

# DIGITISATION OF CONVENTIONAL WATER METERS USING AUTOMATED NUMBER RECOGNITION

A THESIS SUBMITTED TO AUCKLAND UNIVERSITY OF TECHNOLOGY  
IN FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF  
MASTER OF ENGINEERING

Yi Gao

School of Engineering, Computer and Mathematical Sciences

August 2019

# Attestation of Authorship

I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been accepted for the qualification of any other degree or diploma of a university or other institution of higher learning.



---

Signature of candidate

# Acknowledgements

First of all, I would like to convey my appreciation to all those who helped me during this research project. This thesis will not complete without their cooperation and stimulation. My deepest gratitude goes to my supervisor Dr Xuejun Li for his constant encouragement and guidance. He advised me over every problem of this project, from topic selection to system design until the final process of writing.

I must also thank Senior Technician Jian Huang for helping me with hardware design. From hardware selection to circuit design, his help was irreplaceable.

Last but not least, I would like to thank my beloved family for their backup and great support in me all through this year. I would also like to express my gratitude to my friends and classmates, who gave me help to work out my problems, for which I will never forget.

Yi Gao

August 2019

# Abstract

Nowadays, New Zealand families are still using traditional analog water meters. Even if the traditional water meter has the characteristics of accuracy measurement precision, simple structure, zero power dissipation and high reliability, it still needs a worker to record the meter readings door-by-door. With the development of image processing technology, people spend a large amount of research in the field of character recognition, a growing number of character recognition products are applied to industrial production and daily life. In this thesis, an accessory will be designed for water meters by using optical character recognition technology to discern the meter reading. This accessory can translate the analog meter reading into a digital signal, prepare for the automated meter reading system, and Internet of Things in the future. The accessory will sit on top of a conventional water meter, using a camera to capture the number, and display the reading on an additional LCD, which can be sent wirelessly to a database server. This provides a significant alternative meter reading, reducing human resource waste. The contribution of this thesis is a novel light-weight optical character recognition algorithm and its implementation in a prototype with embedded digital system. Experimental results show that the reading accuracy can be as high as 95%, which indicates a promising candidate technology for automated meter reading.



# Contents

<b>Attestation of Authorship</b>	<b>ii</b>
<b>Acknowledgements</b>	<b>iii</b>
<b>Abstract</b>	<b>iv</b>
<b>List of Tables</b>	<b>vii</b>
<b>List of Figures</b>	<b>viii</b>
<b>Glossary and Notations</b>	<b>x</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Background . . . . .	1
1.2 Objective . . . . .	3
1.3 Research Questions . . . . .	3
1.4 Research Contribution . . . . .	4
1.5 Thesis Organisation . . . . .	4
<b>2 Literature Review</b>	<b>6</b>
2.1 Optical Character Recognition . . . . .	6
2.2 Image Processing Algorithm . . . . .	7
2.2.1 Image Binarization . . . . .	7
2.2.2 Noise Removal . . . . .	9
2.2.3 Image Segmentation . . . . .	12
2.3 Recognition Algorithm . . . . .	14
2.3.1 Template Matching . . . . .	15
2.3.2 Geometric Moments . . . . .	18
2.3.3 Deep Learning . . . . .	21
2.4 Hardware . . . . .	24
2.4.1 Water Meter . . . . .	24
2.4.2 Camera . . . . .	29
2.4.3 Microcontroller Unit . . . . .	33
2.5 Chapter Summary . . . . .	35

<b>3</b>	<b>Proposed Design</b>	<b>36</b>
3.1	Pre-processing Method . . . . .	37
3.1.1	Noise Remove Method . . . . .	38
3.1.2	Image Binarization Method . . . . .	39
3.1.3	Image Segmentation Method . . . . .	39
3.2	Recognition Method . . . . .	40
3.2.1	Template Matching Algorithm . . . . .	40
3.2.2	Projection Template Matching Algorithm . . . . .	41
3.2.3	Geometric Moments Algorithm . . . . .	42
3.2.4	Proposed Moment Template Matching Algorithm . . . . .	43
3.3	Chapter Summary . . . . .	45
<b>4</b>	<b>Prototype Implementation</b>	<b>46</b>
4.1	Hardware Design . . . . .	46
4.1.1	Micro-controller Module . . . . .	47
4.1.2	Camera Module . . . . .	48
4.1.3	Liquid Crystal Display Module . . . . .	51
4.1.4	Main Communication Interface . . . . .	52
4.1.5	Printed Circuit Board Design . . . . .	55
4.2	Software Design . . . . .	61
4.2.1	Hardware Initialization . . . . .	62
4.2.2	Image Data Collection . . . . .	65
4.2.3	Noise Removal Function . . . . .	67
4.2.4	Binarization and Segmentation . . . . .	69
4.2.5	Recognition Function . . . . .	70
4.3	Chapter Summary . . . . .	71
<b>5</b>	<b>Results and Discussions</b>	<b>72</b>
5.1	Character Template . . . . .	72
5.2	Experiment . . . . .	74
5.2.1	Experimental Environment . . . . .	74
5.2.2	Experiment Design . . . . .	75
5.2.3	Image Data Collection . . . . .	75
5.2.4	Microcontroller-based OCR . . . . .	79
5.3	Results Analysis . . . . .	83
5.4	Chapter Summary . . . . .	84
<b>6</b>	<b>Conclusion and Recommendation for Future Work</b>	<b>85</b>
6.1	Conclusion . . . . .	85
6.2	Recommendation for Future Work . . . . .	86
6.2.1	System Improvement . . . . .	87
6.2.2	AMR Expect In The Future . . . . .	88

# List of Tables

4.1	ATmega128 compared with ATmega1284P [1] . . . . .	47
4.2	OV7620 Pixel Data Bus . . . . .	49
5.1	Templates From 0 To 9 . . . . .	73
5.2	Segmentation Result For Each Number . . . . .	77
5.3	Recognition Accuracy Comparison of Different Algorithms . . . . .	79

# List of Figures

2.1	Image after Binarization Process . . . . .	9
2.2	Original and Grey Image . . . . .	10
2.3	Noise Removal Filter . . . . .	11
2.4	Template T On Top Of Input Pattern S . . . . .	15
2.5	Feed-forward Network Architecture . . . . .	23
2.6	Convolutional Neural Network Architecture [2] . . . . .	23
2.7	Recurrent Neural Networks Architecture [3] . . . . .	24
2.8	V100 PSM Volumetric Water Meter [4] . . . . .	25
2.9	OV7620 Block Diagram [5] . . . . .	31
2.10	ATmega1284P Block Diagram [6] . . . . .	34
3.1	Water Meter Digitisation Flowchart . . . . .	37
3.2	Input Image Received From The Camera . . . . .	38
3.3	Image After The Noise Removal Process . . . . .	38
3.4	Image After Binarization Method . . . . .	39
3.5	Image After Segmentation . . . . .	40
4.1	Hardware Block-diagram . . . . .	47
4.2	Output Signal Timing . . . . .	50
4.3	Direct Drive Mode Display Output Waveforms . . . . .	52
4.4	I2C Data Flow . . . . .	53
4.5	Typical Data Transmission Sequence . . . . .	54
4.6	MCU PCB . . . . .	56
4.7	I2C Bus Interconnection . . . . .	58
4.8	LED Driver Circuit . . . . .	59
4.9	LCD PCB Circuit . . . . .	60
4.10	Program Flowchart . . . . .	62
4.11	Hardware Initialization Flowchart . . . . .	63
4.12	ATmega1284p And OV7620 Connection . . . . .	66
5.1	Image Data Displayed In Text File . . . . .	76
5.2	Image Displayed In MATLAB . . . . .	76
5.3	Recognise Result . . . . .	78
5.4	Meter Reader Result 1 . . . . .	80
5.5	Meter Reader Result 2 . . . . .	81

5.6	Meter Reader Result 3 . . . . .	82
-----	---------------------------------	----

# Glossary and Notations

## Glossary

AGC	Automatic Gain Control
AMR	Automatic Meter Reading System
CCD	Charge Coupled Device
CMOS	Complementary Metal Oxide Semiconductor
CNN	Convolutional Neural Network
EEPROM	Electrically Erasable Programmable Read-Only Memory
HREF	Horizontal Valid Data Output Window
I2C	Inter-Integrated Circuit Bus
IC	Integrated Circuit
IoT	Internet of Things
LCD	Liquid Crystal Display
MLPS	Multilayer Perceptions
OCR	Optical Character Recognition
PCLK	Pixel Clock
PCB	Printed Circuit Board
RAM	Random Access Memory
RNN	Recurrent Neural Network
RGB	Red Green Blue
SCCB	Serial Camera Control Bus

---

SCL	Synchronous Clock
SDA	Synchronous Data
SRAM	Static Random Access Memory
TWI	Two Wire Serial Interface
USART	Universal Synchronous Asynchronous Receiver Transmitter
UAV	Unmanned Aerial Vehicle
VLSI	Very Large Scale Integrated Circuits
VSYNC	Vertical Sync Pulse
YUV	Luminance Chrominance Chroma

---

## Notation

$M$	matrix column number
$N$	Matrix row number
$V_{(j)}$	vertical projection value
$H_{(i)}$	horizontal projection value
$T$	template
$S$	pattern
$r_{cos}$	cosine value of the angle between vectors
$D$	Euclidean distance
$E$	error
$E_0$	error threshold value
$m_{pq}$	the moment of order $p$ and $q$
$\mu_{pq}$	centralized moments
$\eta_{pq}$	normalized central moments



# Chapter 1

## Introduction

### 1.1 Background

Internet of Things (IoT) refers to a new networking paradigm that connects everything to Internet, which features an extension of the Internet combining all sorts of equipment sensors as an ubiquitous network, at any time, any place, human and machine, communications between objects. IoT is a very popular topic now. For example, IoT technology applied in the family is a smart home system [7]. Light sensors, humidity sensors, and temperature sensors with a variety of electrical appliances in the house through a router, can be set by the software to control the air conditioning, table lamps, and curtains in the house [8].

The other application that uses IoT in the family is the Automated Meter Reading (AMR) system. AMR makes it possible for various meters to be connected to the Internet. Though homeowners cannot operate a water meter as turning on and off, they can understand the current meter reading through AMR system [9, 10].

With the development of science and technology, an AMR system will replace the traditional human transcribing. Actually, New Zealand families have been using this system on electricity meters. Power distribution companies do not need staff to come

to read the meter, and the meter reading will be automatically transferred back to the server, improving the efficiency of transcribing. Watercare, the water conservancy company in New Zealand, is still using the traditional mechanical water meter, which needs transcribing by human beings [11], with low efficiency and huge salary expenses. Applying AMR technology to the water meter, one of the important problems needing to be solved is to realise smart water meters. This provides a good opportunity for research of intelligent water meters.

The development history of digital image processing technology is not long but has brought evolutionary products to human kinds. Vision is the most important way of perception, and the image is the basis of vision. Therefore, digital image processing becomes the effective tool like psychology, physiology, and computer science aspects of visual perception. Digital image processing has wide applications in military, meteorology, industrial, medical, and automation products.

General digital image processing includes image analysis and image processing. Image processing is to transform an image into another modified image. Image analysis is to convert an image into a kind of data, such as a set of measurements or a decision, by analyzing the information of interest in the image.

Pattern recognition is closely related the recognition of images based on digital image processing and also known as an important field of image processing. The pattern recognition system is to make a computer system simulate the human senses to accept outside information, identify, and understand the surrounding environment. Pattern recognition is widely used in image recognition and speech recognition, digital signal processing, and automatic control.

### 1.2 Objective

The objective of this research is to digitise a conventional mechanical water meter through image processing. Optical character recognition (OCR) on printed character seems to a perfect candidate for the required technology, but it requires costly system resources such as memory mostly based on computers and other large systems, not feasible for low-cost 8-bit microcontroller implementation. The lack of portability impedes actual use. This system would be in such a design under the constraint of the corresponding hardware to realize the required OCR algorithm to run on a single chip microcomputer during the entire identification process. The main content of this research includes the system design, analysis, and implementation of the image capture module, the processing module, and display module.

### 1.3 Research Questions

In this research, by using micro-controller technology, electronic technology, automation technology, communication technology, pattern recognition theory, and digital image processing technology, we designed an automated meter reading system. The following two research questions are considered in this thesis.

- Question 1: Can we design a light-weight OCR algorithm for the digitisation of conventional mechanical water meter?
- Question 2: Can we implement the designed algorithm by using a low-cost microcontroller-based system while maintaining the required level of character recognition accuracy?

## **1.4 Research Contribution**

This thesis makes the following contributions to the development of smart cities using AMR technology.

1. We proposed a OCR algorithm, which combines the advantage of template matching algorithm and geometric moments algorithm.
2. We design a microcontroller-based OCR system to implement the proposed OCR algorithm.
3. We perform both simulation and experimental studies to compare the performance of the proposed OCR algorithm with that of existing algorithms.

## **1.5 Thesis Organisation**

The thesis is organised in six chapters.

Chapter 1 explains the background of the Internet of things, the automated meter reading system and the objective of this research.

Chapter 2 gives a comprehensive review of the origin of optical character recognition and introduces the image processing steps. It also summarises the features of the hardware used in this research.

Chapter 3 describes the image processing method used in this research, especially the recognition algorithm. By analysis of the advantages and disadvantages of existing recognition algorithms, a new recognition algorithm is obtained.

Chapter 4 presents the components and communication theory used for hardware design, image processing theory in program implementation and the idea for printed circuit board design.

## CHAPTER 1. INTRODUCTION

---

Chapter 5 discusses the results of the image processing method, analyses the results.

Finally, Chapter 6 concludes the whole research, and provides recommendations for future research on this topic.

# Chapter 2

## Literature Review

### 2.1 Optical Character Recognition

Optical character recognition (OCR) is usually adopted in electronic devices (such as digital cameras or scanners) that check printed or hand writing characters by testing the pattern of dark and bright to determine its shape, using a character recognition algorithm to translate them into plain text. In other words, OCR refers to the method of image processing which transfers text documents as black and white bitmap image files, and uses the image recognition software to convert them into text format [12].

In this research, the characters to be recognised are not printed on paper; they are the numbers displayed on a water meter. The key objective of this research is to translate the display numbers into plain text through OCR techniques.

In 1870, an American inventor, C. R. Carey, invented the retinal scanner, which is considered to be the first invention of OCR technology. In the early 1990s, OCR technology was applied in industry to help blind people to understand visual information – a machine could recognise words or Morse code and convert them into a special voice. Early OCR technology by converting Morse code into a special voice helped people

in need. In the 1960s, OCR techniques could recognise letters and help post offices process mail. In the beginning of the 21st century, OCR stepped into online service, in the cloud computing environment, as well as in mobile applications like real-time translation of foreign languages [13, 14].

OCR in computer platforms has experienced a long process of development, the existing recognition technology has been well-explored. With the computer processor speed and storage capacity increasing, in order to improve recognition accuracy, the algorithm tends to be complicated, and involves machine learning like deep learning. This research uses a single-chip micro-controller as the processing unit, which cannot adopt the existing character recognition algorithms as they are mostly designed for computer platform with plenty of computational resources. We will propose an algorithm suitable for this embedded digital system [15].

## **2.2 Image Processing Algorithm**

OCR belongs to a part of image processing. Processing the input image file before recognising its contents improves the accuracy of recognition [16, 17]. The main parts of the digital image processing include image transformation, image compression, image enhancement and recovery, image segmentation, representation and description ,and object recognition [18].

### **2.2.1 Image Binarization**

In digital image processing technology, image binarization attends a very important position reducing the amount of image data in order to highlight the target contour. In a colour image, each pixel can be constituted by RGB data or YUV data. R is red, G is green and B is blue. Any colour in nature can be described by mixing these three

colours in different saturations. YUV is another kind of colour expression. Y represents Luminance, U represents Chrominance and V represents Chromam [19]. RGB and YUV values can transform from each other, the formula is

$$Y = 0.59 \times G + 0.31 \times R + 0.11 \times B \quad (2.1)$$

$$U = R - Y \quad (2.2)$$

$$V = B - Y \quad (2.3)$$

Usually, when we get an image with an RGB value, we need to put it into a grey image. For image recognition, we only need an image with the most basic information, be getting rid of redundant data, thus reducing the computational complexity. Each pixel in a grey image only has a luminance ( $Y$ ) value.

The luminance value of each pixel is between 0 and 255. For highlighting the target contour [20], we simplify the luminance value to only 0 or 1 [21].

$$Y_{BIN}(x, y) = \begin{cases} 1, & \text{if } Y(x, y) > t \\ 0, & \text{if } Y(x, y) \leq t \end{cases} \quad (2.4)$$

$Y(x, y)$  is the value of the pixel at the coordinate before binarization, and  $t$  is the binarization threshold,  $Y_{BIN}(x, y)$  is the value of the pixel at the coordinate after binarization.

The pixel with value 1 reflects a white square and value 0 reflects a black square. After binarization, a grey image becomes an image only in black and white.





(a) Luminance Image

(b) Binarization Image

Figure 2.1: Image after Binarization Process

The key point of image binarization is to find out the appropriate threshold value to exhibit the integrated target contour.

### 2.2.2 Noise Removal

Image noise is present in the image data which contains redundant interference information. The existence of noise seriously affects the quality of the image, which must be corrected before image recognition.

