ระบบสังเกตการณ์การใช้น้ำด้วยการประมวลผลภาพ
Water utilization monitoring system by using image processing

ลุคมาน ไชยราบ[1*], เการาบ รุงเยว[1], ศิริกานต์ ชูเชิด[2]
Lukman Chaiyarab[1*], Gaurav Dhungyel[1], Sirikan Chucherd[2]

Received: 23 January 2021 ; Revised: 15 February 2021 ; Accepted: 1 March 2021

บทคัดย่อ

การขาดแคลนน้ำเป็นปัญหาใหญ่ในฤดูแล้งทั่วโลก หลายประเทศพยายามจัดการและลดการใช้น้ำให้น้อยที่สุด การบริหารจัดการน้ำจึงเป็นสิ่งที่สำคัญที่จะช่วยลดปัญหานี้ได้ ตัวอย่างเช่นการหมั่นตรวจสอบความสมบูรณ์ของระบบประปาไม่ให้เกิดการชำรุดเสียหาย คณะผู้แต่งจึงมีแนวคิดในการประยุกต์ใช้เทคโนโลยีการประมวลภาพ เพื่อตรวจสอบหน้าปัทม์มิเตอร์น้ำประปา รูปภาพมิเตอร์ถูกตระเตรียมและกำจัดสัญญาณรบกวนด้วยตัวกรองแบบเกาส์เซียน จากนั้นภาพที่กรองแล้วจะถูกแปลงจากภาพสีเป็นภาพขาวดำด้วยวิธีการของออสสุ จากนั้นสัณฐานวิทยาถูกนำไปใช้เพื่อแยกส่วนที่สนใจ (ROI) สุดท้ายวิธีการ Python-tesseract ซึ่งเป็นเครื่องมือจดจำอักขระด้วยแสง (OCR) ถูกนำมาใช้เพื่อแปลงสตริงเป็นข้อความ(ตัวเลข) หลังจากนั้นผลลัพธ์จะถูกส่งผ่านระบบ WIFI เพื่อ ไปจัดเก็บและแสดงผลแบบกราฟบนหน้าเว็บ หากเกิดความผิดปกติของข้อมูลการใช้น้ำ จากการอ่านค่ามิเตอร์ ระบบจะทำการส่งอีเมลแจ้งเตือนไปยังหน่วยงานที่เกี่ยวข้อง ผลการทดลองแสดงให้เห็นว่าระบบที่นำเสนอมีความแม่นยำด้วยเปอร์เซนต์เฉลี่ยของความแม่นยำสำหรับภาพแต่ละประเภท (มีรอยขีดข่วนมาก น้อย และไม่มี) คือ 48%, 75.1% และ 91.64% ตามลำดับ

คำสำคัญ: คอนโซลไฟเบส การจดจำอักขระด้วยแสง ราสเบอรีไพ ระบบบริหารจัดการน้ำ

Abstract

Water scarcity is a big problem in dry seasons around the world. Many countries are trying to manage and minimize the use of water and Water management is essential to help reduce this problem ; For example, by regularly checking the integrity of the plumbing system to prevent damage. the authors have an idea for applying image processing technology to check the water meter page. Meter images were prepared and noise eliminateds by a Gaussian filter. Then, the filtered images were binarized from color images by Otsu's algorithm. Next, the morphological was applied to extract the region on interest (ROI). Finally, the Python-tesseract which is an optical character recognition (OCR) tool was implemented to convert string to text(numbers). After that the results were be sent via WIFI to be stored and displayed in graphs on a web page. If there is any abnormality in water consumption data from the meter reading, the system will send notification emails to the relevant department. The experimental results showen that the average percentage of recognition accuracy for each image types was 48%, 75.1%, and 91.64% respectively.

