	500	cu.m.	52.36 /cu.m.	55.34 /cu.m.	Next	500	cu.m.	61.01 /cu.m.	64.49 /cu.m.
Next	500	cu.m.	52.65 /cu.m.	55.65 /cu.m.	Next	500	cu.m.	61.27 /cu.m.	64.76 /cu.m.
Next	500	cu.m.	52.77 /cu.m.	55.78 /cu.m.	Next	500	cu.m.	61.72 /cu.m.	65.24 /cu.m.
Next	500	cu.m.	52.90 /cu.m.	55.92 /cu.m.	Next	500	cu.m.	62.10 /cu.m.	65.64 /cu.m.
Next	500	cu.m.	53.18 /cu.m.	56.21 /cu.m.	Next	500	cu.m.	62.37 /cu.m.	65.93 /cu.m.
Next	500	cu.m.	53.29 /cu.m.	56.33 /cu.m.	Next	500	cu.m.	62.79 /cu.m.	66.37 /cu.m.
Over	10000	cu.m.	53.43 /cu.m.	56.48 /cu.m.	Over	10000	cu.m.	63.18 /cu.m.	66.78 /cu.m.

** Based on IRR-2008-03 dated 31 March 2008 and confirmed by MWSS BOT Res. No. 2008-064 dated 24 April 2008, the first 10 cubic meter of water consumed in Semi-Business customers shall be billed at Residential Rate.*

*** Low-income residential customers consuming 10 cubic meters or less will have a discounted rate. Low-income Customers are those qualified based on the definition provided by Article 1 of the Concession Agreement.*

B. Foreign Currency Differential Adjustment (FCDA) - A percentage of the Basic Charge subject to periodic review and adjustment. The FCDA is 2.62% of the new Basic Charge.

2. A. ENVIRONMENTAL CHARGE (EC) - 20% of Water Charge applicable to all customers of MWCI

B. SEWERAGE CHARGE (SC)
0% of Water Charge for Residential and Semi Business
30% of Water Charge for Business Group I and II customers

3. MAINTENANCE SERVICE CHARGE

METER SIZE	AMOUNT (per conn.)
1/2" or 13mm	₱ 1.50
3/4" or 20mm	2.00
1" or 25mm	3.00

METER SIZE	AMOUNT (per conn.)
1 1/4" or 40mm	₱ 4.00
2" or 50mm	6.00
3" or 75mm	10.00

METER SIZE	AMOUNT (per conn.)
4" or 100mm	₱ 20.00
6" or 150mm	35.00
8" or 200mm	50.00

4. VALUE-ADDED TAX (VAT) 12% of the Charges 1, 2 and 3

THE MONTHLY BILL IS THE SUM OF 1, 2, 3 and 4.

Approved by:

PATRICK LESTER N. TY
Chief Regulator, MWSS-RO

Approved by:

PDDG. REYNALDO V. VELASCO (Ret.)
Administrator, MWSS

FERDINAND M. DELA CRUZ
President and CEO, MWCI

For further inquiries you may call Manila Water Company Hotline at 1627 or visit www.manilawater.com

Figure 4.18: Manila Water Rates

Figure 4.19: Two images of the same water meter taken in different times

In figure 4.20, the bill breakdown is included in the table as well. Based from the readings in the sample, the water consumption would be 131 cubic meters. The corresponding bill is then calculated in the code following the updated tariff for water meters.