There are two kinds of noise common to image processing: salt and pepper noise, and Gaussian noise [22]. The main source of Gaussian noise in digital images appears during the image capture step, caused by poor lighting and/or high-temperature sensor noise [23]. Salt and pepper noise, also known as impulse noise, is a kind of random white or black point in the image. Maybe the bright area gets black pixels, or the dark area gets white pixels (or both) [24]. The cause of the salt and pepper noise may be image signals effected by sudden strong interference, transmission errors, or

analog-to-digital converter error, etc.

Using a noise removal filter can reduce the influence of noise on image processing. The common filters used for removing noise are Mean Filter, Gaussian Filter, and Median Filter [25, 26, 27]. A median filter and Gaussian filter are the linear filters. We select a template for the target pixel, this template includes the approach of the surrounding pixels. Generally, we choose a  $3 \times 3$  template. There are eight surrounding pixels and one target pixel, these nine pixels constitute a filter template [28, 29]. A median filter uses the average value of all the pixels in the template to replace the target pixel. A Gaussian filter is to use a weighted average value of all the pixels in the template to replace the target pixel.

A median filter is based on the theory of statistics of a nonlinear signal processing technology, which can effectively restrain noise. According to the order of all the pixels in the template, we use the median value instead of target pixels, letting the surrounding pixel values close to the real value of the order remove isolated noise points.

Different filters for different types of noise have different performances, so in the actual application we need to select a filter according to the type of noise [30].



(a) Original Image

(b) Grey Image

Figure 2.2: Original and Grey Image

We add Gaussian noise and salt and pepper noise on Figure 2.2(b), and compare the result by using different noise removal filter.



(a) Image With Gaussian Noise



(b) Image with Salt and Pepper Noise



(c) Median Filter for Gaussian Noise



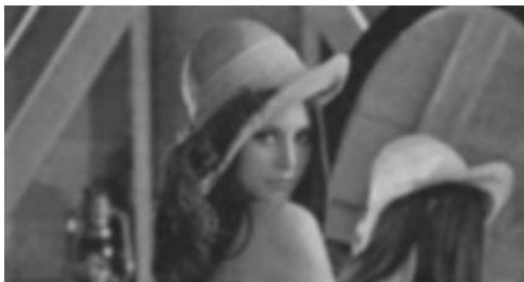
(d) Median Filter for Salt and Pepper Noise



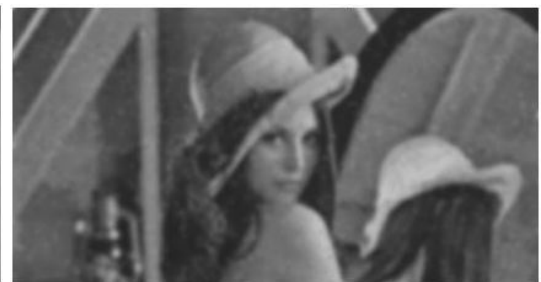
(e) Mean Filter for Gaussian Noise



(f) Mean Filter for Salt and Pepper Noise



(g) Gaussian Filter for Gaussian Noise



(h) Gaussian Filter for Salt and Pepper Noise

Figure 2.3: Noise Removal Filter

From Figure 2.3 we can find out, the median filter can remove Gaussian noise and salt

and pepper noise from image, especially best for salt and pepper noise removal. Mean filter can remove Gaussian noise but not salt and pepper noise. Gaussian filter reduces the image definition.

### 2.2.3 Image Segmentation

Image segmentation is one of the basic problems of image processing and computer vision is used to divide the image into a number of specific areas with a processing target inside. The image segments into some meaningful area corresponding to the various target objects. Through the description of segmentation results, we can understand the information contained in the image. It is the key step in image processing and image analysis [31].

Image segmentation is the process of pixel classification, the basis of classification can be the characteristics of pixel similarity or discontinuity. The existing image segmentation methods fall in the following categories: threshold segmentation method, region segmentation method and edge segmentation method, etc. Threshold segmentation and region segmentation are according to the similarity between the pixels. Edge segmentation belongs to the division of discontinuity.

Image segmentation is an important pre-treatment of image recognition and computer vision. Without the correct segmentation, it is impossible to have the correct identification.

#### 2.2.3.1 Threshold Segmentation

Image binarization is the most basic and has been the most used segmentation method used for the grey image. It transfers the input image  $f$  to the output image  $g$  :

$$g(i, j) = \begin{cases} 1, & \text{if } f(i, j) \geq T \\ 0, & \text{if } f(i, j) < T \end{cases} \quad (2.5)$$

$T$  is the threshold value. For the pixel on the target,  $g(i, j) = 1$ , and the pixel on background,  $g(i, j) = 0$ .

In threshold segmentation algorithm, the key is to determine the threshold value. If we can determine a suitable threshold value, the partition image can be accurately determined. Because of the advantage of simple calculation, high operation efficiency and speed, the threshold segmentation method used in the applications attaches importance to the operation efficiency [32, 33].

### 2.2.3.2 Region Segmentation

Region growth method and splitting and merging method are two typical region segmentation approaches. The follow-up step of processing is according to the segmentation process and determined by the results of the previous step.

Region growth is the most basic region segmentation method [34]. It is a process which assembles pixels or small areas into larger areas. Starting from a set of growing points, we search the nearby area, dividing the image into regions which have similar features and comparing the growth characteristics of similarity of adjacent areas. If they are sufficiently similar, they merge as the same area and become a new growing point. In this way we merge the areas with similar characteristics until completion.

The splitting and merging method is almost the reverse of region growing: starting from the whole image, dividing each area, then combining similar areas, achieving the goal to segment the image [35].

### 2.2.3.3 Edge Segmentation

Edge segmentation is achieved by edge detection, by detecting the mutation of greyscale, which indicates the end of a region and the beginning of another region [36]. This discontinuity is called edge. In general, the area between the target image and the background has obvious edges. Gray levels of pixels in the image edge are discontinuities, which can be detected by calculating the derivatives. Therefore, the differential operator is commonly used for edge detection. The operator is sensitive to noise and only suitable for small or uncomplicated images.

In this research, the process targets are numbers. The idea of image segmentation is to find the rectangular area which has only one number inside [37]. We are going to use the projection method [38]. The projection method is a kind of eigen function. It makes a two-dimensional image pixel distribution characteristic simplify to x axis and y axis to become to two one-dimensional functions.

$$\begin{cases} \text{Vertical Projection} & V(j) = \sum_{i=1}^N f(i, j), j = [1, M] \\ \text{Horizontal Projection} & H(i) = \sum_{j=1}^M f(i, j), i = [1, N] \end{cases} \quad (2.6)$$

Using the vertical projection method to segment the input image, according to the numerical statistical characteristics of projection image the grey value accumulation finds the left and right edges around the number. Using the horizontal projection method, we find the top and bottom edges around the number.

## 2.3 Recognition Algorithm

Image recognition belongs to the category of pattern recognition. After some preprocessing such as image enhancement, restoration, and compression, we identify the target

area. Recognising the contents of the area is the most important step in the optical character recognition process.

### 2.3.1 Template Matching

The first patent obtained about template matching was in 1929 by Germany scientist Tausheck, who combined optical and mechanical template matching. He used 10 pieces of templates to match the numbers from 0 to 9. The light passed through mechanical masks was captured by a photodetector and scanned mechanically. When an exact match occurred, the light failed to reach the detector, then the machine recognised the characters printed on paper [39].

Template matching is a technique in digital image processing for finding small parts of an image which match a template image [40]. Template matching is one of the most primitive, most basic pattern recognition methods. Research of a particular object pattern in parts of the image, and then identifying objects is a method of matching. It is the most basic, the most commonly used method in image recognition [41].

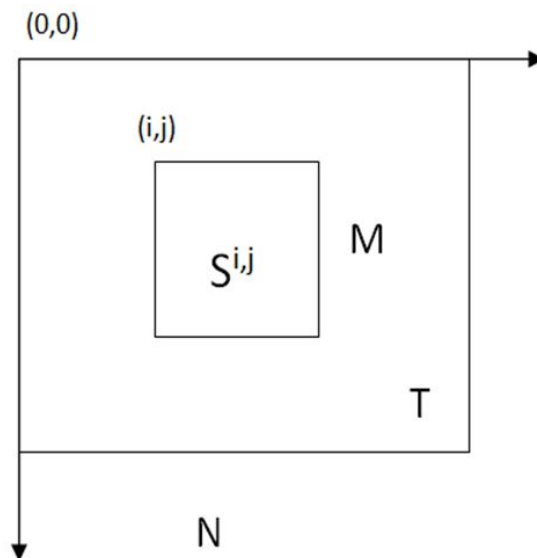


Figure 2.4: Template T On Top Of Input Pattern S

We compute the similarity measurement between template and pattern to find the distance. The idea of the template matching method is using Euclidean distance formula. Euclidean distance is the easiest understood method of distance calculation formula derived from the distance between two points in Euclidean space. In a two-dimensional plane, the Euclidean distance between two points is:

$$D = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \quad (2.7)$$

Using Euclidean distance formula for image processing applications, the distance between two images is:

$$D(i, j) = \sum_{m=1}^M \sum_{n=1}^N [S^{i,j}(m, n) - T(m, n)]^2 \quad (2.8)$$

Expansion Equation (2.8),

$$D(i, j) = \sum_{m=1}^M \sum_{n=1}^N [S^{i,j}(m, n)]^2 - 2 \sum_{m=1}^M \sum_{n=1}^N S^{i,j}(m, n) \times T(m, n) + \sum_{m=1}^M \sum_{n=1}^N [T(m, n)]^2 \quad (2.9)$$

In Equation (2.9), the first part of the equation is the energy of the pattern and the last part of the equation is the energy of the template. Both the energies are stable, not changing by following the pattern shift on the template. Only the middle part of the equation has significance.

Normalising the middle part of Equation (2.9), we get a normalised correlation coefficient:

$$R(i, j) = \frac{\sum_{m=1}^M \sum_{n=1}^N S^{i,j}(m, n) \times T(m, n)}{\sqrt{\sum_{m=1}^M \sum_{n=1}^N [S^{i,j}(m, n)]^2} \sqrt{\sum_{m=1}^M \sum_{n=1}^N [T(m, n)]^2}} \quad (2.10)$$



The normalised correlation coefficient can be interpreted geometrically as the cosine of the angle between the two normalised vectors [42]. A related similarity measure that does not require zero average normalisation is given by the cosine of the angle between the vectors representing the two patterns:

$$r_{\cos} = \frac{x \cdot y}{\|x\| \|y\|} \quad (2.11)$$

The Cauchy–Schwarz inequality states that for all vectors  $u$  and  $v$  of an inner product space it is true that [43]:

$$(u \cdot v) \leq \|x\| \|y\| \quad (2.12)$$

From Equation (2.10), we can get  $0 \leq R(i, j) \leq 1$ . Only the result value  $\frac{S^{i,j}(m,n)}{T(m,n)}$  is a constant value,  $R(i, j)$  gets the maximum value (equals 1). Thus, when pattern  $S$  is completely overlapping with template  $T$ , the coefficient  $R(i, j)$  equals 1. After template  $T$  is completely searched, we can find the maximum coefficient value  $R_{\max}(i_m, j_m)$ , the corresponding  $S_{i_m, j_m}$  is the matching target [40, 44].

The correlation method has great computational complexity, thus slow recognition speed.

In order to overcome these shortcomings, the error method was proposed.

The error method is used to find the error between image  $S$  and template  $T$ :

$$E(i, j) = \sum_{m=1}^M \sum_{n=1}^N |S^{i,j}(m, n) - T(m, n)| \quad (2.13)$$

When the location of the point  $(i, j)$  gets the minimum value of  $E(i, j)$  means the image matching the template. To increase the computation speed, we define an error threshold value  $E_0$  as  $E(i, j) < E_0$ . If the template is matched with an error smaller than  $E_0$ , we stop the calculation.

The template matching method is simple in principle and the error method decreases the amount of calculation. However, template matching has its own limitations: image shift can only be in parallel. If matching the target of the original image rotation or size changes, the algorithm is invalid.

### 2.3.2 Geometric Moments

Geometric moments are kinds of simple moment functions, with the main idea about the product of the pixel coordinates. The benefit of geometric moments is that people can use the moment space to express the image coordinate transformation and analyse the corresponding transformations. Geometric moments also as known as regular moments or Cartesian moments [45].

Moments describe a shape's layout. In other words, moments describe the arrangement of a shape's pixels, its combined area, compactness, irregularity, and higher-order descriptions together [46]. Moments are global description of a shape, and the advantage there is discrimination which is an instinctive ability to detect and filter noise. In image analysis, we do not use mechanical moments but the statistical one, both are similar. For example, the mechanical moment of inertia describes the rate of change in momentum; the statistical second-order moment describes the rate of change in a shape's area. Moments for image analysis was first introduced by M.K Hu in 1962 [47].

A moment of order  $(p + q)$  is dependent on scaling, translation, rotation, and even on grey-level transformations, moment  $m_{pq}$  of a function  $I(x, y)$  is given by

$$m_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q I(x, y) dx dy \quad (2.14)$$

For discrete images, Equation (2.14) is usually approximated by

$$m_{pq} = \sum_x \sum_y x^p y^q I(x, y) \quad (2.15)$$

where  $x, y$  is the region point coordinates, the pixel coordinates are in digitised images. The descriptor has a unique property that it can be shown if the function satisfies certain conditions, then moments of all orders exist.

The zero-order moment  $m_{00}$  is:

$$m_{00} = \sum_x \sum_y I(x, y) \quad (2.16)$$

which represents the total mass of a function. In the binary image case, the zero-order moment gets region area.

The two first-order moments,  $m_{01}$  and  $m_{10}$  are given by

$$\begin{aligned} m_{10} &= \sum_x \sum_y x I(x, y) \\ m_{01} &= \sum_x \sum_y y I(x, y) \end{aligned} \quad (2.17)$$

For binary images, these two values are proportional with the shape's centre coordinates.

Then  $x_c, y_c$  are the coordinates of the region's centre of gravity as

$$\begin{aligned} x_c &= \frac{m_{10}}{m_{00}} \\ y_c &= \frac{m_{01}}{m_{00}} \end{aligned} \quad (2.18)$$

For image recognition, we care about some invariance to position, size, and rotation. Now we have an estimate of a shape's centre, the centralised moments are invariant to translation and computed with respect to the intensity centre. ( $x_c, y_c$ ) are defined as

$$\mu_{pq} = \sum_x \sum_y (x - x_c)^p (y - y_c)^q I(x, y) \quad (2.19)$$

From the definition of central moments, we have

$$\begin{cases} \mu_{00} = m_{00} \\ \mu_{10} = \mu_{01} = 0 \end{cases} \quad (2.20)$$

The second-order moments give a measure of the variance of the image intensity distribution about the origin. The central moments  $\mu_{02}$ ,  $\mu_{20}$  give the variance about the mean (centred). The covariance measure is given by  $\mu_{11}$ . The centralised moments are invariant to translation but not to rotation.

For getting both rotation invariant and scale-invariant, seven-moment characteristics are used [46].

$$M1 = \eta_{20} + \eta_{02} \quad (2.21)$$

$$M2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \quad (2.22)$$

$$M3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \quad (2.23)$$

$$M4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \quad (2.24)$$

$$M5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})((\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2) \\ + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})(3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2) \quad (2.25)$$

$$M6 = (\eta_{20} - \eta_{02})((\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2) + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \quad (2.26)$$

$$M7 = (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})((\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2) \\ + (3\eta_{21} - \eta_{30})(\eta_{21} + \eta_{03})(3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2) \quad (2.27)$$

where,  $\eta_{pq}$  is the normalized central moments.  $M1$  and  $M2$  are second-order moments, thus  $p + q = 2$ . The rest of them are third-order moments since  $p + q = 3$ .

The geometric moment has many applications in image processing, image recognition, and computer vision. Due to geometric moment invariant features, a geometric moment can be applied in rotating images or different sizes in image recognition.

### 2.3.3 Deep Learning

Deep learning is used for applications in image processing and image recognition technologies. The translation application software in the mobile phone can in real time translate the words into other languages by capturing the words using a camera.

Deep learning is the current popular topic in the area of computer science, not only in academic circles, but in many practical applications in industry. Deep learning is a new technology in the study of machine learning algorithms, the motivation is to establish and simulate the human brain analysis neural network [48].

Deep learning (also known as depth structure learning, hierarchical learning or the

deep machine learning) is a set of algorithms [49] and is a part of machine learning. By a multilayer process, we gradually transform the initial low-level features into a high-level, using a simple model to complete the classification of the complex learning task. Deep learning can be understood as feature learning or representation learning.

The idea for deep learning is to build up an artificial neural network. In general, the neural network is a kind of machine learning framework, all of the individual units linking together, in the form of weight. The weights are trained through the network, so it can be called a neural network algorithm. An artificial neural network algorithm is propounded from imitating the human brain's way of thinking. The human brain reacts to an external stimulus by getting input signals from the nervous system and accepts external stimuli with neuron nerve endings as converted electrical signals. Thus, by simulating the way the human brain works, we can build up a deep learning neural network. We can adapt the way neurons are layered, one layer can connect with the other layer next to it. In the past, because of the hardware calculation limit, we could not add a lot of layers in a neural network. Nowadays, with the algorithm updating and hardware development, we can use a lot of layers to develop the neural network, thus creating a deep neural network.

Although the classification of the deep learning architecture seems complex, typical examples are recurrent networks, deep feed-forward networks, and convolutional networks.