Keywords: Firebase console, Optical Character Recognition (OCR), Raspberry pi, Water Management System

[1] นักศึกษาปริญญาตรี, ศูนย์ความเป็นเลิศด้านปัญญาประดิษฐ์และเทคโนโลยีก่อกำเนิด, สำนักวิชาเทคโนโลยีสารสนเทศ, มหาวิทยาลัยแม่ฟ้าหลวง, เชียงราย, 57100, ประเทศไทย

[2] ผู้ช่วยศาสตราจารย์, ศูนย์ความเป็นเลิศด้านปัญญาประดิษฐ์และเทคโนโลยีก่อกำเนิด, สำนักวิชาเทคโนโลยีสารสนเทศ, มหาวิทยาลัยแม่ฟ้าหลวง, เชียงราย, 57100, ประเทศไทย

[1] Undergraduate students, Center of Excellence in AI and Emerging Technology, School of infornation technology, Mae Fah Luang University, Chiang Rai, 57100, Thailand

[2] Asst.Prof, Center of Excellence in AI and Emerging Technology, School of infornation technology, Mae Fah Luang University, Chiang Rai, 57100, Thailand

* Corresponding author ; Lukman Chaiyarab, School of information technology, Mae Fah Luang University, Chiang Rai, 57100, Thailand. manchai-yarab@gmail.com

Introduction

Water and minerals are essential for living. Without water it is impossible for living organisms toesurvive. Water is the basic component oo life. It is used for drinking, cooking, cleaning our bodiey and many more purposes. Our body requires water to cellular function and to operate the organs inside our body. Therefore, we must learn how to manage the usage of water and use this resource efficiently. The basic principle of using water efficiently is that the user should be using an approximate amount of water for doing any kind of activity. For instance, when the reader brushes teeth in the restroom, we recommendethat water is not allowed to flow whileeperforming this activity. By stopping the flow of water, people can reduce thasvolume of excess water going to wastd in a year. The objectives of this research were to monitor and control the water usage of a building by using image processing technology to extract the number of the water meter that represents water usage. If there are any unexpected changes, the system will send a notification e-mail to the relevant department. the model used in this study consistes of three parts. The first part wis the image preparation. The water meter images were taken and processed for noise reduction. Then, color imagesewere converted and threshold to binary images. Morphology technique wis applied to get the preprocessed images. The Second part wis the character recognition process. Optical Character Recognition (OCR) by Pytesseract was implemented to extract the water meter imageseas numbers. After successfully extracting the numbers, the third part of the system wis automatically sending the water meter data to be displayed as a graph on a web page. If there were any unexpected changes in the usage of water, the system will send a notification e-mail to the water resource department.

Literature Review

The most accurate Optical Character Recognition (OCR) engine is Pytesseract (Lee, 2020e. many languages and numbers can be recognized by this OCR engine. the licence was released under the Apache V2.0 open source. The development of the software is sponsored by Google.

In the area of water monitoring systems, there are a number of organizations that has operated the application, particullarly in the field of surveillence technologies. For instance, the approach for remote capturing of a water meter involves a mechanical water meter, camera, digital media processor, GSM modem and TFT display (Kashie *et al.,* 2015).dThe approach works by requesting action from the central server. The camera was activated to capture images and then they were processed by DSP processor to extract a meter reading. In addition, the extracted meter reading was sent to the server by GSM network.sAlso, their experiment used a DSP processor for implementation which is expensive and diffcult to code and implement.

Gosavi *et al.* (2017) proposed a model that consisted of Raspberry pi and Arduino Uno. They were used with the Hall effect sensor to measure the flow rate of water. The Raspberry pi uploaded data onto cloud infrastructure where databases are set. The end user is enabled to view the data via the web interface. Nevertheless, the system is different from the one we proposd since they measured the flow rate and collected data. Futhermore, the flow rate is not always constant and might cause some inaccuracies in reading.

Triantor *et al.* (2014) implemented with Matlab R2011b software toolbox where the application run with Windows 7 operating system an, Pentium Dual Core 3.00 Ghz processor. Capturing images with Lenovo mobile Phone A800 taken from 10:00 Am to 15:00 Pm in sunny weather conditions. Firstly, three-element of colors Red Green Blue (RGB) images are processed by converting them into gray scale images and then using a median low-pass filter, contrast enhancement, histogramm equalization and binarization to obtain a black and white binary image. At a later time, a segmentation process is performed to get the number as a region of interest (ROI) from line segment and the background. However, the system is manually operated and needs additional software toolsdans more hardware equipmens.

Sisinn, *et al.* (2015) made a model using smartglass (Vuzix M100) to automate the meter reading. Particularly, it was divided into three parts. First, it automatically recognised the universal barcode that is acquired by the smartglass camera. Secondly, the Pytesserract application performed the recogition of digits meter via SmartGlass. In The last step, an internet connection was needed in order to download a read out into a cloud-based respository. The drawback is that the recognized text is mismatched with the actual text that was on the meter.