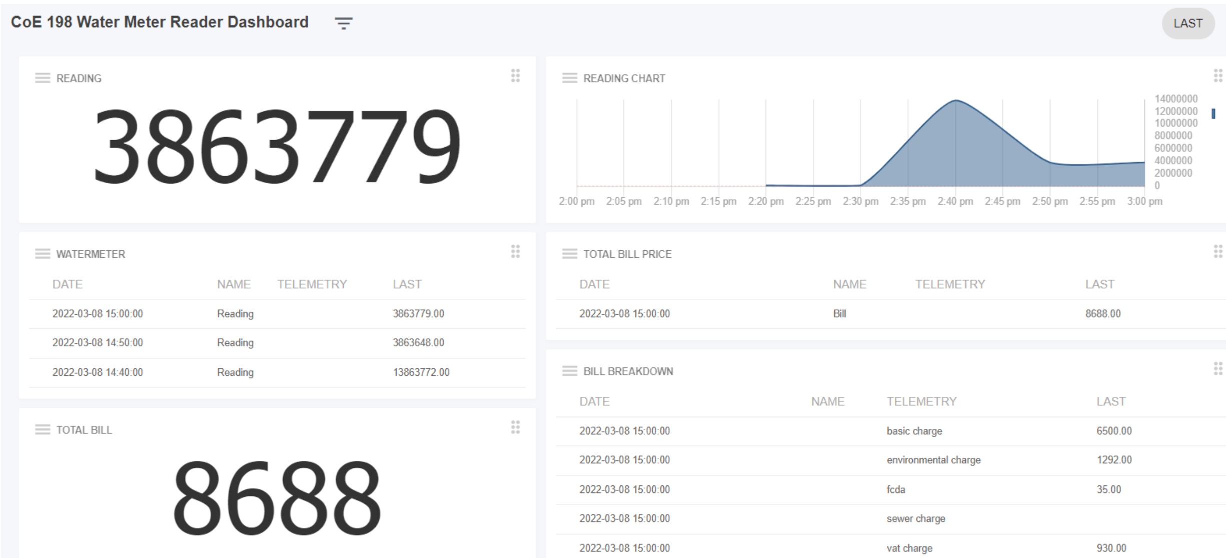


Figure 4.20: Estimated Water Bill of Fig 4.19

Chapter 5

Results and Discussion

5.1 Data and Results

The researchers needed to test the model if it was accurate and working correctly. This meant that the researchers had to create a program that checks each image from the dataset and see if the predicted reading is the same as the actual reading.

5.1.1 Digit Reading Test

With the importance of viewing the digits of the water meter, the digit reading model's results needed to be correct. To aid in this objective, the researchers used a standard accuracy test to see how correct the predicted label per image was. Another test that the researchers used is the F1 score. This is because of the potential of an imbalanced classification problem posed to the digit reading model due to the high amount of classes used.

5.1.1.1 Digit Reading Accuracy

First, the researchers checked the accuracy and loss of the digit reading model. The researchers checked this first, as this is a vital part of the model. In Figure 5.1, a graph of the accuracy and loss of the training dataset's digit reading model is seen. In Figure 5.2. the validation dataset of the digit model is seen.

From the digit reading model, one can see that at Epoch 24 the accuracy of the training dataset is **99.66%** and the loss is **1.42%**. This proves satisfactory to the researchers. Some examples of the digit images and the predicted reading can be seen in Figure 5.3.