Deep feed-forward networks, also known as Multilayer Perceptions(MLP), are the essence of the deep learning model. Feed-forward networks work like a function. For a classifier,  $y = f(x)$ , it will be an input value  $x$  into the corresponding output  $y$ . The feed-forward network is to define a mapping  $y = f(x; \theta)$ , and study the parameter  $\theta$  produces the best function approximation. A neural network has an input layer, many hidden layers, and an output layer. The input layer accepts data, the hidden layers

process data, and the output layer gives the result. This model does not have any feedback connection; therefore, it is called the feed-forward network.

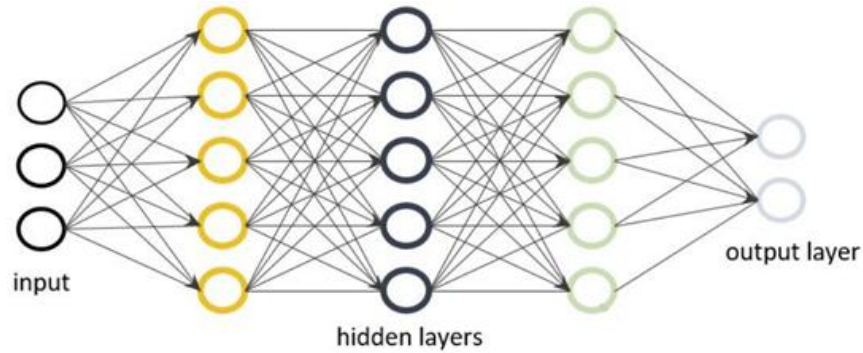


Figure 2.5: Feed-forward Network Architecture

Convolutional Neural Network (CNN) is a kind of feed-forward network in machine learning [50]. The way layer connection afflatus comes from the animal visual cortex. A single cortical neuron can react to the stimulation of a limited space area. This limited space can be referred to as the receptive field. Different neuron's receptive fields can overlap, building up the visible areas.

One neuron reacting to accept stimulus in the domain, can use the convolutional operation to approximate in math. The convolutional neural networks are inspired by biology, designed with a minimum of pretreatment of multilayer perception variants.

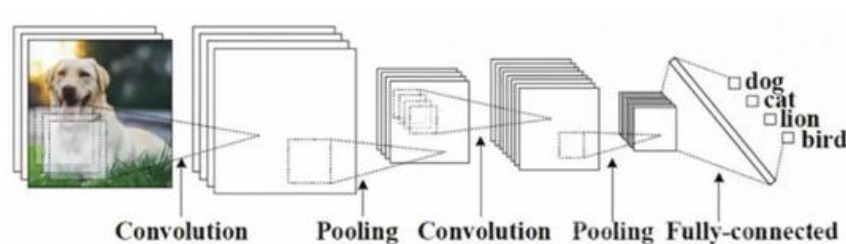


Figure 2.6: Convolutional Neural Network Architecture [2]

In the traditional neural network, we assume that all inputs are independent of each other. But for many tasks, this is not true. If someone wants to know what the next

word is in a sentence, they need to know what is before that word. Recurrent Neural Networks (RNN) call for all of the elements in a sequence performing the same task and all of the outputs depend on the previous calculation. Another way to understand RNN is, that it will remember all the information calculated before [51].

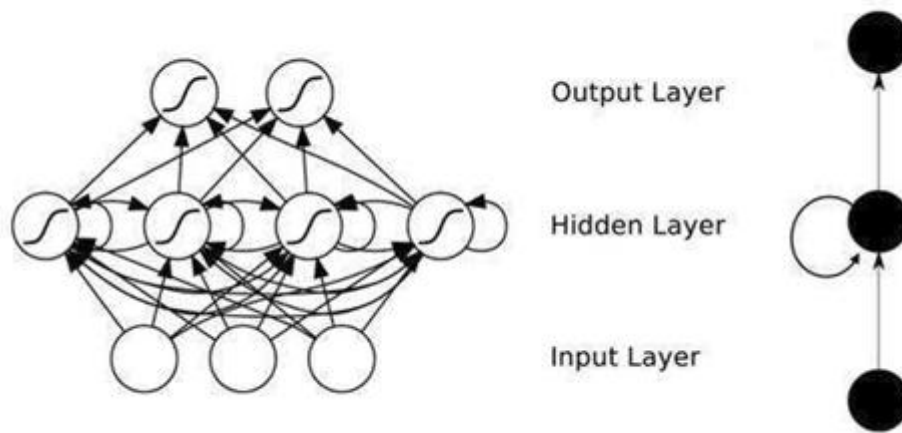


Figure 2.7: Recurrent Neural Networks Architecture [3]

In the field of image recognition, deep learning has many mature applications. However, due to the limited computational resources of a micro-controller, deep learning is not suitable for this research.

## 2.4 Hardware

### 2.4.1 Water Meter

Most New Zealand homes are using the Elster Kent V100 PSM Volumetric Water Meter. It is the world's favourite domestic water meter which sold over 50 million in over 100 countries totally.

Its diameter is 15 mm, the overload flow rate can be up to  $20 \text{ kL/h}$ . Its suitable water



## CHAPTER 2. LITERATURE REVIEW

---

temperature is up to 50 degrees and maximum working pressure up to 16 bar. It is manufactured from the highest quality materials suitable for potable water to follow the Australian and New Zealand standards [4]. It maintains its accuracy in any position for any flow. By using volumetric rotary piston measurement technology, the meter can keep the highest reading accuracy even at a low flow rate in any installed position. The meter has a separate all-in-one counter gear unit above the measuring gear. It is fully sealed and can be easy to read.



Figure 2.8: V100 PSM Volumetric Water Meter [4]

There are 8 number rollers, the left four rollers are black numbers on a white background. The showing number units are in kilo-litres. The right four rollers are white numbers on a red background. The showing number units are in litres. When reading numbers, we only need to consider the black numbers on white background, which is the number showing on the water bills [52].

The meter is reliability guaranteed. The inside measuring parts are made of a special material which is resistant to magnetic interference. It has a non-return valve which provides against outside interference, preventing the meter being operated in the reverse direction to reduce the reading. Each meter is tested for accuracy before dispatch. Another model of the water meter has a type of built-in magnetic device which can send the water flow signal to an extra external induction detector. But this is not the meter installed in New Zealand homes.

As the Kent V100 PSM is the most typical mechanical water meter, it uses a machinery gear to measure water flow and uses a counter gear to display the reading number. Compared with the traditional mechanical water meter, more electronic water meters are being deployed in the rest of the world. The electronic water meter adopts a novel sensor technology, weak signal processing technology, computer and communication technology. It has a wide measurement range, higher measurement level, low-pressure loss, convenient data operation, and transfer function. Its performance is superior to a mechanical water meter.

The main technologies about the electronic water meter include flow measuring technology, signal processing, and anti-interference technology.

The flow measuring technology is by using a flow sensor to detect water fluid in the measuring tube cross-section of average flow velocity to obtain volume flow or mass flow. There are four common detection technologies: electromagnetic flow sensor technology, ultrasonic flow sensor technology, vortex flow sensor technology, and jet

flow sensing technology [53].

Electromagnetic flow sensing technology is commonly used in volume flow measurement. It has a simple structure, no mobilizable and choking components inside the measuring tube, and small fluid pressure loss. It is not affected by measured fluid temperature, viscosity, density or water quality. The output signal is only directly proportional to the average flow velocity and has nothing to do with the flow state. In general, it has a wide measurement range, the measuring accuracy is high, no mechanical inertia and good dynamic characteristics.

Ultrasonic flow sensor technology is also used for volume flow measurement. The induction part is composed of an ultrasonic transducer and a previous signal processing circuit. The ultrasonic transducer converts electrical energy into ultrasonic and emits it through the measurement fluid. The receiver to receives the signal and it is processed by an electronic circuit as an electronic signal on behalf of the flow volume. It can finish the measurement without interference to the fluid's own movement velocity, sports part and choke components, and pressure loss and wear. It does not have any special measuring requirement and measurement accuracy is not affected by temperature, pressure, density, viscosity, and other parameters. It has a wide measuring range.

Vortex flow sensor technology is one of the main forms of fluid vibration sensing technology. It has no moving parts inside the measuring tube, reliable operation, and long service life. It is measured in a certain range of Reynolds number, the fluid vibration frequency is only proportional to the volume flow of fluid under the working state, but not sensitive to the fluid physical properties. The output signal is a frequency value which is easily processed. The scale range is relatively wide.

Jet flow sensing technology, also known as fluidic flow sensing technology, is a new type of flow sensor technology with the development of fluidic technology which, using the Coanda effect feedback, has been widely used for domestic gas and water meters. It is

mainly used for low Reynolds number fluid flow measurement, has obvious advantages in small flow detection and has a wide measurement range and even overload testability. Because the output signal is proportional to the fluid velocity of the fluidic oscillation frequency, it is convenient to get a flow signal with high detection sensitivity. The application by using a new type of electromagnetic detection principle of the jet flow sensor can remove air bubbles [54] and sediment in the liquid which could affect the results of the measurement. This sensor system with no inside moving parts, firm internal structures is not affected by vibration and impact and is easily integrated into manufacturing.

Along with the technology development, no matter the traditional conventional water meter or the new type electronic water meter, the next step is using an automatic meter reading (AMR) system instead of the human meter reader. An electronic water meter can send an electronic signal to the data centre. However, it is more complex to link a conventional meter to an AMR system. The reading number showing on the meter needs to be translated to digital data and the data is then sent to the data centre.

Adding an electronic module to a conventional water meter can enable the required functions for AMR. The electronic module completes the signal collection, data process, data buffering, and transmission to the data centre via a transmission line. According to the proposed method, the mechanical signal is translated to the electronic signal. A remote water meter can be divided into a pulse-type remote water meter and direct reading remote water meter.

When the pulse-type remote water meter works, the receiver counts pulse numbers and gets the fluid flow value. The direct reading remote water meter reads the number display on the meter. Compared with the pulse type remote water meter, the direct reading remote water meter has the following advantages [55]:

- The number showing on the meter is the value directly sent to the receiver.

- The number record on the number rollers does not require a power supply in off periods. Power is only needed when reading numbers.
- The direct reading remote water meter does not worry about power outage and cable break. With the pulse-type remote water meter the receiver collects the signal and counts the pulse number. However, in this process, if the cable breaks or there is a backup power supply outage, the counting process will be affected and cause a mistake. When reconnecting, the whole system needs a reset.

### 2.4.2 Camera

In the late 1960s, a charge coupled device (CCD) and a Complementary Metal Oxide Semiconductor (CMOS) were developing at the same time. Because the inchoate CMOS sensor got a lower image quality than the CCD sensor, in the next 30 years the CCD sensor was mainly used in commercial and scientific research. After the 1990s, with the development of the Very Large Scale Integrated Circuits (VLSI), the CMOS sensor became popular in the low and medium image resolution area.

In this research, we chose to use OV7620 as the image sensor chip. OV7620 is a very powerful camera chip produced by Omnivision. OV7620 is a highly integrated high resolution interlaced or progressive scan CMOS digital colour or black and white video camera chip. The built-in SCCB interface provides an easy way of controlling the built-in camera functions [5].

Main feature of OV7620 include:

- Single-chip digital Colour/Black&white video camera chip
- Scanning-Interlaced/Progressive Scan
- 10-bit two-channel internal A/D converter and 8/10-bit output

- PCLK and HREF polarity programmable
- Digital output format:
  - YCrCb 16-bit/8-bit selectable
  - RGB Raw data digital output 16bit/8bit selectable
- CCIR601/CCIR656 standard
- YCrCb or YUV output format to support a TV or monitor display
- SCCB interface, support fast mode in 400 kBit/s
- External Field Sync input
- Support external micro-controller and RAM interface
- Auto-Exposure/White balance/Brightness and Contrast/Color Saturation/Aperture correction auto/programmable
- Software/Hardware Reset
- Software/Hardware power saving control

Technical parameters:

- Format - 1/3" lens compatible
- Image area -  $4.86mm \times 3.64mm$
- Total active pixel elements -  $664 \times 492$  square pixel
- Default active pixel element -  $640 \times 480$
- Pixel size -  $7.6 \times 7.6\mu m$

## CHAPTER 2. LITERATURE REVIEW

- Programmable frame rate - 0.5 to 30 fps
- PreAmp Gain 6/12 dB
- Signal to Noise Ratio > 48 dB
- Minimum illumination 2.5 lux at f1.4(3000k)
- 5 volt for analog and Digital circuit; 5volts or 3.3volts for digital interface
- Power consumption < 120 mW
- 48pin in LCC package

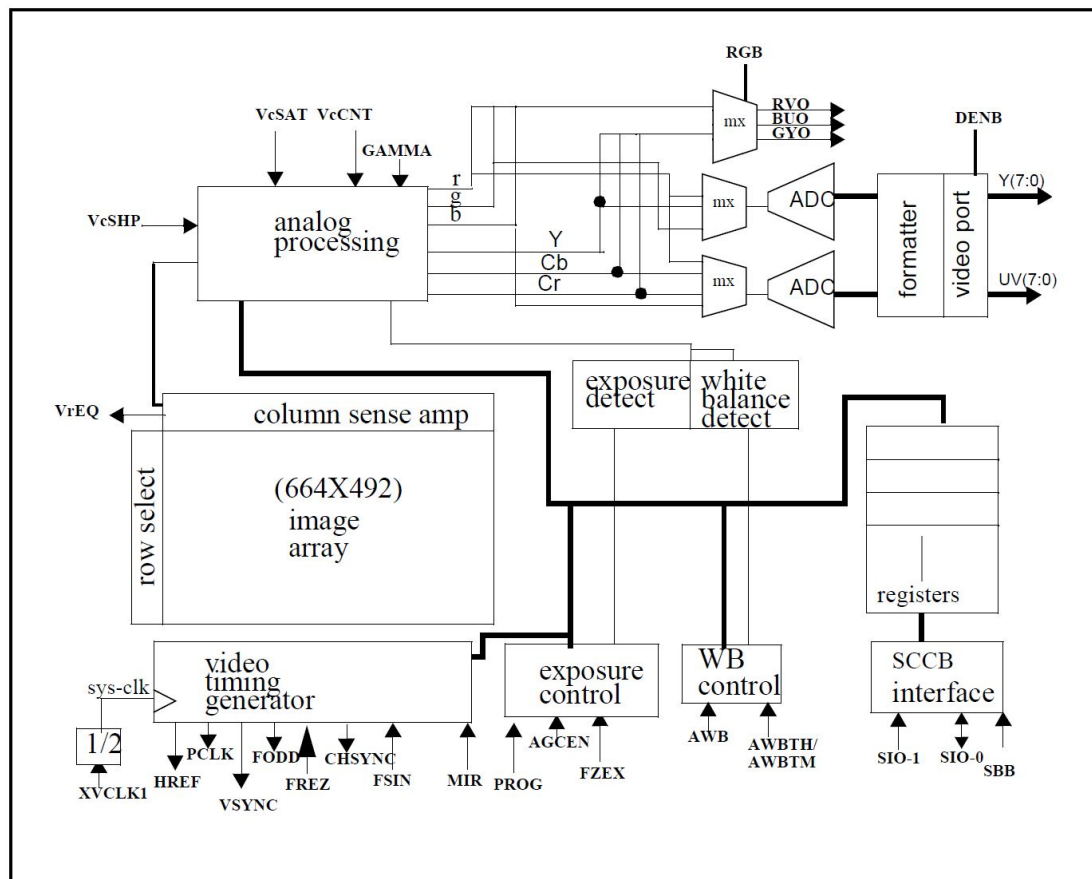


Figure 2.9: OV7620 Block Diagram [5]

As shown in Figure 2.9, OV7620 has a  $664 \times 492$  image array, an analog processing unit, two 10-bit analog to digital converters, analog video mix, digital data formatter and video port, SCCB interface with its registers, digital controls including timing block, exposure block and white balance.

OV7620 is a 1/3-inch CMOS imaging device. The sensor contains approximately 326688 pixels. It is a base on-field integration read-out system with line-by-line transfer and an electronic shutter with a synchronous pixel readout scheme. The colour filter of the sensor consists of a primary colour RG/GB array arranged in line alternating fashion.

The analog processing unit implements most signal processing work, such as colour separation, matrixing, AGC, gamma correction, colour correction, colour balance, black level calibration, knee smoothing, aperture correction, controls for luminance and chrominance picture and anti-alias filtering. The analog video output signal is YUV, the formula is the same as Equation (2.1) to (2.3), where R, G, B are the equivalent colour components in each pixel.

Another output data format is YCrCb, its formula is as follows:

$$Y = 0.59 \times G + 0.31 \times R + 0.11 \times B \quad (2.28)$$

$$Cr = 0.713 \times (R - Y) \quad (2.29)$$

$$Cb = 0.564 \times (B - Y) \quad (2.30)$$

The YCrCb and RGB raw data go to two 10-bit analog to digital converters; one is for the CrCb/BG channels and the other one is for the Y/RG channel. The data stream comes from the analog to digital converter processed by the digital formatter, the digital video port exporting the 16-bit or 8-bit data mixing.



The user can manipulate the window according to their need by using a windowing feature. The window is adjustable from  $4 \times 2$  to  $664 \times 492$  and placed anywhere inside the  $664 \times 492$  range. This function noise does not affect the frame rate or data rate. The user simply sets the parameters of the HREF matching the horizontal and vertical region. The default output window is  $640 \times 480$ .

OV7620 supports grey level image data output. In this mode, all data will be output from the Y port and the UV port will be tri-state. The data output rate is the same as the 16-Bit mode.

The highly integrated OV7620 image sensor has low-cost features and a simple process with image data acquisition and operation. The other advantage to using OV7620 is once OV7620 starts to collect data, it will keep outputting data following the system rate without giving any command to its analog processing unit.