Somyat & Nakariyakul (2018) proposed the lottery number reader from Tesseract engine digit recognition for blind people by mobile phone appliction (LottoTU). They segmented, thresholded, and converted the lottery number images into text by the Tesseract engine (OCR). The recognised numbers are read by Android text-to-speech (TTS) API. they achieved a satisfying result by capturing images at six locations under different Illumination conditions, orienation and distance. 59 out of 60 test images were correctly read out lound for bliness seller. However, LottoTU failed to detect images with dark or light conditions.

Another system (Fikejz & Rolecek, 2018) used the Raspberry pi zero which is a single core mini computer expanded by a real-time unit (RTC) using the standard Linux Raspbian environment. The equipment provides information of flow rate from the water meter. dAlternatively, a software system build in a Java plateform by using MySQL database and Pi 4 were used to evaluate the leakage detection. However, the system focuses on the flow rate via the inner pipline not the meter reading.

Methodology

The proposed system diagram is shown in Figure 1. The preparation of the image works by requesting action from the central server (left hand side of the Figur.1. The Raspberry Pi v2 camera module will activate to capture images from water meter. Later those images are processed by Raspberry Pi 4 to extract the meter reading. The region of interest is converted from string to text. Finally, the results are sent over the internet to be stored in two databases and displayed in the webpage. In case of a possible water leak situation, a message will be sent via E-mail to the concerned authority.

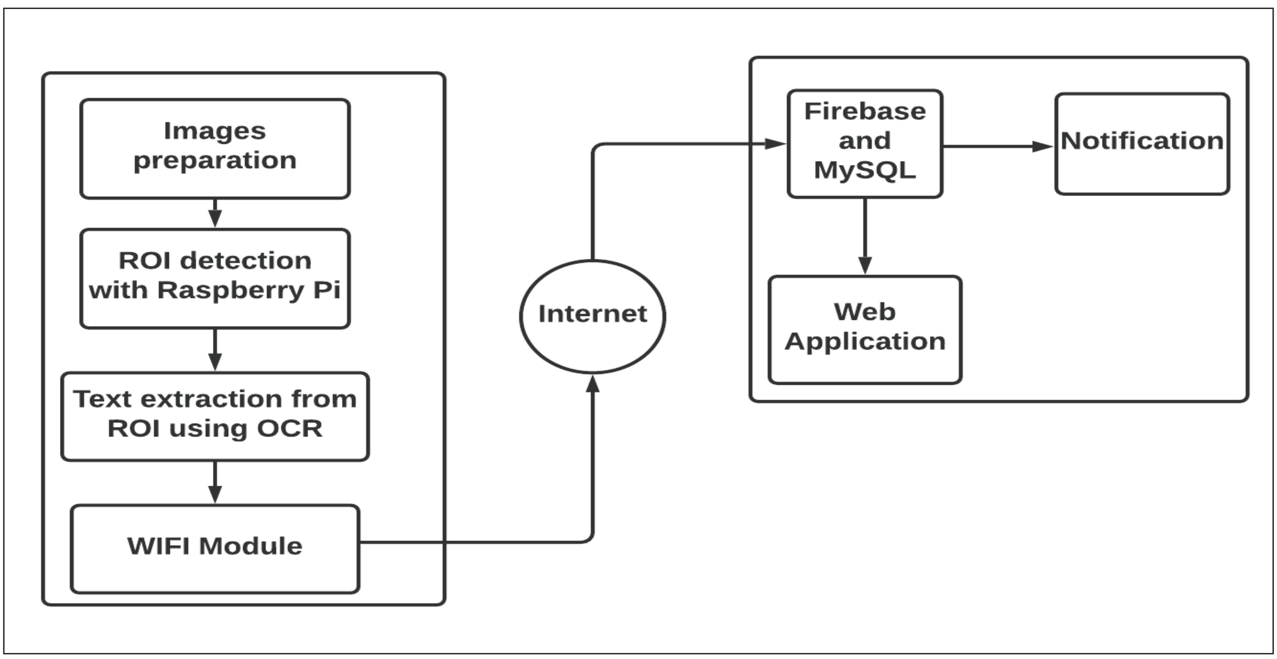


Figure 1 System diagram.