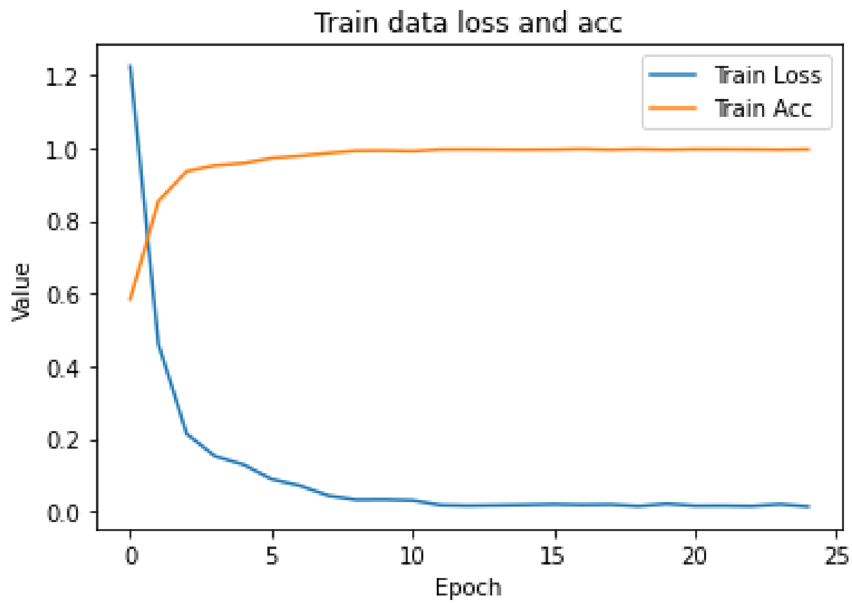


Figure 5.1: Accuracy and Loss of training dataset

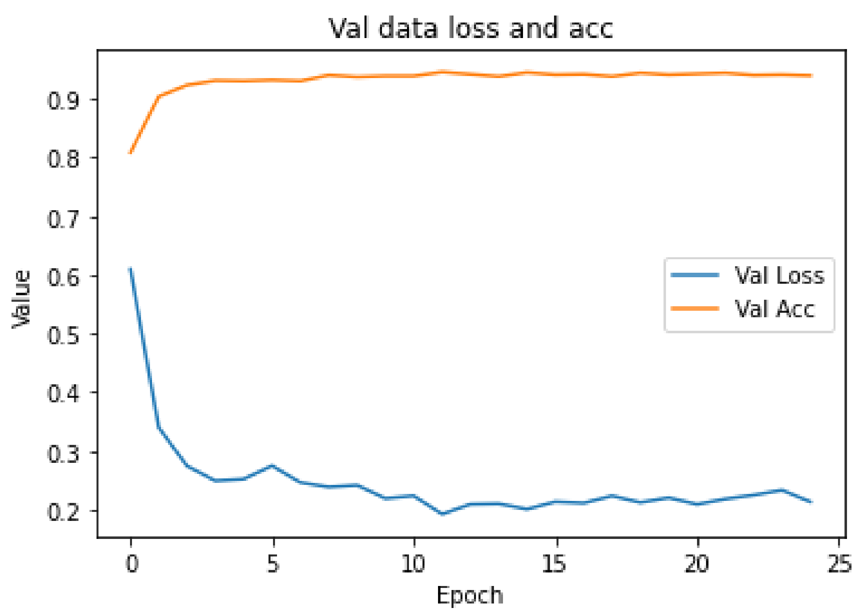


Figure 5.2: Accuracy and Loss of testing dataset

Figure 5.3: Sample images of digits w/ predicted reading

5.1.1.2 Digit Reading F1 Score

A common metric used for machine learning that uses classification models, is the F1 score [11]. This is used to solve and observe class imbalances in a model, which is where the accuracy lacks in. This was particularly useful for the Digit Reading model. This is done by first getting a precision and recall value. The formula for the precision value can be seen in Figure 5.4, while for the recall value it can be seen in Figure 5.5. Finally, to calculate for the F1 score, Figure 5.6 is used.

$$Precision = \frac{\#\ of\ True\ Positives}{\#\ of\ True\ Positives + \#\ of\ False\ Positives}$$

Figure 5.4: Precision Formula for F1 Score taken from [11]

$$Recall = \frac{\#\ of\ True\ Positives}{\#\ of\ True\ Positives + \#\ of\ False\ Negatives}$$

Figure 5.5: Recall Formula for F1 Score taken from [11]

$$F1\ score = 2 * \frac{Precision\ * Recall}{Precision + Recall}$$

Figure 5.6: F1 Score Formula taken from [11]

In particular, the Digit Reading model's confusion matrix for the multi-class values of the digits can be seen in Table 5.1. For the Digit Reading model, its classes' precision value, and recall value is seen in Table 5.2. Finally, the F1 score per class is seen in Table 5.3.

			A	C	T	U	A	L			
		0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
P	**0**	2439	2	1	1	3	2	4	0	0	2
R	**1**	2	542	4	1	5	0	0	5	0	1
E	**2**	2	0	472	2	1	0	1	0	0	3
D	**3**	1	1	1	496	1	4	0	0	0	2
I	**4**	0	2	3	0	392	1	2	0	0	0
C	**5**	1	0	1	4	0	354	2	0	2	0
T	**6**	4	0	2	1	1	1	390	0	0	2
E	**7**	0	10	5	1	5	0	0	366	0	0
D	**8**	1	0	1	2	1	5	6	0	384	5
	9	4	0	0	3	0	3	0	1	6	328

Table 5.1: Confusion matrix of the multi-class values

CLASS	PRECISION	RECALL
0	0.993887531	0.993887531
1	0.967857143	0.973070018
2	0.981288981	0.963265306
3	0.980237154	0.970645793
4	0.98	0.958435208
5	0.972527473	0.956756757
6	0.972568579	0.962962963
7	0.945736434	0.983870968
8	0.948148148	0.979591837
9	0.950724638	0.956268222

Table 5.2: Table of Precision and Recall values of the Digit Reading model

Averaging all of the F1 scores of all classes, the researchers obtained the Macro F1 score of **0.969490**. Also considering the amount of images per class, the researchers got the Weighted F1 score of **0.977804**.

CLASS	F1 SCORE per class
0	0.993887531
1	0.97045658
2	0.972193615
3	0.975417896
4	0.969097651
5	0.964577657
6	0.967741935
7	0.964426877
8	0.963613551
9	0.953488372

Table 5.3: The table of F1 score per class

5.1.2 Model Accuracy Test

Next, the researchers tested the accuracy of the entire system - i.e. the cascade of the ROI Detection Model, Digit Detection Model, and Digit Reading Model. The researchers were able to do this by first noting that the water meters inside the dataset had its correct reading written in its filename after the word *value*, as seen in Figure 5.7. The researchers then had to evaluate each predicted reading and compare it to the filename of each image.