### 2.4.3 Microcontroller Unit

In this research, we chose to use the ATmega1284P as the processor unit. The ATmega1284P is a high performance but low-power CMOS 8-bit microcontroller based on AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the device achieves throughputs approaching 1 MIPS per MHz, balancing power consumption and processing speed [6].

Main features of ATmega1284P include:

- has advanced RISC architecture, with 131 powerful instructions and  $32 \times 8$  general purpose working registers, up to 20MIPS throughput at 20MHz.
- has a nonvolatile program and data memories: 128K Bytes of in-system self-programmable flash which can endure up to 10,000 times write/erase cycles; 4K Byte EEPROM which can endure up to 100,000 times write/erase cycles; 16K Bytes internal SRAM.

## CHAPTER 2. LITERATURE REVIEW

- has a JTAG interface which supports on-chip debug. It can use a JTAG interface to program Flash, EEPROM, Fuses, and Lock Bits.
- has two 8-bit and two 16-bit Timer/Counters, 8 channel 10-bit ADC, two programmable serial USART, master/slave SPI serial interface.
- can work in 4.5 volts to 5.5 volts with 20MHz top speed.

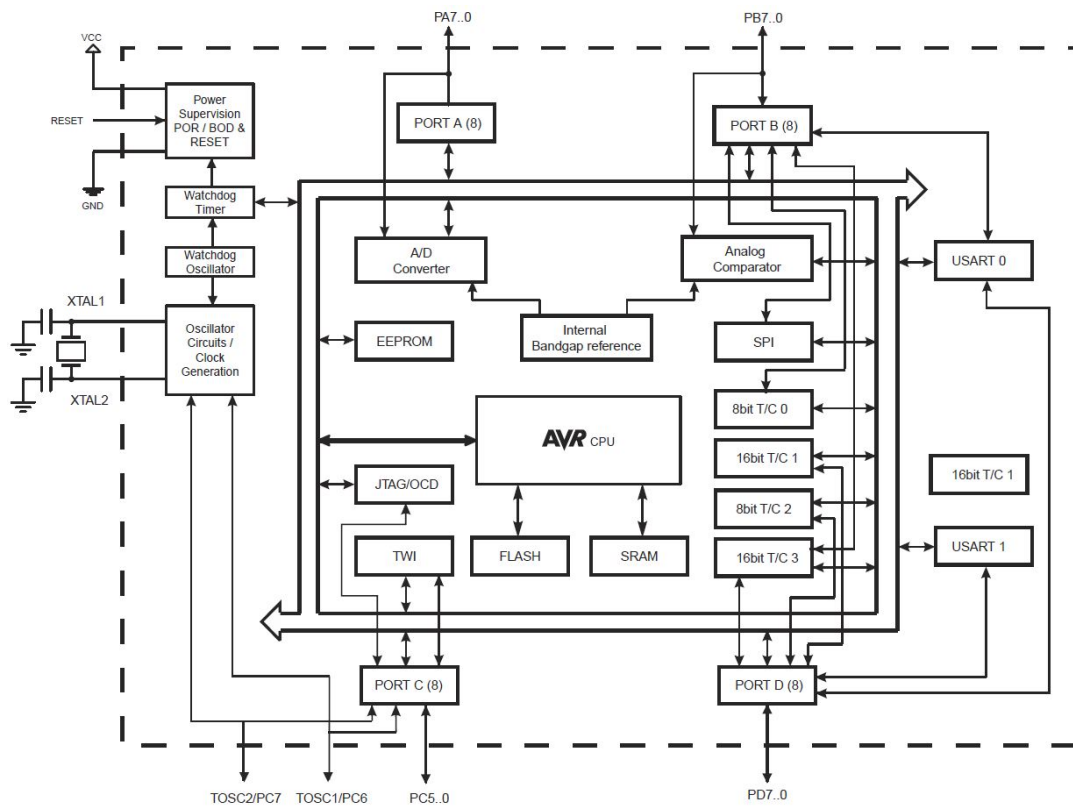


Figure 2.10: ATmega1284P Block Diagram [6]

The AVR core has 32 general working registers inside the instruction set. All of these registers are connect with the Arithmetic Logic Unit (ALU) straightly. In one clock cycle, two independent registers can access in a single instruction. This kind of architecture makes core working more efficiency and up to ten times faster than conventional CISC microcontrollers. The ATmega1284P been supported by a mature

development system include program debugger/simulators, C compilers, evaluation kits, and macro assemblers.

### **2.5 Chapter Summary**

In this chapter, we reviewed the knowledge involved in this research. It includes the techniques about image processing and the main electronic components. Although most of time computers are used to solve image processing problems, an embedded digital system can achieve the optical character recognition by using appropriate method.

## **Chapter 3**

### **Proposed Design**

We propose to use an add-on electronic module to read the numbers displayed on a water meter through optical character recognition (OCR) technology. The system design consists of two stages, namely algorithm design and hardware design.

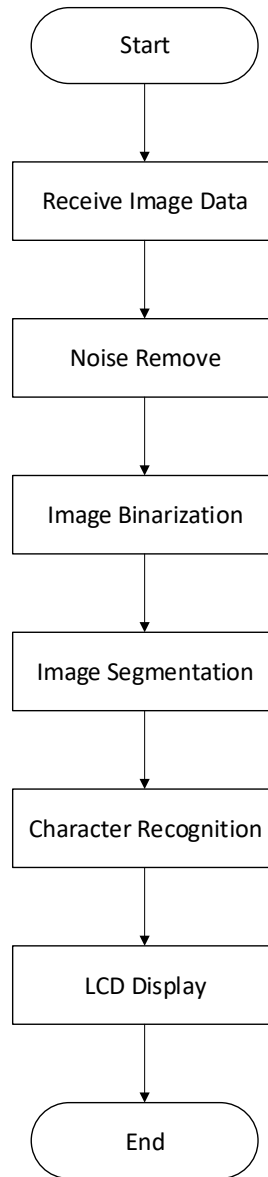


Figure 3.1: Water Meter Digitisation Flowchart

### 3.1 Pre-processing Method

By the camera setting, the output image data only contains the intensity information, as we do not have to process the image from colour to grey. The pre-processing step

includes noise cancelling, image binarization, and image segmentation. After the pre-processing step, we should extract an area which only has a recognition target inside.

### 3.1.1 Noise Remove Method

The image data received from the camera is corrupted with noise. The first step after image data collection is noise removal.



Figure 3.2: Input Image Received From The Camera

The noise appears as random black or white pixels, which conform to the description of salt and pepper noise. We use a median filter to remove the noise.

$$g(x, y) = \text{median} \{f(s, t)\} \quad (s, t) \in S_{x,y} \quad (3.1)$$

$S_{x,y}$  is the median filter template.

A median filter can adequately remove the salt and pepper noise and keep enough information in the image.



Figure 3.3: Image After The Noise Removal Process

### 3.1.2 Image Binarization Method

The image data received only contains the intensity information, which is the value from 0 to 255. We will binarize the data to simplify it to only 0 and 1. Simplifying data can reduce the amount of calculation of image recognition.

$$Y_{BIN}(x, y) = \begin{cases} 1, & \text{if } Y(x, y) > t \\ 0, & \text{if } Y(x, y) \leq t \end{cases} \quad (3.2)$$

where  $Y_{BIN}$  is the binarized image.



Figure 3.4: Image After Binarization Method

### 3.1.3 Image Segmentation Method

After the image binarization step, we get an image containing four numbers. In the recognition step, the four numbers will be processed separately. This step segments each number to prepare for recognition by using the projection method.

$$\begin{cases} \text{Vertical Projection} & V(j) = \sum_{i=1}^N f(i, j), \quad j = [1, M] \\ \text{Horizontal Projection} & H(i) = \sum_{j=1}^M f(i, j), \quad i = [1, N] \end{cases} \quad (3.3)$$

We know each occupied area of a number is  $19 \times 26$  and the approximate location, by using the projection method.



Figure 3.5: Image After Segmentation

## 3.2 Recognition Method

With the binarized image, our next task is to find a suitable algorithm running on the hardware. It is important to consider the limitation of the hardware, optimise the existing algorithm, and evolve the new algorithm.

### 3.2.1 Template Matching Algorithm

Template matching algorithm is the most basic recognition method used for image processing. The procedure for the template matching algorithm is: First, we need to find templates containing the number from 0 to 9. Then, we can use the data we get after the segmentation step to compare with each template to get the error value. The minimum error value indicates the number of readings.

$$E(i, j) = \sum_{m=1}^M \sum_{n=1}^N |S_{ij}(m, n) - T(m, n)| \quad (3.4)$$

As the first attempt, we use the template matching method to identify the data.

For the first step, we intercept from the collected images from 0 to 9; a total of ten numbers as a template. The resolution for each template is 26 (height)  $\times$  19 (width).

Next, we use a template matching formula to calculate the error value between the input image with 10 templates. Now we get ten error values. Finally, we find the minimum



error value, whose corresponding template is the input number.

The advantage of using this method is the high accuracy of identification. The matching is based on a pixel-by-pixel matching. Any pixel different will be reflected in the error value.

Nevertheless, there are many drawbacks. Storage of these templates takes a lot of memory space. One template takes 494 bytes ( $26 \times 19$  resolution), thus 4,940 bytes are required for the ten templates. ATmega1284P SRAM space is of 16K bytes. With the receive image data, the total required memory will exceed the maximum ATmega1284P SRAM space. The other problem is that the template matching will take a lot of computation steps. The computational complexity of one template matching is:

$$T = 3MNn - 1 \quad (3.5)$$

where  $n$  is the number of templates as 10,  $M$  and  $N$  are the sizes of template, 26 and 19, respectively. The total amount of steps to recognise one number is 14,819.

### 3.2.2 Projection Template Matching Algorithm

In order to improve the method of template matching, one goal is to reduce memory space occupied by the templates, by lowering a two-dimensional template into one-dimensional data.

$$\left\{ \begin{array}{ll} \text{Vertical Projection} & V(j) = \sum_{i=1}^N f(i, j), \quad j = [1, M] \\ \text{Horizontal Projection} & H(i) = \sum_{j=1}^M f(i, j), \quad i = [1, N] \end{array} \right. \quad (3.6)$$

We use the projection method to reduce the memory requirement. Before the projection method, one template takes 494 bytes. After the projection method, one template takes 19 bytes (by vertical project template) plus 26 bytes (by horizontal project template)

45 bytes. Ten templates require 450 bytes of memory, even smaller than one two-dimensional template.

Identically, the input image after segmentation needs to be processed by the projection method as well. Now, a 2D template matching becomes a 1D template matching. To find the error value between input image and template,

$$Error = \sum_{m=1}^M |S(m) - T(m)| \quad (3.7)$$

Thus, the calculated amount for one matching reduces to 1,820 times.

To compare with the original template matching method, the projection template matching algorithm reduces the SRAM requirement from 4,940 bytes to 610 bytes. It also decreases the computational complexity from 23,930 to 1,820.

However, using this method, the accuracy drops as well. When compressing a 2D image to become 1D data, lots of details will be lost. After the projection process, one data only can show the total quantity of the pixels in that line but lose the pixel location information. When a number has been mirror-reversed, its projection process data is still the same. For example, for number 6, the number 9 is the mirror reverse, and their projection process data are the same. Using the projection template matching algorithm cannot differentiate number 6 from number 9.

### 3.2.3 Geometric Moments Algorithm

Each number can be seen as a geometry, it has a certain geometric moment, a geometric moment can be used for image recognition.

The zero-order geometric moment can be concluded that the total quality of the image.

$$m_{pq} = \sum_x \sum_y x^p y^q I(x, y) \quad (3.8)$$

The first-order geometric moment of image calculates the centroid coordinates.

$$\begin{aligned} m_{10} &= \sum_x \sum_y xI(x, y) \\ m_{01} &= \sum_x \sum_y yI(x, y) \end{aligned} \tag{3.9}$$

$$\begin{aligned} x_c &= \frac{m_{10}}{m_{00}} \\ y_c &= \frac{m_{01}}{m_{00}} \end{aligned} \tag{3.10}$$

The second-order invariant moment has the translation, scale, and rotation invariant features.

$$\mu_{pq} = \sum_x \sum_y (x - x_c)^p (y - y_c)^q I(x, y) \tag{3.11}$$

$$M1 = \eta_{20} + \eta_{02} \tag{3.12}$$

$$M2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \tag{3.13}$$

Geometric moments are often used in high resolution image recognition. It will not be affected even if the image is rotated or zoomed in and out. However, in this research, the geometrical characteristics of every number of numerical difference are very small. Not to mention the numbers 6 and 9 are font identical, the geometric moment values are the same. Therefore, it is not favourable to use the geometric moments method.

### 3.2.4 Proposed Moment Template Matching Algorithm

By analysing Equation (3.8), we can notice that the principle of geometric moment algorithm actually records each pixel position in the whole image.

$$m_{11} = \sum_x \sum_y x^1 y^1 I(x, y) \quad (3.14)$$

In Equation (3.8), we can set  $p = 1$ ,  $q = 1$  to get Equation (3.14). This is the second-order geometric moment  $p + q = 2$ . This formula can be understood as the quality of each pixel multiplied with its coordinate value, then we sum the total value. When a pixel is multiplied with its coordinate, the result includes its quality and location information. However, at the summation step, all the details are lost.

We propose to combine the template matching algorithm and geometric moment algorithm such that we can ensure the accuracy close to that of the template matching algorithm but reduce the memory space requirement.

The Projection Template Matching Algorithm in Section 3.2.2, can reduce the memory space requirement but after the projection process, image data loses the location information. Combining the projection with the moment idea, we get a new moment template matching algorithm. We propose to use the moment calculation to compress all 2D image data to become 1D data, then use the template matching to find the similarity between input image with all templates.

$$m(i) = \sum_{j=1}^M j \times I(i, j), \quad i = [1, N] \quad (3.15)$$

Where  $I(i, j)$  is the value of each pixel,  $i$  and  $j$  are its coordinate value.  $M$  is a one-dimensional array, each of the  $m$  values is the sum of the corresponding pixel values in that column with the coordinate values multiplication. This is similar to the projection algorithm, but instead of just adding a line or a column of pixels value together, each new  $m$  value contains both pixel value information and coordinate information.

$$Error = \sum_{i=1}^N |m_S(i) - m_T(i)| \quad (3.16)$$

After the compression step, we find out the error between input data and all templates. In practice, templates get pre-processed by the two-dimensional image compression for the one-dimensional array. Combining Equation (3.15) and Equation (3.16), we finally get the function for the proposed Moment Template Matching Algorithm:

$$E_k = \sum_{i=1}^N \left| \sum_{j=1}^M j \times S(i, j) - T(i) \right| \quad (3.17)$$

$S(i, j)$  is the input pattern obtained from the pre-processing steps.  $T$  is the template array.  $E_k$  is the difference between the input pattern and the  $k_{th}$  template. Ten templates correspond to 10 error values, and template associated with the smallest error value indicates the recognised value of the input pattern.

### 3.3 Chapter Summary

In this chapter, we present the system design steps, which include noise cancelling, image binarization, and image segmentation. In second part of this chapter we proposed Moment Template Matching Algorithm. We start from the existing template matching algorithm, combined with computation of the geometric moments, and obtained a new character recognition algorithm that is suitable for implementation using ATmega1284P microcontroller.

## **Chapter 4**

# **Prototype Implementation**

In this chapter, we describe how to implement the proposed algorithm on the hardware. It is easy to demonstrate the effectiveness of the proposed algorithm on a computer, but we also need to enable the algorithm to work on a microcontroller based platform. Implementation of the prototype includes hardware component selection, printed circuit board design to software coding.

### **4.1 Hardware Design**

In the hardware system, once the micro-controller receives the push button signal, it will send a camera control signal, receive image data, and display the reading number on an Liquid Crystal Display (LCD). We use MATLAB as the recognition algorithm developing tool. Once the micro-controller receives the image data, it will send data to a computer via Universal Synchronous/Asynchronous Receiver/Transmitter (USART). The system block diagram is shown in Figure 4.1.

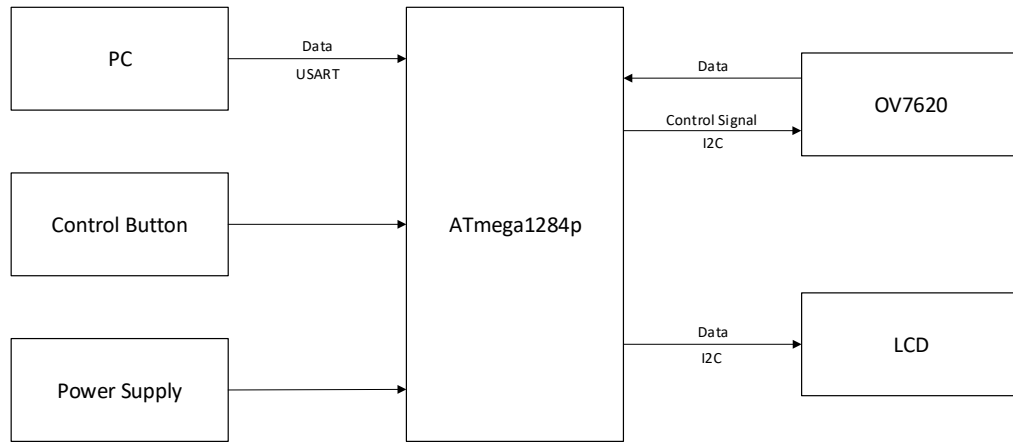


Figure 4.1: Hardware Block-diagram

### 4.1.1 Micro-controller Module

Table 4.1: ATmega128 compared with ATmega1284P [1]

Product	Pins	Flash Size	SRAM Size	Max Speed	Price
ATmega128	64	128Kb	4,096b	16MHz	\$8.25
ATmega1284P	44	128Kb	16,384b	20MHz	\$3.84

Functional requirements of the prototype includes

- Use a micro-controller to control the camera take a photo of the water meter
- Enable the camera to send the photo data back
- Save the photo data inside the micro-controller memory
- Process the photo data and obtain the meter reading number
- Display the number on the LCD display

Considering the requirements above, the microcontroller needs to have the correct peripheral interface, similar speed as the camera, and enough memory to store the image data. The speed of the camera is 27MHz, the speed of micro-controller needs to be close to that to collect the image data. The memory size of the photo is 6.4 kilobytes (KB), which will save into the SRAM flash memory.