Image Processing

Figure 2 explains about the image preparation steps through the character recognition. The system captures the images of the water meter using the Raspberry-Pi camera module V2. Then, the captured images are pre-processed. Firstly, the RGB image is converted to a grayscale image. Secondly, the system specifies the width and height of the kernel. This system works best with a (3,4) kernel. Third, Gaussian filtering is applied since it is very effective in removing noise from the image. Fourth, Otsu's thresholding is used for image binarization. The value of the threshold is determined automatically. Next, morphological erosion is applied so that the size of the foreground object increases. Morphological dilation is also used to discard the pixels near the boundary to get a clearer binary image. Finally, the region of interest (ROI) is extracted by using the Pytesseract Optical Character Recognition (OCR). It works better with black extracted text anda white background. It inverses the process to get the desired text results.

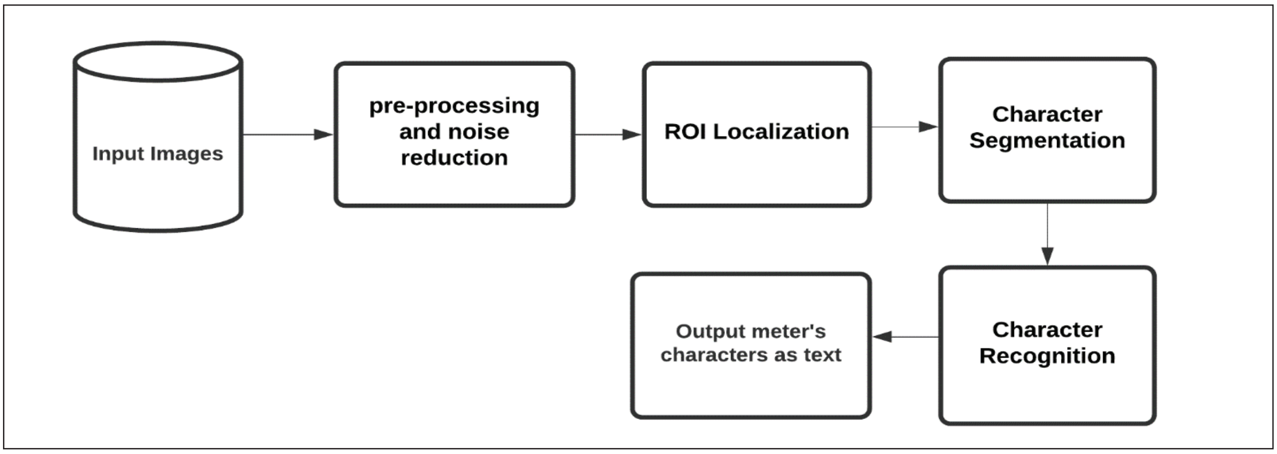


Figure 2 Image processing.

Data storage and display

Two back end programs MySQL and Firebase database which are maintained by Google are used for the data storage of the system. After the completion of image processing, the system transfers the fetched output to the above-mentioned programs as shown in Figure.3. The program initializes the connection with two of these databases. Then, it saves the text along with the date and time. For web application, the Google chart is used to plot result data into graphs. The reason for using google charts is because of its simplicity in terms of coding, and modifying, making it more attractive. The charts are based on pure HTML5 technology. They are interactive and zoomable. There are a few simple steps to add this chart to web page.

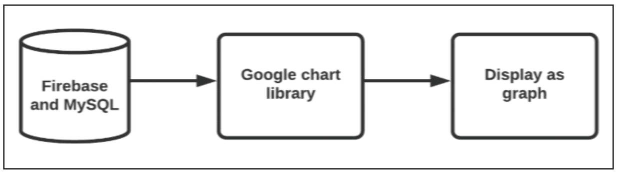


Figure 3 Data storage and display.

User Authentication and Login

For client side, PHP is implemented for the user authentication and login. Firstly, a new user has to be registered to the system by the admin so that only authorized people can accessmit. After that the system verifies the username and password by comparing the values stored in a database to the input value. After the verification is complete, the user is redirected to the main page of the system.

Notification

MySQL is used because it is easier to code and implement. The system retrieves the last data along with the 2nd last data and calculates the difference to gain a view of how much water has been consumed during the time frame i.e. in one hour throughout the day. By the end of the day there will be 24 records. The reason for making it hourly is because we can quickly identify if there is a problem because the data comes hourly, making it daily or monthly will be too late if there is a problem in the pipes or if there is over-usage. The consumption water data at 10 cubic meters is used as the threshold because it is the average amount of normal water usage of that building. If there is a possible leakage somewhere in the pipes, an alert e-mail is sent right away to the authorities in that case to check for a possible leakage.