The researchers then created a program that tested each result, added up the number of correct readings then divided it by the total amount of readings to see the accuracy of the system. The final accuracy that the researchers attained for the system was 83.993%.

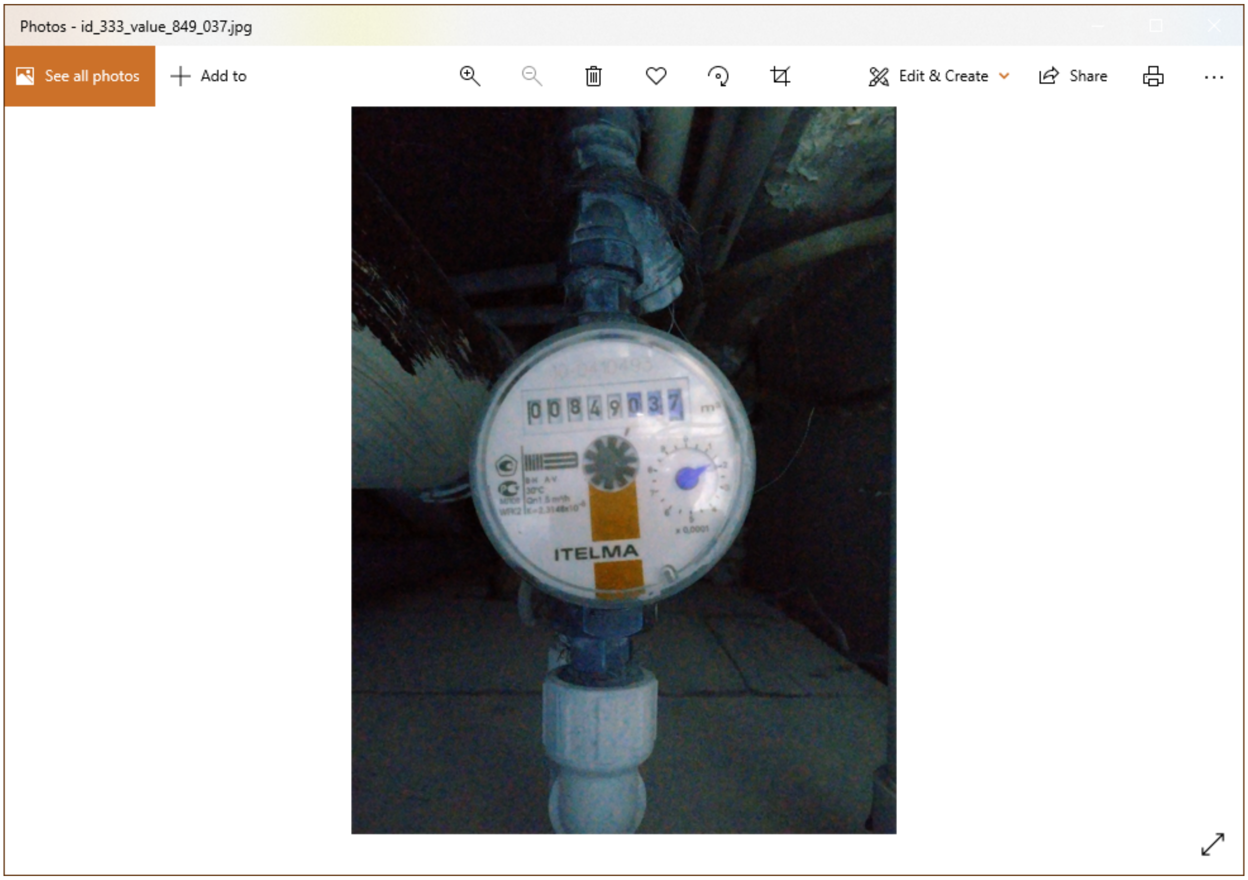


Figure 5.7: Water Meter image shown with its filename

5.1.3 Validation of System using ESP32-CAM Water Meter Images

Inputting the 2 images from Figure 4.19 into the system, the researchers were able to validate and test the system for Philippine-based water meters. In Figure 5.8, the output of the system taken for the initial and final water meter reading is seen. The water consumption seen in the upper half of Figure 5.8 is 0 due to it being the initial reading. In the lower half of Figure 5.8, the current reading was subtracted by the initial reading. In doing so, the water consumption for the final reading is calculated.