ATmega1284P has 16KB internal SRAM to store image data, 20MHz maximum speed to synchronise with a camera, a two-wire serial interface to communicate with camera and LCD display, and a USART interface to communicate with a computer. We finally decided to use ATmega1284P as the micro-controller in this research.

### **4.1.2 Camera Module**

When OV7620 uses a 16-bit output format, Y data bus and UV data bus pixel byte sequence as shown in Table 4.2. From Table 4.2 we can find Y data bus exports the luminance value Y, UV data bus exports the chrominance value U and chroma value V alternately. As we need the grayscale image in this research, Y values from the Y data bus will be collected.



Table 4.2: OV7620 Pixel Data Bus

Data Bus	Pixel Byte Sequence					
Y7	Y7	Y7	Y7	Y7	Y7	Y7
Y6	Y6	Y6	Y6	Y6	Y6	Y6
Y5	Y5	Y5	Y5	Y5	Y5	Y5
Y4	Y4	Y4	Y4	Y4	Y4	Y4
Y3	Y3	Y3	Y3	Y3	Y3	Y3
Y2	Y2	Y2	Y2	Y2	Y2	Y2
Y1	Y1	Y1	Y1	Y1	Y1	Y1
Y0	Y0	Y0	Y0	Y0	Y0	Y0
UV7	U7	V7	U7	V7	U7	V7
UV6	U6	V6	U6	V6	U6	V6
UV5	U5	V5	U5	V5	U5	V5
UV4	U4	V4	U4	V4	U4	V4
UV3	U3	C3	U3	V3	U3	V3
UV2	U2	V2	U2	V2	U2	V2
UV1	U1	V1	U1	V1	U1	V1
UV0	U0	V0	U0	V0	U0	V0
Y FRAME	0	1	2	3	4	5
UV FRAME	0		2		4	

VSYNC is the vertical sync pulse. HREF is the horizontal valid data output window. PCLK is the pixel clock used to clock valid data. The rising edge of PCLK is used to clock the 16-bit data. Y<0:7> is the 8-bit luminance data bus. UV<7:0> is the 8-bit chrominance data bus.

From Figure 4.2 we can see that a new frame starts after the VSYNC falling edge. The system starts and checks the HREF rising edge. The rising edge of HREF is the start of a new horizontal valid data. Then the system starts to check the PCLK rising edge. When PCLK is on the rising edge, the Y data bus is stable. We collect the Y value from the Y data bus. The falling edge of HREF means one horizontal line finishes. The system needs to wait for the next HREF rising edge.

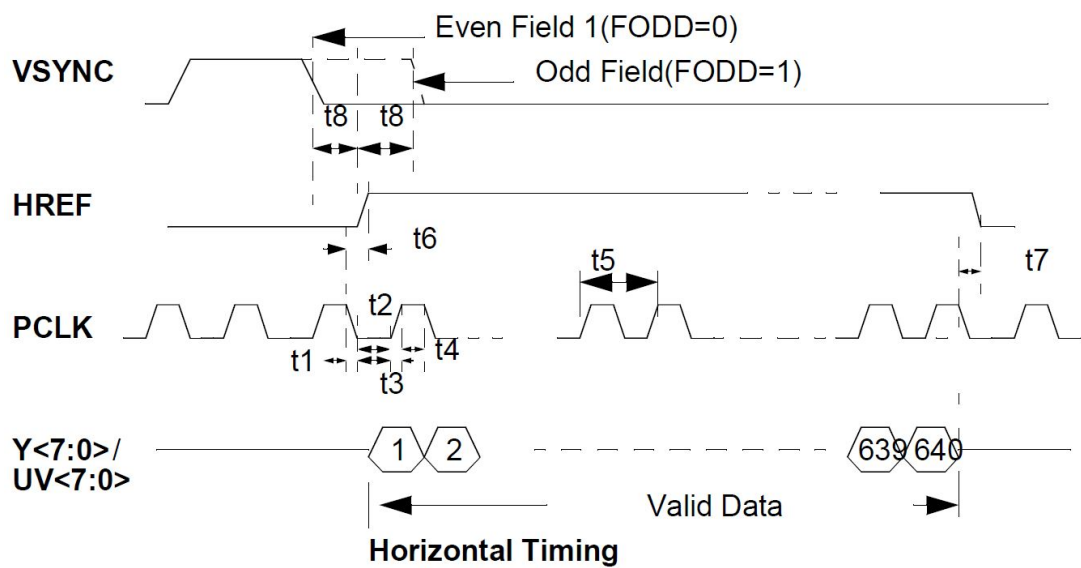


Figure 4.2: Output Signal Timing

By checking the VSYNC, HREF and PCLK signal, we can get a correct image data for one frame.

OV7620 SCCB bus and register setting is explained as follows. The Serial Camera Control Bus (SCCB) is developed by OmniVision company. It is a two-line or three-line synchronous serial bus protocol, supporting 100Kb/s or 400Kb/s transfer speed. Used to set the registers of OV7620, the SCCB control bus function is achieved by the condition of electrical level of line SCCE, SIO\_C and SIO\_D. SCCB is active low when connecting it with the ground, SCCB bus works in two-line mode. At this time, the workings of SIO\_C and SIO\_D are very like a I2C data bus. Similar to a I2C bus,

SIO\_C is serial clock input line and SIO\_D is serial two-way data line. SCCB bus timing is similar to the I2C bus. The ACK signal is the ninth data of the transfer data group, which is Don't Care and NA. The Don't Care is signal generated by slave device, the NA signal is generated by the master device. Because SCCB does not support multi-data reading and writing, the NA signal must be at an electrically high level. OV7620 uses EEPROM as a memory to store the register data. EEPROM is unstable so, every time when we reset the camera, we need to re-initialise the registers.

### 4.1.3 Liquid Crystal Display Module

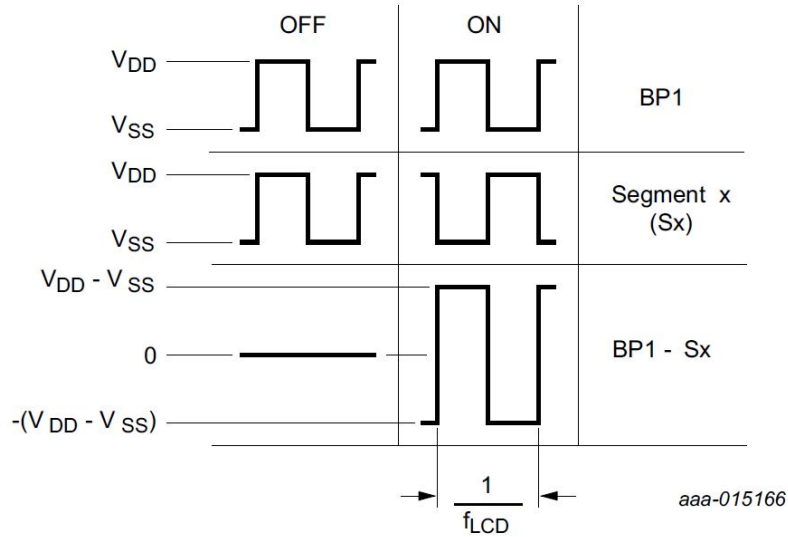
LCD panels are widely used for cell phones, touch panels, and computer monitors in small, medium or large formats. For developing a new product, selecting a suitable LCD as an output module is very important.

In this research, the LCD needs to display only four-digit numbers. There is no integrated  $4 \times 1$  LCD module in the marketplace. Based on the principle of low power consumption and low cost, we designed our own display module by using a digit clock panel LCD-S401C39TR.

LCD is not like a Light Emitting Diode (LED). It cannot be driven by DC voltage. Under the influence of voltage, liquid crystal molecules will change the arrangement of their molecular state, and can make an incoming light deflection phenomenon which is the principle of the liquid crystal display. Adding the electric field onto the LCD pixels makes the pixels work [56], and the electric field is generated by electrode potential at both sides of this display pixel. It is easy to build a direct current field on the pixel, but the direct current field will result in reaction liquid crystal materials and electrode ageing, quickly shortening the LCD display life [57]. We choose to use PCF8577C as the LCD driver IC.

On the PCF8577C setting, we are going to use direct drive mode for driving 32 LCD

segments. Direct drive output wave-forms are shown in Figure 4.3 below.



$$V_{on(RMS)} = V_{DD} - V_{SS}; V_{off(RMS)} = 0.$$

Figure 4.3: Direct Drive Mode Display Output Waveforms

PB1 is connected to the LCD COM pin and Sx is connected with each LCD segment pin. VDD is supply voltage and VSS is ground supply. Both PB1 pin and Segment pin generate waveforms at the same time. When the BP1 waveform and Segment waveform are the same, we turn off the pixel. When BP1 waveform and Segment waveform are contrary, we turn on the pixel.

### 4.1.4 Main Communication Interface

In the world of serial data communication, a protocol such as RS-232, RS-422, RS-485, Serial Peripheral Interface (SPI) and Microwire are used for high and low interfacing speed peripheral connections. These protocols require more pin connections for serial data communication to take place, as the need of the integrated circuit is to reduce the physical size of the chips, we require less numbers of pin connections for serial data transfer.

Inter-integrated circuit (I2C) protocol is a system developed by Philips to provide communications between IC devices during the 1980s to overcome pin connection problems, which requires only two lines for communication. The more advantage beyond the above protocols is that I2C protocol can communicate with two or more chips as a network. It has become a method to communicate with a multitude of peripheral devices.

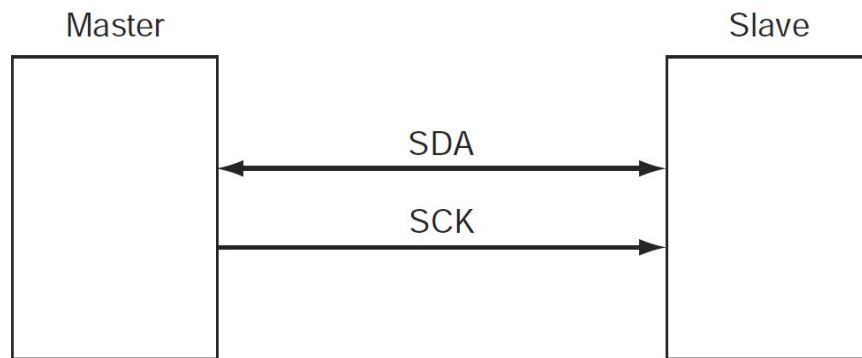


Figure 4.4: I2C Data Flow

The bus operates using two lines: a data line called Synchronous Data (SDA) and a clock line called Synchronous Clock (SCL). Unlike SPI where data can flow simultaneously from master to slave and from slave to master, I2C is sequential as data starts flowing from master to the slave and then from slave to master if needed. Figure 4.4 shows the clock flowing one way (master to slave) and the data flowing both ways on a single wire [58, 59, 60].

AVR series MCU internal integration the two-wire serial interface (TWI) bus, which is the inheritance and development of the I2C bus. It has its own function module and registers, makes the TWI bus on the operation and use simpler than the I2C bus.

The I2C protocol connection is constituted by a master device with several slave devices. The two active wires, SDA and SCL, carry information in the connection. The master device controls the data flow, but data can flow in any direction on the I2C bus. A

typical I2C bus has two types of addressing mode: a 7-bit addressing mode can connect 128 devices and a 10-bit addressing mode can connect 1000 devices. An AVR series MCU only can use a 7-bit addressing mode, which means the maximum connecting devices number is 128. Every device in the protocol has its own address. The master device by called the address number to activate the corresponding slave device.

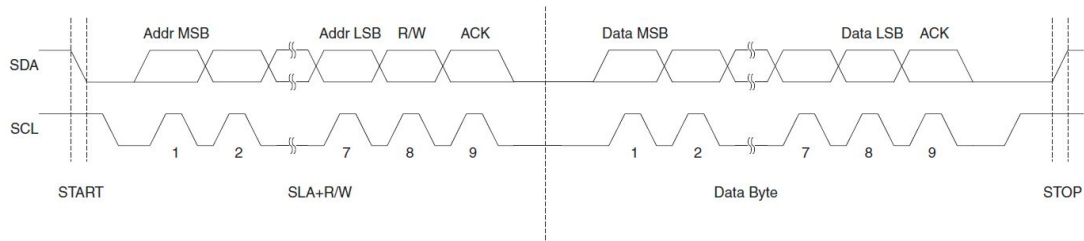


Figure 4.5: Typical Data Transmission Sequence

A set of data transmission starts from the IDLE condition, both SCL and SDA are at logic 1 level. If the bus is IDLE, the master can generate a START signal by pulling the SDA to logic 0 and keeping the SCL in logic 0 levels. After the START signal, the master will send a 7-bit slave address with a read or write bit(R/W) for accessing the particular slave device. If the master device wants to write data to the slave device, the R/W bit is logic 1. If the master device wants to read data from the slave device, the R/W bit is logic 0. After that, an acknowledgement (ACK) bit signal is returned from that slave device for feedback. Then, the master device will start sending the 8-bit data through the SDA start from the most significant bit to the least significant bit. Any byte of data can be transmitted with an ACK signal returned from the slave device. Finally, to terminate the data transmission, the master device will send a STOP signal by pulling the SDA to logic 1 and keeping SCL in logic 1. The typical data transmission sequence is shown on Figure 4.5.

In this research, the LCD driver IC PCF8577C and camera OV7620 are both driven by I2C protocol.

Every time MCU sends 3 bytes of data to control PCF8577C. The I2C-bus protocol of this chip includes a slave address byte, a control byte and a segment data byte. The PCF8577C 7-bit slave address is 0x3A. We only write LCD segment data to the chip to control LCD display output numbers, the R/W bit is in WRITE condition which is logic 1. The full 8-bit address byte is 0x74. The control byte contains display mode information and segment byte register address information. The display mode is used to control the chip in direct drive mode or duplex drive mode. The segment byte register address is used to select which register we are going to write data through. The LCD control can display 4 numbers, which means the control byte is used to select a certain number. On the LCD, every number is displayed by 7 lines, the segment data byte is used to indicate which line will appear. By assembling the lines in the correct location, the number from 0 to 9 will show on the LCD.

The camera OV7620 is controlled by the SCCB bus. The SCCB bus is very similar to a I2C bus. In this research, we can use the I2C protocol sending data to control this chip. SCCB protocol used to control the OV7620 registers. Every time the MCU sends 3 bytes, the first byte is the slave address byte, the second byte is the sub address for selecting the individual on-chip registers, and the third byte is the data associated with this register. The register number is from 0x00 to 0x7C, totally 56 controllable registers. Some registers are reserved for internal use, so when writing data to those registers there will be no function. The registers can control brightness, analog sharpness, white balance, clock rate, window size, pixel shift, etc. We just need to find the specific register to set, the others we do not need to care about.

### **4.1.5 Printed Circuit Board Design**

In this research, hardware is constituted by three Printed Circuit Board (PCB) boards. The camera module is a pre-built complete part obtained from an electronics supplier.

We designed our own MCU module and display module. The two modules are two pieces of an independent PCB, covered in the case mounting the water meter.

### 4.1.5.1 MCU Board

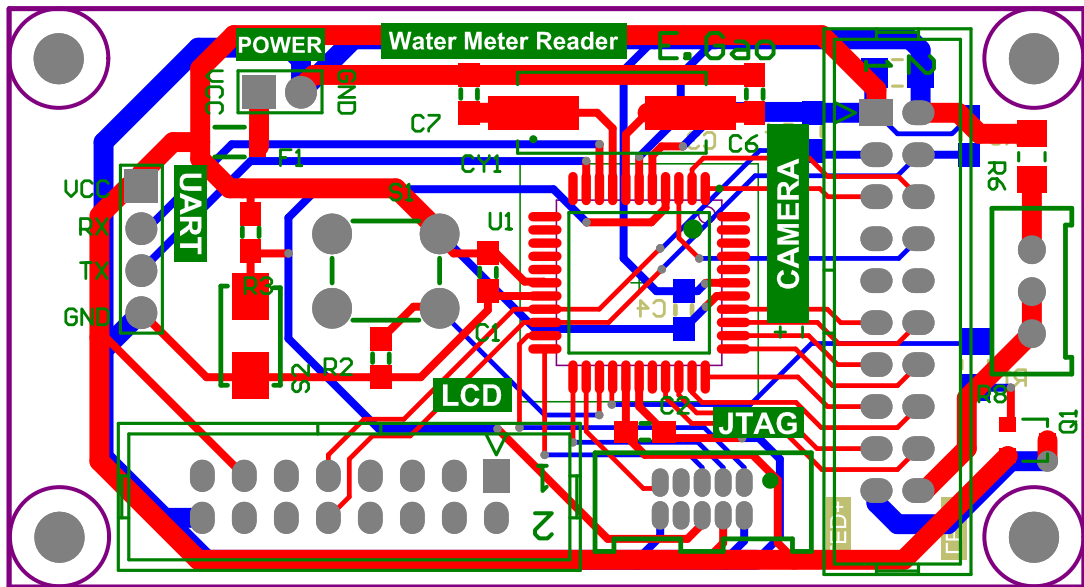


Figure 4.6: MCU PCB

The MCU board is comprised of ATmega1284P and peripheral components (resistors, capacitors, crystal, fuse), pushbuttons, LED control circuit and ports. ATmega1284P is powered by 5V DC voltage and connected to a 20MHz crystal. We chose to use a 0.5A fuse to protect the whole circuit. There are two buttons on the board: one is a reset button that connects with the micro-controller reset pin to reset the chip, and the other one is used to give a signal to the micro-controller. Once the button is pushed, the micro-controller starts to work – reading data from the camera, processing information, and sending number data to the LCD.