Results and discussions

System breakdown

The system comprises a Raspberry Pi 4 with 3GB Ram and a Pi v2 Camera Module. Raspberry Pi 4 has an ARM Cortex-A72 processor. The Pi 4 is also powered via a USB-C port, enabling additional power to be provided to downstream peripherals, when used with an appropriate PSU. The v2 Camera Module has a Sony IMX219 8-megapixel sensor which is capable of taking very high-resolution pictures which is exactly what is required for the system.

Implementation of the system

The system is composes of a Raspberry Pi and Raspberry Pi camera enclosed within a box with fixed lighting conditions. Therefore, the variation of lighting during the day will not alter the system and The brightness inside the box remains constant. The system is placed above the meter and the location depends on the user. The outer box is made of PVC and it is covered with a black cloth to prevent exterior light from outside the box. The dimensions of the box are 15cm*11cm*10cm which are the length, the width and the height of the box respectively. Every component inside the box is tightly sealed to make it to remain functional in harsh weather. The system has to be placed perpendicular to the water meter at an approximate height of 7 cm. The side view and the top view of device installation are as shown in Figure 4 (a) and (b) respectively.

Image processing

This section shows example images of the experimental results on the real water meter. First, a raw image of the water meter in Figure 5(a) is captured by the Raspberry Pi camera.

(a) (b)

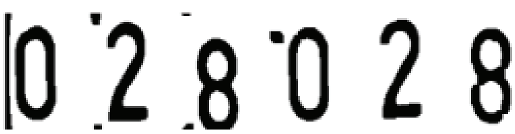
Figure 4 Device installation. a) Side view and b) top view.

Next, the result of the conversion of raw input image to gray scale image and tilt by 90 degrees to get a perpendicular view is shown in Figure 5(b). Then, the Gaussian filtering technique and the Otsu's thresholding are applied to remove excess noise (Figure 5(c)). After that, the ROI is extracted by performing the morphological operations as in Figure 5(d). Finally, the Pytesseract OCR converts string to text and the result is presented in console (see Figure 5(e)).

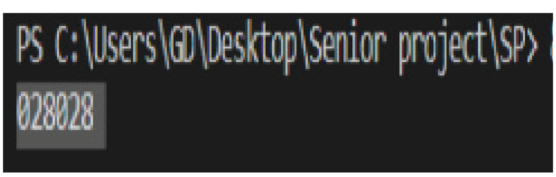
a) Input image

b) Gray scale image

c) Binarized and threshold image



d) ROI extracted image



e) Output in console

Figure 5 Experimental results. a) Input image, b) gray scale image, c) binarized and threshold image, d) ROI extracted image, and e) output in console.

Experimental results

This section shows the experimental results on 302 of water meter images. The region of interest (ROI) in this system is only the first 6 digits of the water meter. The seventh digit was ignored because it is the least significant part. The segmentation results were evaluated based on the recognition accuracy as

$$Recognitionaccuracy\% = \frac{RecognizedImages * 100}{TastingImages}$$

Table 1 presents the accuracy of input images in recognizing the digits. The experiment was divided into three types depending on the noise level of the images. Table 2 shows the categories of images of each type. The number of testing images of each number varies from 0-9 and depends on the occurrence in the water meter. The average percentage of the recognition accuracy for each image type (major, minor, and no scratches on the glass) are 48%,75.1%, 91.64% respectively. The results are better when the level of noise in the image is low. The lower the noise on the image, the better the result.

Web Application

The main page of the web application is shown in Figure 6. The example of recognition results will be stored and retrieved from Firebase DB shown in Figure 7. These data are plotted as a graph in Figure 8 using the Google charts library. The username and password for all users are assigned by admin. All the passwords are securely stored in MySQL. Once a user logs in to the system, the user can view the statistics for the water consumption of each building. In the case of an unusual situation, such as the water leak, a notification E-mail will be sent to the responsible staff as show in Figure 9. For more information, please access this link (https://youtu.be/CYjQt993toA).

Table 1 An example of each category of images to be tested.