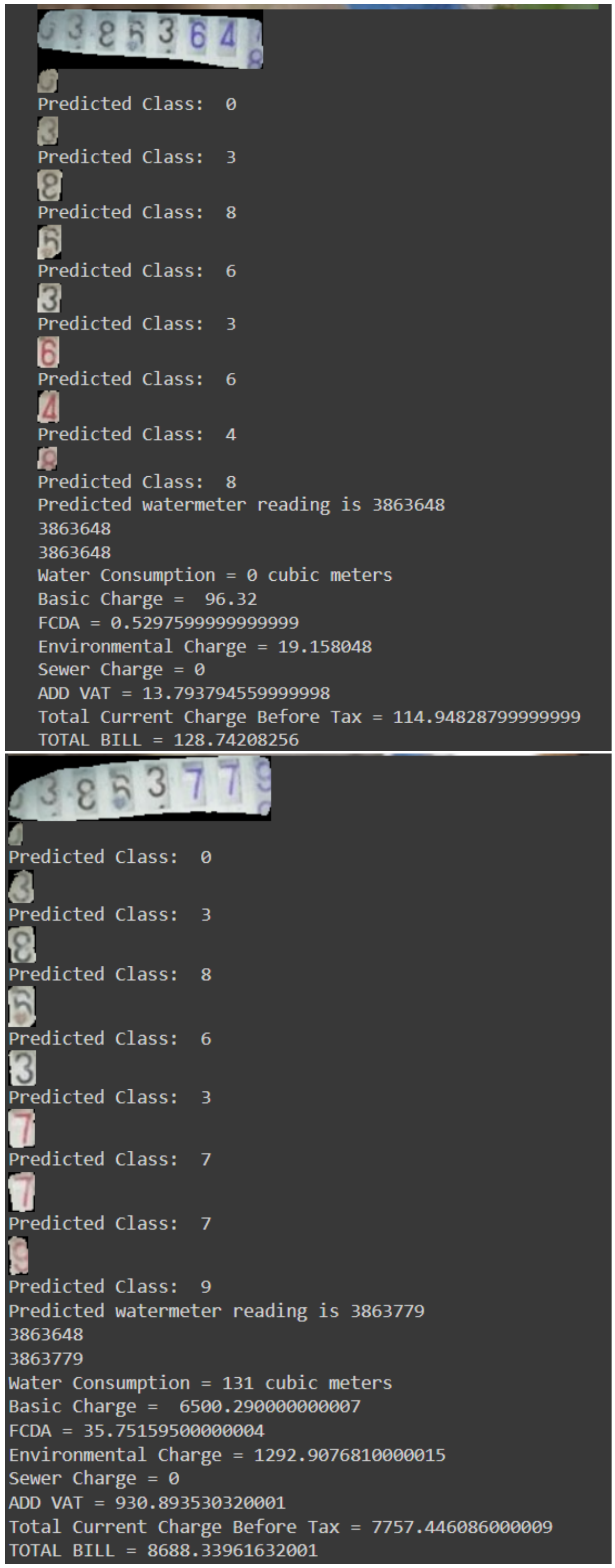


Figure 5.8: Initial and final image's output from the system

For this figure, we saw how the system was able to predict a reading. It started with creating an ROI using the ROI Detection model, then it split the ROI into individual digits using the Digit Detection model, finally the Digit Reading model predicted and labeled each individual image. Putting it all together, the system was able to calculate an estimated water bill.

Next, the results were inputted into things.ph for the IoT Dashboard to display. This can be seen in Figure 5.8, where the predicted water bill was Php 8688 with a predicted water consumption of 131 cubic meters. Comparing this to Maynilad's bill calculator [12] as seen in Figure 5.9, the accuracy is obtained by dividing the predicted water bill by the actual water bill. In this case, the accuracy is 99.977%.