There are four ports on the MCU board – USART port, LCD port, JTAG port and camera port. The USART port connects with an FT232 module which used to upload the image data to a computer for simulation. The recognition algorithm is exploited on the computer by using MATLAB. The micro-controller SRAM size is not enough to store a full  $640 \times 480$  size photo. To minimize the photo size, we use the USART port to upload the image to the computer to ensure befitting window size and window position. After the image binarization and segmentation steps, we can check if the result is correct by uploading the data to the computer.

The JTAG port connects with an AVR programmer used to program the micro-controller. The advantage compared with SPI is that, JTAG can use a debug function which is very helpful for fault-finding.

The LCD port connects with the LCD module and the camera port connects with the camera module. Both modules adopt the I2C protocol. In this I2C bus, the micro-controller is the master device, and both LCD and camera are slave devices. On the I2C bus, there are two pull up resistors connected with the Vcc and SDA line and SCL line separately. I2C protocol defines the IDLE condition when both SDA and SCL are logic HIGH. The register connects the Vcc and I2C bus, to pull up the electrical level to logic HIGH.

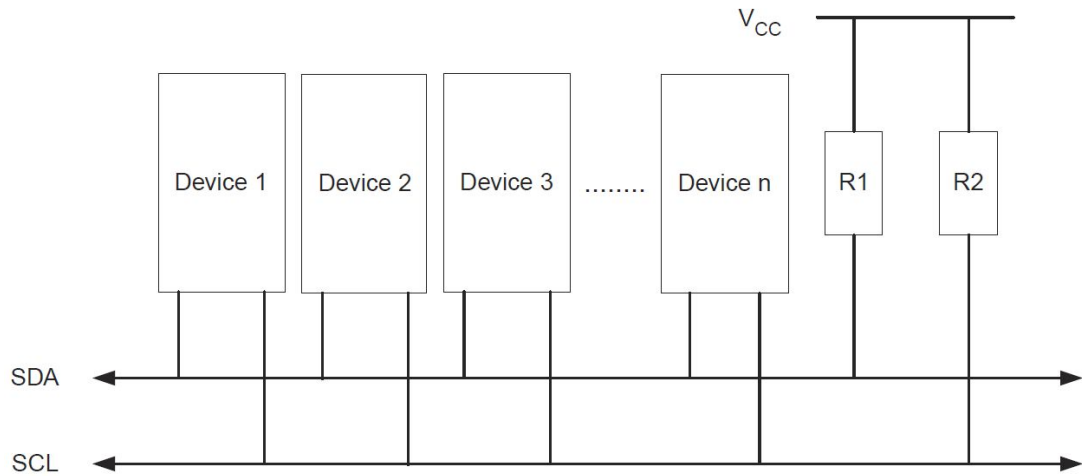


Figure 4.7: I2C Bus Interconnection

There is a LED control circuit on the MCU module board. The domestic water meter is installed inside a black box underground. For the optical vision we chose to use a 2835 surface-mounted device (SMD) LED as the light source by using a PNP transistor control circuit to isolate the LED from the micro-controller for safety reasons using a trimmer potentiometer to adjust the current through the LED to adjust the luminous intensity.

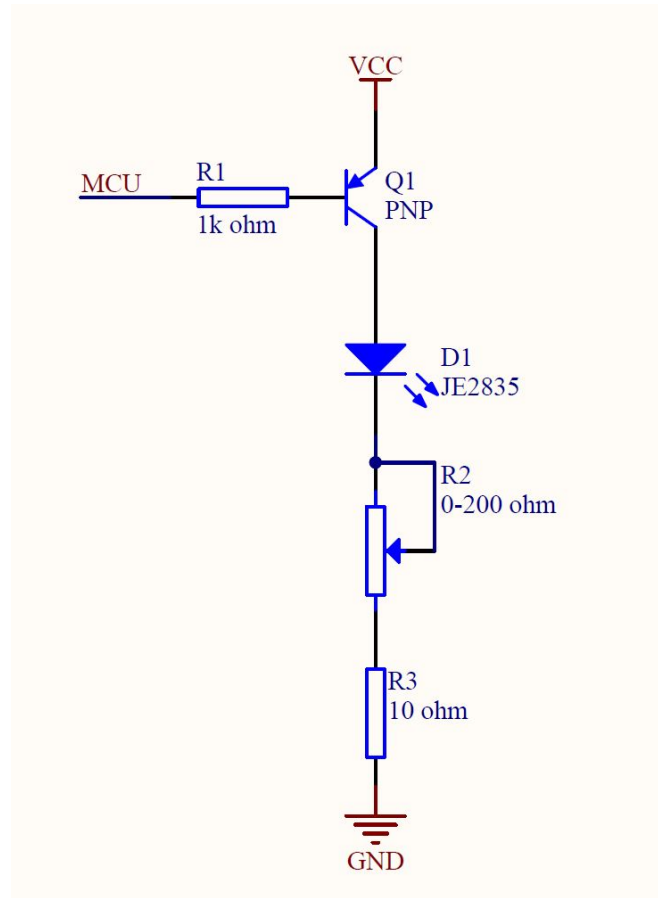
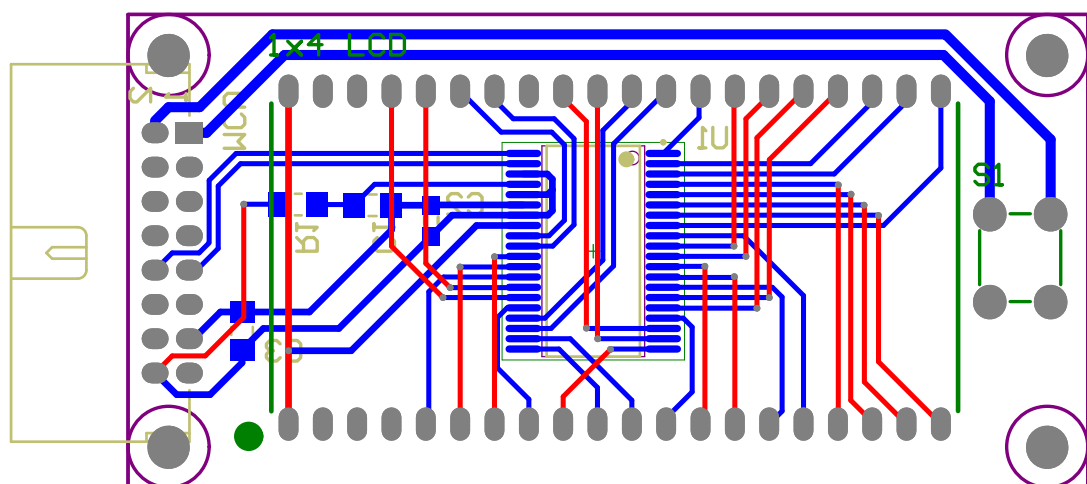


Figure 4.8: LED Driver Circuit

To avoid high current to damage the micro-controller chip, we used a PNP type transistor as a switch to protect it [61]. On the schematic diagram above, Q1 is the PNP type transistor. Its base connects to the microcontroller I/O pin, the emitter connects to the VCC and the collector connects to the LED used to support light to the camera. R2 is a trimmer potentiometer with resistance range from 0 Ohms to 200 Ohms. R3 is a fixed value resistor, the value is 10 Ohms.

When the microcontroller I/O pin is connected to the ground, the current can flow from emitter to base, the transistor gate opens, and the mean current flow is from emitter to collector. R2 and R3 are used to adjust the current flow through the LED. The LED we used is Cree JE2835. Its forward voltage is 3V, working current is 20mA and maximum forward current is 240mA. R3 is used to protect the LED. Once we adjust the trimmer



The LCD-S401C39TR totally has 40 pins, 32 of them are Segment pins, 2 of them are Com pins. PCF8577C has 32 segment outputs, they are directly connected to the LCDS401C39TR Segment pins. The Driver IC periphery circuit uses a  $1M\Omega$  resister and a 680 pF capacitor as an RC oscillator. The oscillator frequency is 90Hz, which is required by the datasheet. The push button on board is linked from the MCU board, which does the same job as the button on MCU board. The water meter installation direction is lying inside the water meter box. At that position, the push button on the LCD board is convenient for people operating the meter reader.

## 4.2 Software Design

In this section, we discuss the software design. The idea is to design the code to program the microcontroller. Because we are using an ATmega1284p as the microcontroller, the compilation environment is Atmel Studio 7 and Atmel ICE. Atmel Studio 7 is the integrated development platform for developing and debugging all AVR and SAM microcontroller applications and the Atmel ICE is the corresponding programmer. Figure 4.10 shows the flowchart of the program.

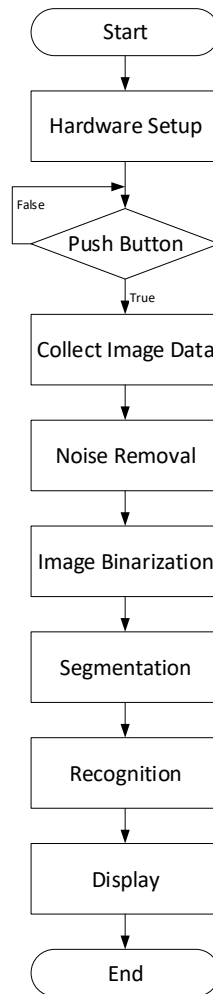


Figure 4.10: Program Flowchart

### 4.2.1 Hardware Initialization

The first part of the main program is to set up the hardware. The setup includes the input and output serial port registers setup, I2C initialize, I2C connection protocol setup, camera setup, and LCD setup.

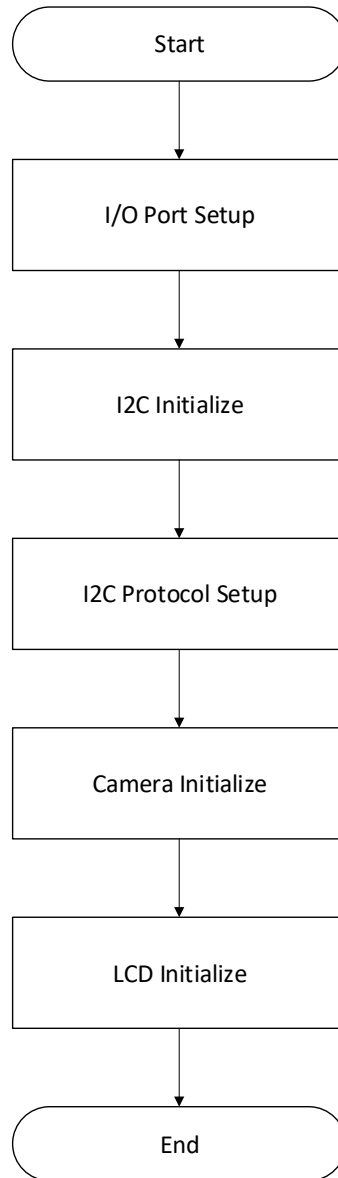


Figure 4.11: Hardware Initialization Flowchart

Figure 4.11 shows the step of the hardware setup. The first step is to set up the I/O port registers. The microcontroller connects to the camera, and collects image information from the I/O ports. The next step is to initialize the I2C registers and prepare for the I2C protocol function. In this research, we need to use the I2C protocol function to send three data to the peripheral components. The data includes peripheral component slave

address, register address and register data.

```
1 void IICWrite(unsigned int slave_add,unsigned int register_add,
2             unsigned char data)
3 {
4     IIC_Start;
5     IIC_Wait;
6     TWDR = slave_add;
7     IIC_Ack;
8     IIC_Wait;
9     TWDR = register_add;
10    IIC_Ack;
11    IIC_Wait;
12    TWDR = data;
13    IIC_Ack;
14    IIC_Wait;
15    IIC_Stop;
```

Listing 4.1: I2C Protocol Function

Next, using the I2C protocol function to initialize the camera and LCD display. OV7620 has its own setup registers. By setting the register's value, we can adjust the camera clock rate and output image resolution.

The crystal OV7620 is using 27MHz, and the microcontroller's speed is 20MHz. We need to use a prescaler to reduce the camera speed. The faster speed will create noise during data transfer. Dropping speed can reduce data noise. OV7620 default resolution is  $640 \times 480$ . We don't have to collect the full image. We only accept the interest area which is the water meter reading number display area. Shrinking the image size can reduce the image data collect time and can decrease the microcontroller SRAM size request. Through observation, we found the best image size is  $151 \times 41$ .



```
1 void cam_init(void)
2 {
3     IICWrite(0x42,0x11,0x08); // set clock rate
4     IICWrite(0x42,0x17,0x74); // setup window. Horizontal Window
    start
5     IICWrite(0x42,0x18,0x9B); // setup window. Horizontal Window end.
6     IICWrite(0x42,0x19,0x6A); // setup window. Vertical Window start
7     IICWrite(0x42,0x1A,0x95); // setup window. Vertical Window end.
8     // set window size 151*41
9 }
```

Listing 4.2: Camera Initialize Function

where 0x42 is the OV7620 slave address. Register 0x11 is the clock rate control. Setting prescaler to value 0x08. Register 0x17 is horizontal window start, register 0x18 is horizontal window end, register 0x19 is vertical window start and register 0x1A is vertical window end. By setting these four registers we can adjust the window size and location.

OV7620 uses EEPROM as a memory to store the register data, which will lose data once the power is off. It is better to reset the camera once we use it.

### 4.2.2 Image Data Collection

Once the system finishes setting the hardware, the next important step is receiving image data from the camera. OV7620 sends data via pins VSYNC, HREF, PCLK and Y<0:7>. VSYNC is the vertical sync pulse. HREF is the horizontal valid data output window. PCLK is the pixel clock used to clock valid data. Y<0:7> is the 8-bit luminance data bus.

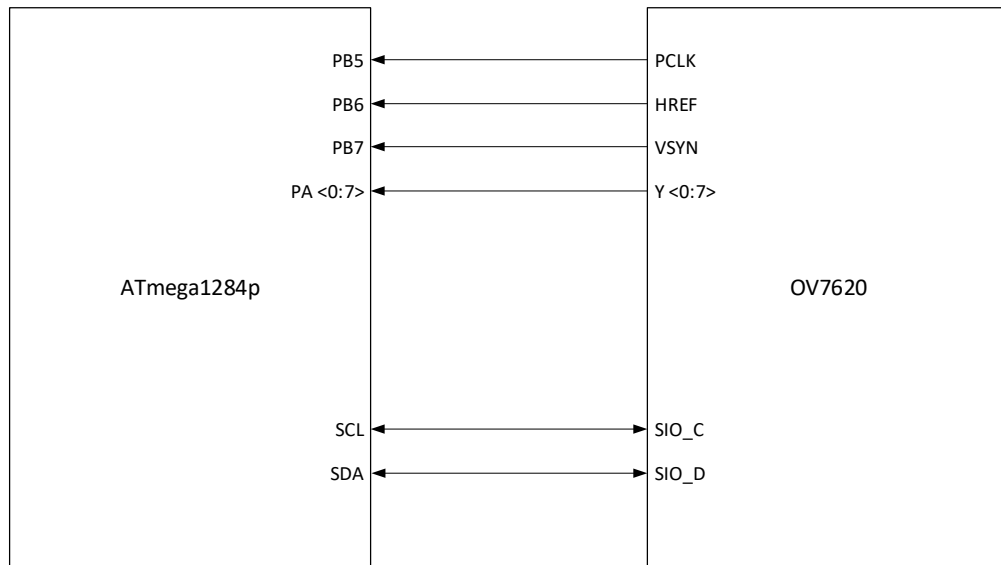


Figure 4.12: ATmega1284p And OV7620 Connection

```

1 void ImageCollect(void)
2 {
3     for(j=0;j<column;j++)
4     {
5         while(!(PINB&(1<<PINB7))); // VSYN down !VSYN
6         while(PINB&(1<<PINB7)); // VSYN up VSYN
7         for(i=0;i<row;i++)
8         {
9             while(PINB&(1<<PINB6)); // HREF up HREF
10            while(!(PINB&(1<<PINB6))); // HREF down !HREF
11            for(int a=0;a<=j;a++)
12            {
13                while(PINB&(1<<PINB5)); // PCLK up PCLK
14                while(!(PINB&(1<<PINB5))); // PCLK down !PCLK
15            }
16            Image_Data_1[i][j] = PINA;
17        }

```

```
18 }  
19 }
```

#### Listing 4.3: Image Collect Function

First, we use a while loop to wait for the VSYN signal. Once the VSYN signal changes from high to low, means a new image has started. The next step is checking the HREF signal. Once the HREF signal changes from low to high, means a new row has started. Then, we check the PCLK signal, once the PCLK signal changes from high to low, the microcontroller accepts the pixel data from data bus Y.