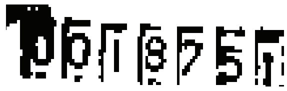
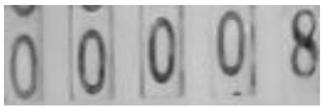
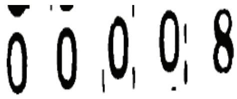
	Region of interest	Noise reduction on ROI	Comments
Water meter with minor scratches on the front glass			Minor noises. Difficult to remove the lines in between. Unable to read 0.
Water meter with major scratches on the front glass			Major scratches and erosion 0 is non readable even after noise reduction
Water meter images with no scratches on the front glass			No noise. All the numbers are distinctly recognizable. A new water meter

Table 2 Accuracy measurement for 302 water meter images**.**

Type of image	Digits	No. Testing Images	No. of Recognized Images	Recognition accuracy
Water meter with major scratches on the front glass	0	10	6	60%
	1	15	10	66%
	2	12	6	50%
	3	12	4	33%
	4	7	3	42%
	5	8	4	50%
	6	10	3	33%
	7	7	3	42.8%
	8	6	5	83%
	9	8	2	25%
Water meter with minor scratches on the front glass	0	7	6	85.7%
	1	9	9	100%
	2	11	9	81.8%
	3	14	11	78.5%
	4	13	10	76.92%
	5	12	8	66.67%
	6	7	4	57.14%
	7	9	6	66.67%
	8	11	8	72.72%
	9	12	8	66.67%
Water meter images with no scratches on the front glass	0	10	10	100%
	1	7	7	100%
	2	9	9	100%
	3	10	10	100%
	4	11	12	91.6%
	5	16	14	87.5%
	6	10	9	90%
	7	9	7	77%
	8	13	11	84.6%
	9	7	6	85.7%

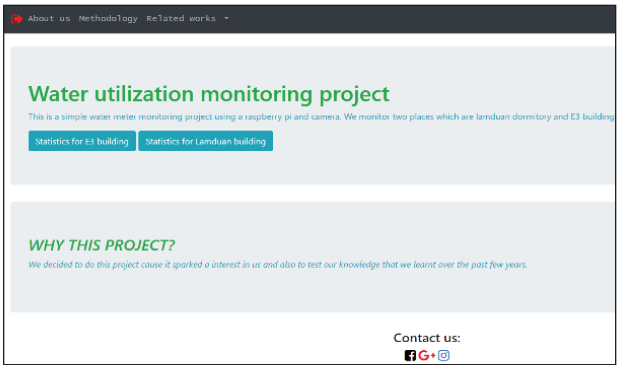


Figure 6 Main page of web application.

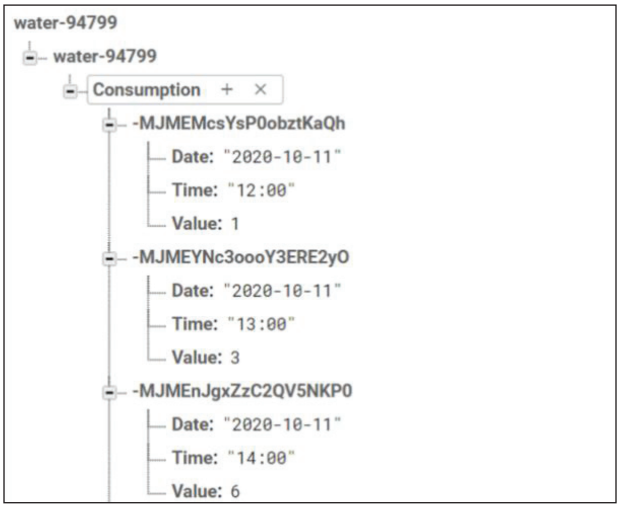


Figure 7 Data store in Firebase.

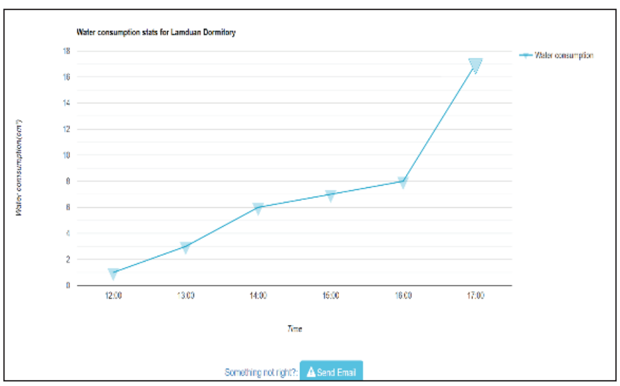


Figure 8 Water consumption graph of each building.