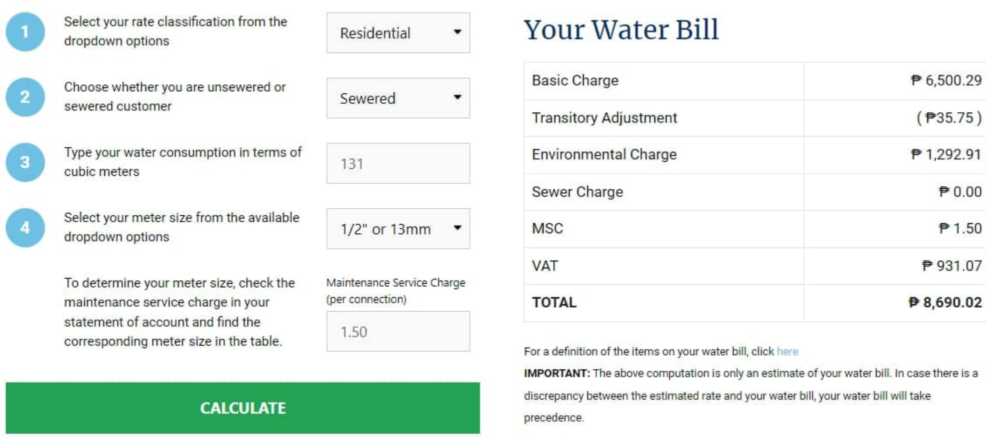


Figure 5.9: Maynilad Bill Calculator Results

5.2 Discussion

5.2.1 Accuracy improvements

Before the researchers could test the accuracy of the system, they first controlled the quality of the input images. This is because, the test dataset consists of Kucev's Dataset taken from Kaggle.

This means that the dataset to be inputted into the accuracy test had to be a *Data-Centric AI*. A Data-Centric AI means that the input dataset should be suited to how the AI should act [13]. In the researchers' case, each image of the water meter is expected to not be tilted, and be clear. This is because, in real-life application, the images will be captured by a properly mounted ESP32-CAM on the water meter. Doing this, the amount of images from the dataset to be inputted into the accuracy test went from 1244 images to 706. An example of an image that does not correspond to a Data-Centric AI for this particular system can be seen in Figure 5.10.

Figure 5.10: Sample of a non Data-Centric AI image

Initially, before changing the input images, the accuracy of the system was only 49.759%. Making the input into a Data-Centric AI by removing tilted and unclear water meter images, the researchers were able to increase this accuracy to 69.122%. However, this was unsatisfactory to the researchers.

The researchers were able to identify a common problem with the wrong predicted readings. It is when a digit in the meter is rotating to a different digit, the digit reading model becomes inaccurate. Additionally, the researchers were able to gather data indicating that among all the predicted readings that had a 1 digit disparity, 80.645% of those were from the last digit. An example of this can be seen in Figure 5.11. This is because, the last digit frequently rotates due to it being the smallest number. Due to the researchers prioritizing a test for accuracy from an image to a predicted reading, they ignored this edge case scenario. The researchers then made the stricter Data-Centric AI, which lead to the removal of images that contained an error only in the last digit from the accuracy test. This resulted in a greater accuracy of 83.993% for the system, but this reduced the total number of images from 706 to 581.

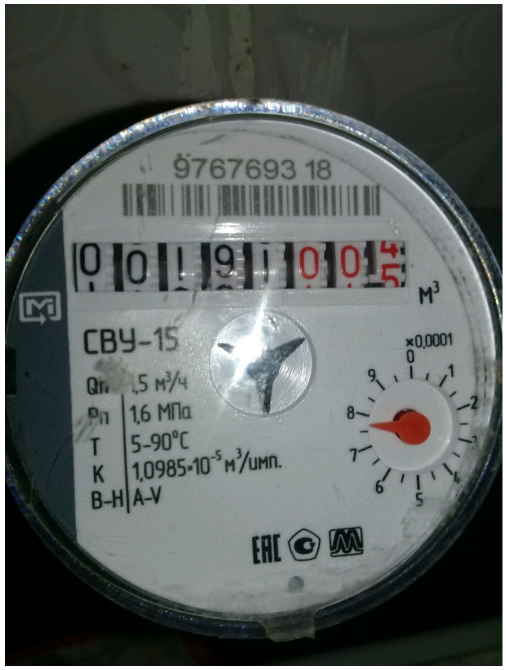


Figure 5.11: Sample image of the last digit rotating

A summary of the progression of accuracy for the model can be seen in Table 5.4.

	Original	Data-Centric AI	Stricter Data-Centric AI
Model Accuracy:	49.759%	69.122%	83.993%
Total Number:	1244	706	581
Total Correct:	619	488	488
Total Error:	625	218	93
Single Digit Error	334	155	30
Single Error only in Last Digit:	280	125	-

Table 5.4: Summary of the accuracy of the system's results

5.2.2 Set-up for our Validation Dataset

The researchers first had to encase the ESP32-CAM together with the FTDI adapter. The ESP32-CAM and the FTDI then connected through a breadboard, and had power running it using an outlet. To note, the set-up allows for a power bank, or an outlet, or batteries with 2-pins to power it. Shown in Figure 5.12, is the researchers ESP32-CAM and FTDI adapter encased, in a plastic container.