It will take time for the microcontroller to receive data and store them in memory. The next data may come at that moment. To avoid data corruption, we do not accept the whole row data once. The code is designed to accept the column data instead of accepting row data. Which means when every new row starts, the microcontroller only accepts one data, waiting for the next row start. The microcontroller has enough time to store the data to its memory. The first frame only accepts the first column data, the second frame only accept the second column data. To accept a  $151 \times 41$  resolution image, it totally accepts 151 frames. By comparing with the row data collection, the column data collect method can effectively avoid data loss.

### 4.2.3 Noise Removal Function

After receiving the whole data of one image, before binarization, we need to remove noise. Noise is the point which is fully white or black on the image. Too much noise will affect the recognition result. The noise showing on the image belongs to salt and pepper noise, so the median filter is used to remove them.

```
1 void NoiseRemoval(void)  
2 {
```

```
3     unsigned char D[9];
4     for(i=1;i<row-1;i++)          // Median filter Noise Removal
5     {
6         for(j=1;j<column-1;j++)
7         {
8             D[0]=Image_Data_1[i-1][j+1];
9             D[1]=Image_Data_1[i][j+1];
10            D[2]=Image_Data_1[i+1][j+1];
11            D[3]=Image_Data_1[i-1][j];
12            D[4]=Image_Data_1[i][j];
13            D[5]=Image_Data_1[i+1][j];
14            D[6]=Image_Data_1[i-1][j-1];
15            D[7]=Image_Data_1[i][j-1];
16            D[8]=Image_Data_1[i+1][j-1];
17            Image_Data_2[i][j]=medianfilter(D);
18        }
19    }
20 }
21 int medianfilter(unsigned char D[9])
22 {
23     unsigned int temp;
24     for(i=0;i<9;i++)
25     {
26         for(j=0;j<9-i;j++)
27         {
28             if(D[i]>D[j+1])
29             {
30                 temp=D[i];
31                 D[i]=D[j+1];
32                 D[j+1]=temp;
33             }
34         }
```

```
35     }  
36     return D[4];  
37 }
```

Listing 4.4: Noise Removal Function

The noise removal function is composed of two parts. First, we set a removable window on the image. The window size is  $3 \times 3$ , totally 9 elements in the window. Then we arrange these nine numbers in order, using the middle value instead of the middle number of that window.

#### 4.2.4 Binarization and Segmentation

After using the median filter to remove noise, the next step is transforming the image data to become 0 or 1. The key point of the binarization is to find the correct binarization threshold.

In this stage, we use Matlab to help us process data. We upload the image data after the noise removal step via the UART port to the computer and open Matlab. Matlab can display the data as an image. This visually aids us observe whether the process is as expected.

After using Matlab to find the suitable threshold value, we segment the image. In the recognition step, we only recognize a number once. Therefore, we need to segment each number out of the whole image. We use the projection method. We do the whole image processing as a horizontal projection and vertical projection, find the gap between each number, and segment the number from each gap.

### 4.2.5 Recognition Function

After the segmentation step, we start to recognize the number by using the proposed Moment Template Matching algorithm in Section 3.2.4.

First, we process the number of data by putting a two-dimensional image compressed into a one-dimensional array via the algorithm, and then compare it with the template. A template is the pre-typed data from numbers 0 to 9. We find the different value between input image and templates. The smallest value indicates the reading number. Recognizing the input number one-by-one, we will get four numbers of the reading.

```
1 void Recognition(struct number *nnum)
2 {
3     int k,min,l=0;
4     int Match[10];
5     int Reg[num_wide];
6     memset(Reg,0,num_wide*sizeof(int));
7     memset(Match,0,num_wide*sizeof(int));
8     for(j=nnum->left;j<=nnum->right;j++)
9     {
10         k=0;
11         for(i=nnum->bottom;i>=nnum->top;i--)
12         {
13             Reg[l]=Image_Data_1[i][j]*k+Reg[l];
14             k++;
15         }
16         l++;
17     }
18     for(i=0;i<10;i++)
19     {
20         for(j=0;j<19;j++)
21         {
22             Match[i]=fabs(Reg[j]-Template[i][j])+Match[i];
```

```
23     }
24 }
25 min = Match[0];
26 for(i=0;i<10;i++)
27 {
28     if(min>Match[i])
29     {
30         min=Match[i];
31         nnum->reading=i;
32     }
33 }
34 }
```

Listing 4.5: Recognition Function

### 4.3 Chapter Summary

In this chapter, we discuss the prototype design process. To make the prototype works, hardware design and software design are integrated. We start with the description of the electronic circuit, and then move on to the PCB design for each module. Finally, we discuss how to compile the image processing algorithm as software code.

## **Chapter 5**

# **Results and Discussions**

In this research, we use MATLAB as the platform to develop the whole software system. The first key task is to build up a hardware system, and the brain of the system is a microcontroller. Considering that we are going to process pictures, it is intuitive to observe the changes in the pictures. A microcontroller and a low-resolution LED cannot compete with this job. After connecting the camera with the microcontroller, the microcontroller receives the image data from the camera, and uploads the data to the computer.

### **5.1 Character Template**











In this research, the main idea is based on the template matching method. In the method, it is very strict with the template. The pre-populate templates and the recognition target need to have the same size, the same proportion or even the same font. The closer the template is to the target, the higher the recognition accuracy.

The best way to get the template is to select from the image we have got from the prototype. At the beginning of this research, we ordered a water meter which is the same as those used in New Zealand homes. Then, using a wooden frame to hold the



meter with the camera, we fixed the relative position between them. This is the most important step, as only by fixing the position can we get an effective image. The recognition target, which are the numbers showing on the meter will keep in the same position in different pictures. The wooden frame and the camera is the prototype of this research. Disassembling the water meter and manually adjusting the water meter display number from 0 to 9, shot by the camera, we obtained the required templates.

Table 5.1: Templates From 0 To 9

Number	Template
0	
1	
2	
3	
4	
5	
6	
7	
8	
9	

From Table 5.1 we can find the number displayed on the water meter is not the same as the printed letters. It is obvious to find in the number 7, there is a dot at the left bottom which is a tooth of the gear. In a water meter, each displayed number of the rotary table is connected by the gears, the previous digit rolling each circle, which is from 0 to 9,

the superior number turns one, such as from 1 to 2. The gap next to number 7 is helping gears, rolling the plate next to it. However, as the numbers 2, 3 and 5 have similar black in the same position, this feature is not a standard identification for number 7.

The size of each template is  $26 \times 19$ , which will match the size of the recognition target. In the program, each pixel is represented by numbers 0 or 1. In order to make the data visualisation, we used the MATLAB *imshow()* function. The *imshow()* function can display a binary data in a figure, the pixel with the value 0 displayed as black and value 1 displayed as white [62].

## 5.2 Experiment

### 5.2.1 Experimental Environment

The experiment of this research is divided into two parts.

- The first part of the experiment works on the computer. We use a water meter and a camera with fixed position, the camera connects with a microcontroller developing board, the developing board connect with a computer via a UART module. The model number of the water meter is Honeywell V100. The model number of the camera is OV7620. The microcontroller developing board uses ATmega1284P as MCU, with ISP/JTAG port and UART port. The UART module uses FT232RL chip, connecting with ATmega1284P TXD and RXD pin, transfers data from the microcontroller to the computer. The software used is MATLAB.
- The second part of the experiment works on the water meter reader. The water meter reader is the embedded digital system designed by this research. It uses ATmega1284P as a microcontroller, OV7620 as camera, a  $4 \times 1$  LCD monitor, illuminated by an SMD LED and powered by two 18650 lithium batteries.

### 5.2.2 Experiment Design

Part 1: Using MATLAB, we simulate the water meter reader interior process by using the algorithms mentioned in Section 3.2 to get recognition accuracy.

Part 2: Downloading the algorithm from MATLAB to the microcontroller we check how the algorithm works on the water meter reader.

The water meter used in the experiment is the same used in ordinary homes. In the laboratory, it is not cost-effective to connect the water meter inside a water flow system, using water to adjust the number showing on the water meter. A better way to adjust the display number is to disassemble the meter in parts and transmit the gears by hand. The water meter reader case was made by Acrylonitrile Butadiene Styrene (ABS) by a 3D printer, so multiple dismantling and installation will damage the connecting parts. In the experimental environment for the MATLAB process, the water meter is held by a wooden frame, which will not influence whether dismantling or installation. The other advantage of using MATLAB in the experiment, is that there is only one input data to calculate using different algorithms.

All the operation methods used in MATLAB are the same as used in the microcontroller in the water meter reader, so the result from MATLAB can represent the result of the water meter reader.





### 5.2.3 Image Data Collection

The first step of the experiment is receiving image data. Connecting camera, microcontroller developing board, UART module and computer together, we receive image data and save it in a text file.



The next step is to process the image. After noise removal, binarization, and segmentation, we have four separate numbers.

Table 5.2: Segmentation Result For Each Number

First Number	Second Number
	
Third Number	Fourth Number
	

Recognising numbers one-by-one, we save the result in a text file. The final recognised result shows in Figure 5.3.



Figure 5.3: Recognise Result

In Section 3.2, there are four recognition algorithms mentioned. All method evolution starts from the template matching algorithm. However, it requires a large system memory, which makes it difficult to be implemented in a microcontroller based system in this research. We put this algorithm as a benchmark, to compare the results with other algorithms.

There are four-digit numbers displayed on the water meter. The range for each digit number is from 0 to 9. The method for number selected for testing is each digit number chosen from 0 to 9 three times, plus some random number, testing a total of 200 sets of numbers. We use a template matching algorithm as the benchmark, projection template matching algorithm, geometric moments algorithm, and moment template matching algorithm as an experimental group to recognise samples.

An important technical index of the character recognition system is its identification accuracy, or character recognition success rate. In this research, we talk about the numbers displayed on the water meter. Based on the character recognition accuracy to determine the merits of the recognition algorithm the function of the recognition accuracy is:

$$\text{Recognition Accuracy} = \frac{\text{Number}_{\text{correct}}}{\text{Number}_{\text{total}}} \times 100\% \quad (5.1)$$

The recognition result showing in Table 5.3.

Table 5.3: Recognition Accuracy Comparison of Different Algorithms

Algorithm	Accuracy rate	Execution time
Template Matching Algorithm	99.9%	0.714418s
Projection Template Matching Algorithm	95.3%	0.656923s
Geometric Moments Algorithm	33.6%	1.504131s
Moment Template Matching Algorithm	98.1%	0.760704s

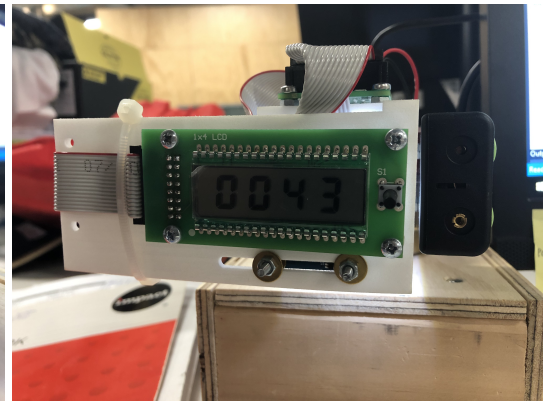
#### 5.2.4 Microcontroller-based OCR

In this part of the experiment, we use the water meter reader designed by this research. The reader case count is on the water meter. The camera faces towards the water meter, and the relative position between them is the same as experiment part 1. The objective of this part is to verify whether the algorithm designed in this research can work properly in the designed hardware system.

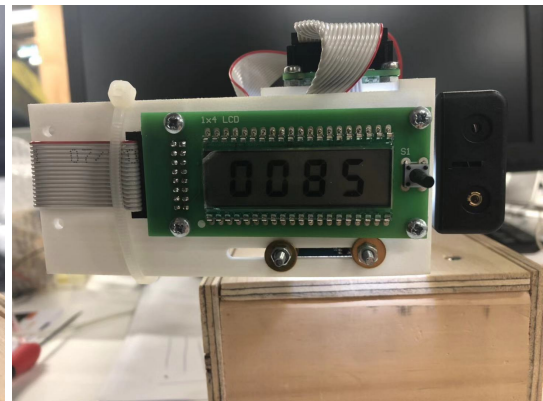
In experiment part 1, all methods are working on MATLAB. According to the MATLAB code, compile to the microcontroller C code. To test the result, we install the reader on to the water meter and turn on the power. Pushing the operating button, the machine will take some time to receive the image data and process, then display the result on the LCD.



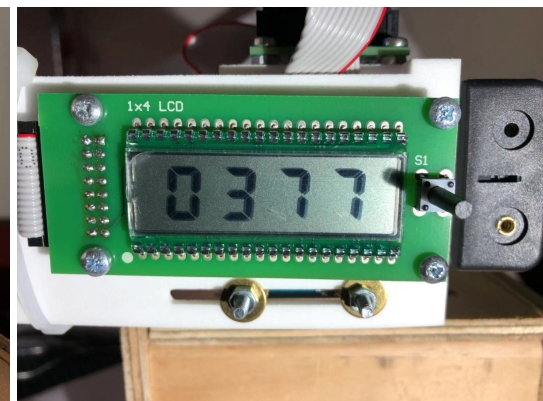
## CHAPTER 5. RESULTS AND DISCUSSIONS



(a) Reading Number Displayed on Water Meter (b) Reading Number Displayed on Meter Reader



(c) Reading Number Displayed on Water Meter (d) Reading Number Displayed on Meter Reader

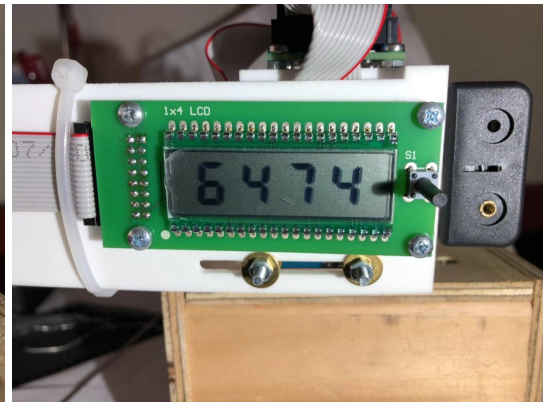


(e) Reading Number Displayed on Water Meter (f) Reading Number Displayed on Meter Reader

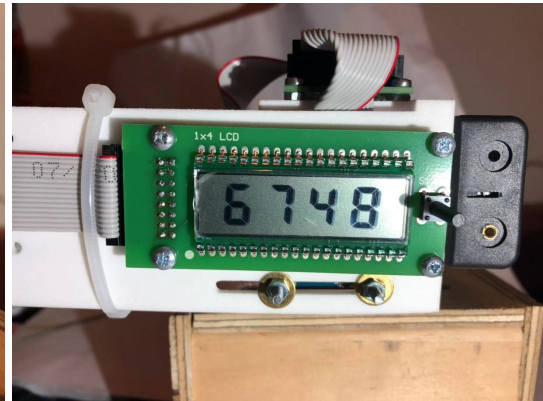
Figure 5.4: Meter Reader Result 1



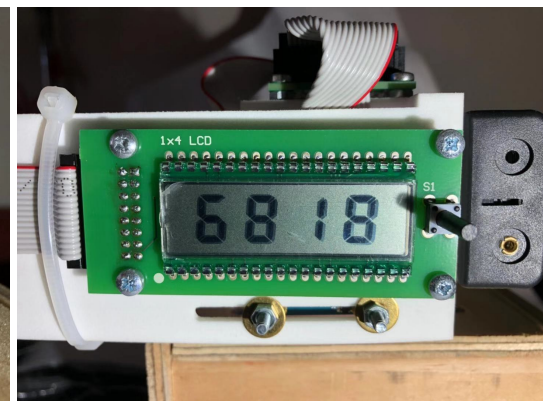
## CHAPTER 5. RESULTS AND DISCUSSIONS



(a) Reading Number Displayed on Water Meter (b) Reading Number Displayed on Meter Reader

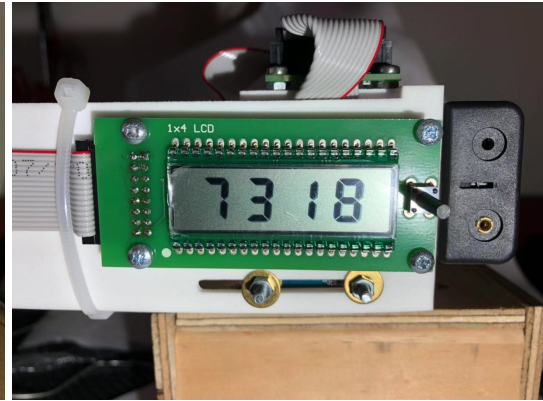


(c) Reading Number Displayed on Water Meter (d) Reading Number Displayed on Meter Reader

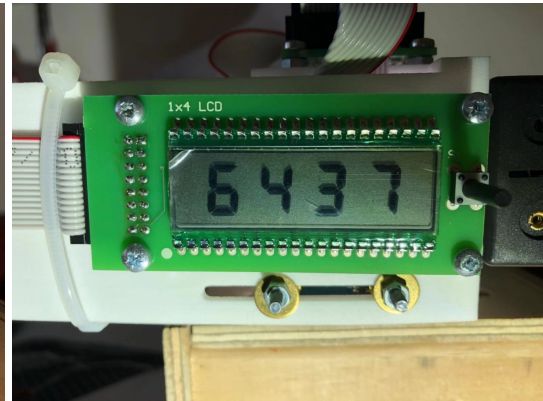


(e) Reading Number Displayed on Water Meter (f) Reading Number Displayed on Meter Reader

Figure 5.5: Meter Reader Result 2



(a) Reading Number Displayed on Water Meter (b) Reading Number Displayed on Meter Reader



(c) Reading Number Displayed on Water Meter (d) Reading Number Displayed on Meter Reader



(e) Reading Number Displayed on Water Meter (f) Reading Number Displayed on Meter Reader

Figure 5.6: Meter Reader Result 3

Figure 5.4, 5.5 and 5.6 showing the water meter reader result. Most of times, the prototype can get a right result. Figure 5.6(d) showing a wrong result. The mistake

happens at the first number recognition. The prototype misreads number 8 to be number 6.