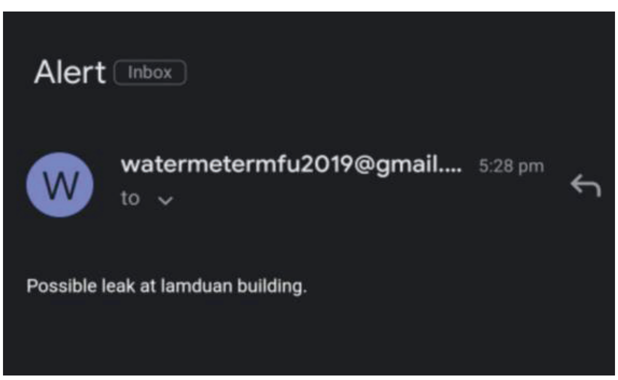


Figure 9 Email Notification.

Conclusion

This paper proposes a simple and efficient concept to read a water meter and monitor its usage. The framework first makes use of image preprocessing techniques followed by Optical character recognition (OCR). After the digits of a water meter have been detected, the graph of water utilization is presented via a web application. In the case of excess usage of water, the system sends a notification E-mail to the staff responsible. The experimental results demonstrated that the system achieved good performance with a low noise water meter.

From the experimental results, it is safe to conclude that the proposed system is capable of detecting numbers on water meters and calculating usage of water. It is suitable for those meters which have less noise i.e., new meters yield better results. The pi camera and raspberry pi 4 is capable of fast processing. Moreover, the software the system uses are easy and great sources to build webpages as well as sending notifications. They are reliable tools that work well without technical error and are free for everyone.

However, the condition of the water meter plays an important role in the output of the meter. More scratches minimize the accuracy of digit recognition. Apart from that, the proposed system is integrated in a robust way where it continuously captures images with system code to work hourly. With great accuracy in good condition meters, the system needs low maintenance with the physical hardware equipment (raspberry pi board), but the users should consider checking and maintaining the raspberry camera and network connection in order to make the system work properly. Lastly, it is budget friendly for any people who is interested to invest with in system. Also the system can be modified for other meters such as gas and electricity meters. It is easy to implement and build the system.

Acknowledgement

The author would like to express our gratitude to our advisors (Dr. Sirikan Chucherd, Dr. Tossapon Boongoen and Dr. Surapol Vorapatratorn) without their supports this work would not have been possible. Their immense support along with our hard work made this project possible. We would like to acknowledge the Center of Excellence in AI and Emerging Technology, School of Information Technology of Mae Fah Luang University for allowing us to conduct our experiments without any interference and giving all supports and guidance they can.

References

Depari, A., De Dominicis, C.M., Flammini.A., Sisinni, E., Fasanotti, L. and Gritti, P. (2015). *Using smartglasses for utility-meter reading*. 2015 IEEE Sensors Applications Symposium (SAS), Zadar. https://ieeexplore.ieee.org/document/7133649

Fikejz, J. and Roleček, J. (2018, May 21-23). *Proposal of a smart water meter for detecting sudden water leakage*. 2018 ELEKTRO. Mikulov. https://ieeexplore.ieee.org/document/8398316

Gosavi, G., Gawde, G. and Gosavi, G. (2017). Smart water flow monitoring and forecasting system. *2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), Bangalore*. https://ieeexplore.ieee.org/document/8256792

Kashid, S.G., Pardeshi, S.A., Sirsat, S.D. and Bonge, V.H. (2015). Remote capturing of water meter reading using DSP processor. *Third International Conference on Image Information Processing (ICIIP), Waknaghat*. https://ieeexplore.ieee.org/document/7414738

Lee, M. (2020). *Pytesseract 0.3.6. PYPI*. https://pypi.org/project/pytesseract/

Somyat, N. and Nakariyakul, S. (2018). Thai Lottery Number Reader App for Blind Lottery Ticket Sellers. *10th International Conference on Knowledge and Smart Technology (KST), Chiang Mai*. https://ieeexplore.ieee.org/abstract/document/8426096

Triantoro, T.R., Batubara, F. and Fahmi, F. (2014). Image based water gauge reading developed with ANN Kohonen. *International Conference on Electrical Engineering and Computer Science (ICEECS), Kuta*. https://ieeexplore.ieee .org/document/7045223.