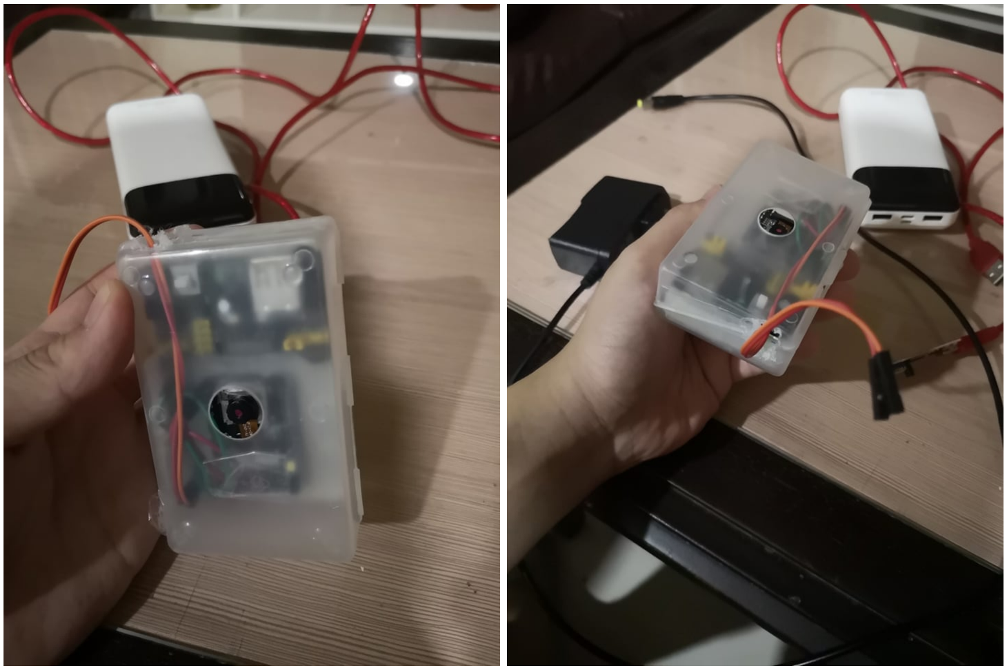


Figure 5.12: Images of the ESP32-CAM encased in a plastic container

Next, in Figure 5.13, the researchers show their set-up used in order to capture images of the water meter. In Figure 5.14, the captured image is shown along with the result after it was evaluated through the system.

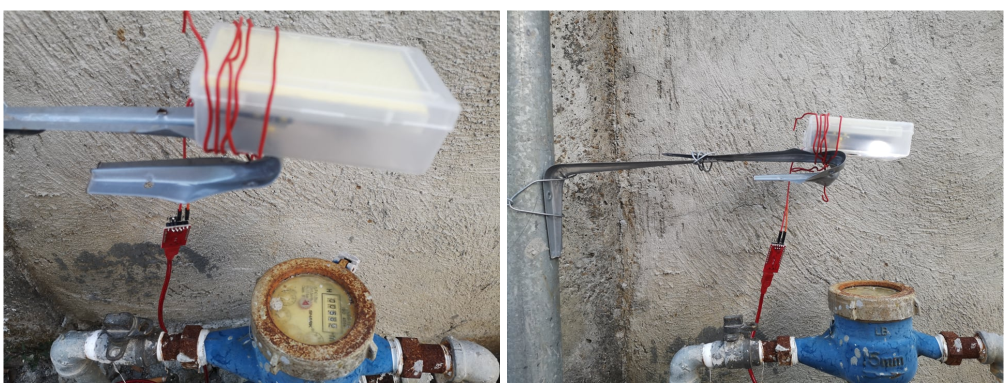


Figure 5.13: Researchers full set-up of a mounted ESP32-CAM

Figure 5.14: ESP32-CAM captured image with its results

Chapter 6

Recommendations for Future Works

Though the results of the project are deemed satisfactory by the researchers, it can still be improved. This section highlights key points for improvements of the system.

First, the usage of Kucev's water meter dataset had a satisfactory result, given its 83.993% accuracy. However, to further improve this, a dataset should be created using the water meters found in the Philippines. Additionally, the ESP32-CAM continuously drained power. This can be improved by having it only turn on when it is about to capture images. For setups that are not powered through an outlet, this can be useful especially for battery powered setups. Another recommendation is to create an alternative to the ESP32-CAM's reliance on Wi-Fi. This can be useful for places where there is no Wi-Fi connection available. Next, the ESP32-CAM captures images for consumers, however, consumers may want to upload their own captured images of different water meters. With that said, manual uploading should be an option. Another one is to create a better and a more mobile visualization for data as an alternative to the IoT dashboard. Here, the researchers recommend the use of a mobile app. Next, is to address the issue of a digit rotating to another digit. For this, the researchers recommend that the system should indicate if it is unsure of a digit. This can be reflected in the IoT dashboard. Finally, due to the different methods for calculating water bills that vary from each water company, the researchers strongly recommend that the system should be able to freely switch in its water bill estimation.

In summary, these are the recommendations:

- Create a dataset of water meters in the Philippine setting
- The ESP32-CAM should only be powered on when it is about to take a picture.
- Have an alternative way for image sending to a server, without reliance on Wi-Fi
- Have a mobile app for a better data visualization
- Be able to indicate if the system is unsure of a digit
- Manual uploading of images without the use of the ESP32-CAM Module by the user
- Have an option to indicate what water company a household is for the estimated water bill

Chapter 7

Conclusion

The researchers were able to create an alternative to manual water meter reading for residential households. Using an ESP32-CAM Module as a way to capture images of a household's water meter, the researchers created a system of cascading deep learning models that takes an input image, and outputs a predicted reading. The system created boasted an accuracy of 83.993%, a Macro F1 score of 0.969490, and a Weighted F1 score of 0.977804, proving its reliability. This system used this predicted reading to output of an estimated water bill to an IoT Dashboard seen in the website - *things.ph*.

This system provides a cost-effective way of monitoring a household's water consumption. The system's ability to automate this process lessens data inconsistencies, human error, and removes the need for manual labor. This is all done using the ESP32-CAM module that is priced at less than $10. The estimated water bill and reading from the water meter is easily viewed in the IoT Dashboard in *things.ph*.

Indeed, the system offers a better alternative to the traditional method of manual water meter reading in the Philippine setting. Thus, providing a foundation to a more automated and transparent approach not only to water consumption monitoring but also to other similar applications that rely on meters such as electricity.

Chapter 8

Bibliography

[1] C. Gonzales, *Mwss probes claims of alleged errors in water bills*, 2020. [Online]. Available: https://newsinfo.inquirer.net/1296722/mwss-probes-claims-of-alleged-excessive-increases-in-water-bills.

[2] R. Domingo, *Mwc, maynilad suspend meter reading*, 2020. [Online]. Available: https://business.inquirer.net/293329/mwc-maynilad-suspend-meter-reading.

[3] Y. Peng and Z. Chen, "Application of deep residual neural network to water meter reading recognition", in *2020 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA)*, 2020, pp. 774–777. DOI: 10.1109/ICAICA50127.2020.9182460.

[4] G. Salomon, R. Laroca, and D. Menotti, "Deep learning for image-based automatic dial meter reading: Dataset and baselines", in *2020 International Joint Conference on Neural Networks (IJCNN)*, 2020, pp. 1–8. DOI: 10.1109/IJCNN48605.2020.9207318.

[5] M Suresh, U. Muthukumar, and J. Chandapillai, "A novel smart water-meter based on iot and smartphone app for city distribution management", in *2017 IEEE Region 10 Symposium (TENSYMP)*, 2017, pp. 1–5. DOI: 10.1109/TENCONSpring.2017.8070088.

[6] A. Ray and S. Goswami, "Iot and cloud computing based smart water metering system", in *2020 International Conference on Power Electronics IoT Applications in Renewable Energy and its Control (PARC)*, 2020, pp. 308–313. DOI: 10.1109/PARC49193.2020.236616.

[7] H. Puttnies, V. Altmann, F. Golatowski, and D. Timmermann, "Cost-efficient universal approach for remote meter reading using web services and computer vision", in *2015 IEEE 16th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, 2015, pp. 1–6. DOI: 10.1109/WoWMoM.2015.7158205.

[8] K. Roman, *Water meters dataset*, 2020. [Online]. Available: https://www.kaggle.com/tapakah68/yandextoloka-water-meters-dataset.

[9] D. Workshop, *Getting started with the esp32-cam*, 2020. [Online]. Available: https://dronebotworkshop.com/esp32-cam-intro/.

[10] K. He, G. Gkioxari, P. Dollár, and R. Girshick, *Mask r-cnn*, 2018. arXiv: 1703.06870 [cs.CV].

[11] J. Korstanje, *The f1 score*, 2021. [Online]. Available: https://towardsdatascience.com/the-f1-score-bec2bbc38aa6.

[12] *Bill calculator - maynilad water services, inc.*. [Online]. Available: https://www.mayniladwater.com.ph/bill-calculator/.

[13] S. Mukherjee, *Why andrew ng favours data-centric systems over model-centric systems*, 2022. [Online]. Available: https://analyticsindiamag.com/why-andrew-ng-favours-data-centric-systems-over-model-centric-systems/.