### 5.3 Results Analysis

In experiment part 1, the result is shown in the Table 5.3. The highest accuracy rate is obtained by using the Template Matching Algorithm. But this algorithm takes the most system resources. It requires the highest system memory and occupies the longest run-time. The memory required to store the template exceeds the memory size of the selected cost-effective microcontroller, so this algorithm is not suitable for this research. The Geometric Moments Algorithm results in the lowest accuracy rate. An image geometric moments has translation, scale, and rotation invariant features, but the result shows this algorithm is not suitable for number recognition. The Projection Template Matching Algorithm develops from the Template Matching Algorithm. Through the projection method, it compresses a 2D image to a 1D array. Because it loses pixel coordinate information, the accuracy is lower than the Template Matching Algorithm. For the proposed Moment Template Matching Algorithm, it combines the projection and moments and adds the coordinate information during the compressing step. It results in a recognition accuracy higher than the Projection Template Matching Algorithm.

About the execution time, to compare with the proposed recognition algorithm, the Template Matching Algorithm and the Projection Template Matching Algorithm are faster than the Moment Template Matching Algorithm. However, the Template Matching Algorithm uses much memory resource than the proposed algorithm. The Projection Template Matching Algorithm has a lower accuracy rate than the proposed algorithm. The Geometric Moments Algorithm has the lowest accuracy rate and takes the longest execution time. The proposed recognition algorithm is not the fastest one, but compare

with the elementary algorithm–Template Matching Algorithm, the execution time of this algorithm is acceptable.

In experiment part 2, the result shows the algorithm can work on the real hardware. The proposed Moment Template Matching Algorithm works well in a resource-limited microcontroller-based OCR system. The water meter reader can recognise the number showing on the meter. However, we do find the recognition result is not as stable as using MATLAB. The possible reasons are discussed as follows.

1. Translation of Matlab code to microcontroller-based code results certain inconsistency, thus the current version of the software code has bugs. In a real working situation, the software occasionally crashed.
2. The connection part between meter and reader is not strong enough. If once operates the button hard, the whole reader shakes. The resulted blurry image causes the reading error.

### **5.4 Chapter Summary**

In this chapter, we presented the results obtained from the prototype. First, we used MATLAB to develop the algorithms and test the result for each image processing step. Then we compared the performance of our proposed OCR algorithm with existing OCR algorithms through Matlab simulations. Finally, we downloaded the code to the designed microcontroller-based OCR system, and verified that the proposed algorithm can work properly with excellent recognition accuracy.

## **Chapter 6**

# **Conclusion and Recommendation for Future Work**

### **6.1 Conclusion**

The objective of this thesis was to design a prototype that could recognise the number displayed on a water meter by using the OCR technique. For the prototype design part, we started from component selection, went through electronic circuit design, PCB design, software coding, and shell case design to finally develop a functional prototype machine. For the image recognition part, we started from accepting image data from the camera, by using image processing skill as noise removal, binarization, segmentation steps, to get a recognition target. We proposed a new recognition algorithm to discern the target. We also compared the recognition accuracy with other recognised algorithms.

We thus answered the research questions of this thesis as follows:

Question 1: Can we design a light-weight OCR algorithm for the digitisation of conventional mechanical water meter?

The answer is yes. Starting from microcontroller selection, we chose to use an AVR



microcontroller instead of a more powerful but more expensive ARM microcontroller. By considering the number of I/O pins, the size of built-in memory, peripheral features, running speed, and power supply voltage several conditions, we finally chose to use ATmega1284p as the processing unit. For the circuit design part, the whole MCU module only has 1 microcontroller, 15 resistors and capacitors, 1 crystal, 2 control buttons and 4 connect ports. We designed to use built-in memory instead of outside memory and use software code instead of a hardware binarization circuit. Simplifying the circuit can reduce the cost but still provide stable operation. According to the results display part, we used the self-design  $4 \times 1$  LCD screen, which has an advantage over existing LCD modules on the market, not only in terms of price but also in terms of lower energy consumption. About the power supply part, we chose to use two rechargeable lithium batteries, to give a hand to the protection of the environment. Finally, from the experiment result from section 5, the whole hardware system could complete the task.

Question 2: Can we implement the designed algorithm by using a low-cost microcontroller-based system while maintaining the required level of character recognition accuracy?

The answer is yes. Although a microcontroller compared with a computer is simple and crude, in this research, the microcontroller chip accomplished the mission. According to the power limit, we designed a suitable recognition algorithm to make a microcontroller using an OCR technique for the water meter number identification.

## **6.2 Recommendation for Future Work**

In this research, the final prototype can work normally according to our purpose. However, if the prototype is directly applied in actual production and daily life, there are still many aspects that need to be improved.

### 6.2.1 System Improvement

The following possible improvements can be considered.

1. The material used for the prototype case needs to be improved. In the final experiment, we found the case was out of shape slightly, which would affect the recognition result. The improvement approach is to use rigid materials and re-design the joints between the water meter and the meter reader. The design needs to consider the force and moment, reinforce stress points, and distribute the total weight to avoid most of the weight on one point.
2. A new version of the software can be developed. One can try to optimise the program logic, improve the operation speed, fix system bugs. The speed time for the recognition process will be shortened.
3. Recognition accuracy is directly proportional to the imaging quality received from the camera. The clearer and sharper image can improve the recognition success rate. OV7620 used in this research has many adjust parameters in terms of exposure settings. If possible, all parameters of the camera could be adjusted as a specific setting, in order to improve the output imaging quality of the camera.
4. The case of the prototype is not perfect. The LED used as a light source in the prototype, its position affects the image quality. It was found in the test when the light source and the camera are at a specific angle, the water meter number display window will reflect light to the lens. The miscellaneous light influences the image quality. For observing and adjusting the light source, the prototype case was designed as not fully enclosed. In the future, the case needs to be improved to be waterproof and dustproof because of the environment, to protect the electronic parts from external influence.

### 6.2.2 AMR Expect In The Future

The main topic of this research is the OCR technique applied to the conventional mechanical water meters in the context of AMR system. However, image recognition is only a part of the system. For a complete AMR system, the next step is to add the signal transmission part. Recognising the meter reading number and transmission of data back to the data centre constitutes a complete AMR system. As for the signal transmission part, here are some ideas for future research:

1. The signal transmission can use wired communications or wireless communications. If people consider using the wired communication type to transfer data, they could link the water meter to the power meter. The power meter used in New Zealand has realised the AMR system. Recording of the power meter does not require the meter reader staff as the power meter will send the data to the data centre automatically. If we could link the water meter to the power meter, the water meter record data could be sent via the power line. At the same time, if we connect the water meter with the power meter it will solve the water meter reader power supply problem. The meter reader could extract power from the power meter and omit the battery power supply module. This will reduce the production cost and solve the battery replacement problem.
2. If people choose to transmit data wirelessly, it can get a meter reader installed wireless transmission module. According to the data collection way, the selection of the wireless module can be done in the following ways:
  - Install a GSM communication module to the device [63]. The GSM module through a mobile phone SIM card can send water meter reading data to the data centre on a regular basis.
  - Install a ZigBee transmission module to the device. ZigBee is a new type



of wireless communication technology suitable for short range between the low data transmission rate of a series of electronic equipment [64]. Water meters in the same area could link together via a ZigBee device, to build up as a small local area network. Each device inside this network could transfer and receive data to each other. We only need to set a data collection device in each area, the water meter data through the transmission between each other, finally packed into the receiver, updates to the data centre.

- Install a Bluetooth or Wi-Fi module to the device. Bluetooth and Wi-Fi transmissions are the most extensive applications of wireless data transmission [65].

A meter recording staff only needs hand-held receiving devices in the reception area to directly download the data from the water meter [66]. Nowadays, a mobile phone is equipped with Bluetooth and Wi-Fi module. A smartphone can become a receiving device by exploiting the special application, so it is not necessary to develop a new receiver separately.

An alternative way is to use an unmanned aerial vehicle (UAV) as a tool for receiving data. Using UAV can reduce human expenditure. Install the data receiving equipment on the UAV, only need to design of UAV flight route in advance, operate the UAV to fly over the target area, the automatic meter reader automatically uploads the data to the receiver.

Connect the automatic meter reader to the smart home system. The smart home system with the popularity of smartphones has gradually begun to spread worldwide. Many of the artificial intelligence systems such as TmallGenie and XiaoiTongxue have already begun to help people control household electrical appliances in the home [67, 68]. After connecting the automatic meter reading system inside the smart home system, the user

## CHAPTER 6. CONCLUSION AND RECOMMENDATION FOR FUTURE WORK

could use a smartphone to check real-time water usage by application. The application could be designed to monitor the water use state, so if abnormal water usage happened, it can warn the user to check if there is any water pipe leaking.

# References

- [1] [Online]. Available: <https://www.microchip.com/wwwproducts/en/ATMEGA1284P>
- [2] G. Carlsson, “Using topological data analysis to understand the behavior of convolutional neural networks,” 2018. [Online]. Available: <https://www.ayasdi.com/blog/artificial-intelligence/using-topological-data-analysis-understand-behavior-convolutional-neural-networks/>
- [3] [Online]. Available: <https://my.oschina.net/u/876354/blog/1621839>
- [4] [Online]. Available: <http://www.thaimeters.co.th/wp-content/uploads/pdf/Kent-PSM.pdf>
- [5] [Online]. Available: <http://www.pmdo.com/download/OV7620.pdf>
- [6] [Online]. Available: <http://ww1.microchip.com/downloads/en/DeviceDoc/doc8059.pdf>
- [7] W. Li, T. Logenthiran, V. Phan, and W. L. Woo, “A novel smart energy theft system (sets) for iot-based smart home,” *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 5531–5539, June 2019.
- [8] M. Khan, B. N. Silva, and K. Han, “Internet of things based energy aware smart home control system,” *IEEE Access*, vol. 4, pp. 7556–7566, 2016.
- [9] C. Brasek, “Urban utilities warm up to the idea of wireless automatic meter reading,” *Computing and Control Engineering*, vol. 15, no. 6, pp. 10–14, 2004.
- [10] Q. Liu, N. Linge, and V. Lynch, “Implementation of automatic gas monitoring in a domestic energy management system,” *IEEE Transactions on Consumer Electronics*, vol. 58, no. 3, pp. 781–786, 2012.
- [11] Watercare, “Meter readings - get information on estimated, final and special readings,” 2019. [Online]. Available: <https://www.watercare.co.nz/Help-and-advice/Help-with-your-account/Meter-readings>

## REFERENCES

---

- [12] J. Song, Z. Li, M. R. Lyu, and S. Cai, "Recognition of merged characters based on forepart prediction, necessity-sufficiency matching, and character-adaptive masking," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 35, no. 1, pp. 2–11, 2005.
- [13] D. Berchmans and S. Kumar, "Optical character recognition: an overview and an insight," in *2014 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT)*. IEEE, 2014, pp. 1361–1365.
- [14] H. Herbert, "The history of ocr, optical character recognition," *Manchester Center, VT: Recognition Technologies Users Association*, 1982.
- [15] D. Ghosh, T. Dube, and A. Shivaprasad, "Script recognition—a review," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 12, pp. 2142–2161, 2010.
- [16] N. Stamatopoulos, B. Gatos, I. Pratikakis, and S. J. Perantonis, "Goal-oriented rectification of camera-based document images," *IEEE Transactions on Image Processing*, vol. 20, no. 4, pp. 910–920, 2010.
- [17] S. Lu, L. Li, and C. L. Tan, "Document image retrieval through word shape coding," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 11, pp. 1913–1918, 2008.
- [18] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 2nd ed. Prentice Hall, 2002.
- [19] R. J. Patan, M. Chakraborty, and S. S. Devi, "Blind color image de-convolution in yuv space," in *2016 International Conference on Communication and Signal Processing (ICCSP)*. IEEE, 2016, pp. 1551–1555.
- [20] K. Shin, I. Jang, and N. Kim, "Block adaptive binarisation of ill-conditioned business card images acquired in a pda using a modified quadratic filter," *IET Image Processing*, vol. 1, no. 1, pp. 56–66, 2007.
- [21] T. Sund and K. Eilertsen, "An algorithm for fast adaptive image binarization with applications in radiotherapy imaging," *IEEE Transactions on Medical Imaging*, vol. 22, no. 1, pp. 22–28, 2003.
- [22] Y.-M. Huang, M. K. Ng, and Y.-W. Wen, "Fast image restoration methods for impulse and gaussian noises removal," *IEEE Signal Processing Letters*, vol. 16, no. 6, pp. 457–460, 2009.
- [23] A. H. Sable and K. Jondhale, "Modified double bilateral filter for sharpness enhancement and noise removal," in *2010 International Conference on Advances in Computer Engineering*. IEEE, 2010, pp. 295–297.

## REFERENCES

---

- [24] S. Singh and A. K. Tiwari, "An enhanced decision based unsymmetric trimmed median filter for removal of high density salt and papper noise," in *2015 Fifth National Conference on Computer Vision, Pattern Recognition, Image Processing and Graphics (NCVPRIPG)*. IEEE, 2015, pp. 1–4.
- [25] S. Suhas and C. Venugopal, "Mri image preprocessing and noise removal technique using linear and nonlinear filters," in *2017 International Conference on Electrical, Electronics, Communication, Computer, and Optimization Techniques (ICEECOT)*. IEEE, 2017, pp. 1–4.
- [26] X. Yin and J. Zhu, "Salt-and-pepper noise removal based on nonlocal mean filter," in *2013 3rd International Conference on Consumer Electronics, Communications and Networks*. IEEE, 2013, pp. 577–580.
- [27] P. Shrivastava and U. P. Singh, "Noise removal using first order neighborhood mean filter," in *2014 Conference on IT in Business, Industry and Government (CSIBIG)*. IEEE, 2014, pp. 1–6.
- [28] G. George, R. M. Oommen, S. Shelly, S. S. Philipose, and A. M. Varghese, "A survey on various median filtering techniques for removal of impulse noise from digital image," in *2018 Conference on Emerging Devices and Smart Systems (ICEDSS)*. IEEE, 2018, pp. 235–238.
- [29] M. S. Darus, S. N. Sulaiman, I. S. Isa, Z. Hussain, N. M. Tahir, and N. A. M. Isa, "Modified hybrid median filter for removal of low density random-valued impulse noise in images," in *2016 6th IEEE International Conference on Control System, Computing and Engineering (ICCSCE)*. IEEE, 2016, pp. 528–533.
- [30] N. Singh, T. Thilagavathy, R. T. Lakshmipriya, and O. Umamaheswari, "Some studies on detection and filtering algorithms for the removal of random valued impulse noise," *IET Image Processing*, vol. 11, no. 11, pp. 953–963, 2017.
- [31] M. A. U. Ekram, A. Chaudhary, A. Yadav, J. Khanal, and S. Asian, "Book organization checking algorithm using image segmentation and ocr," in *2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS)*. IEEE, 2017, pp. 196–199.
- [32] M. A. B. Siddique, R. B. Arif, and M. M. R. Khan, "Digital image segmentation in matlab: A brief study on otsu's image thresholding," in *2018 International Conference on Innovation in Engineering and Technology (ICIET)*. IEEE, 2018, pp. 1–5.
- [33] I.-H. Lee and M. T. Mahmood, "Robust registration of cloudy satellite images using two-step segmentation," *IEEE Geoscience And Remote Sensing Letters*, vol. 12, no. 5, pp. 1121–1125, 2015.

## REFERENCES

---

- [34] L. Shen and R. M. Rangayyan, "A segmentation-based lossless image coding method for high-resolution medical image compression," *IEEE Transactions on Medical Imaging*, vol. 16, no. 3, pp. 301–307, 1997.
- [35] K. Haris, S. N. Efstratiadis, N. Maglaveras, and A. K. Katsaggelos, "Hybrid image segmentation using watersheds and fast region merging," *IEEE Transactions on Image Processing*, vol. 7, no. 12, pp. 1684–1699, 1998.
- [36] A. Pratondo, C.-K. Chui, and S.-H. Ong, "Robust edge-stop functions for edge-based active contour models in medical image segmentation," *IEEE Signal Processing Letters*, vol. 23, no. 2, pp. 222–226, 2015.
- [37] J. Chong, C. Tianhua, and J. Linhao, "License plate recognition based on edge detection algorithm," in *2013 Ninth International Conference on Intelligent Information Hiding and Multimedia Signal Processing*. IEEE, 2013, pp. 395–398.
- [38] C.-N. E. Anagnostopoulos, I. E. Anagnostopoulos, I. D. Psoroulas, V. Loumos, and E. Kayafas, "License plate recognition from still images and video sequences: A survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 9, no. 3, pp. 377–391, 2008.
- [39] S. Mori, C. Y. Suen, and K. Yamamoto, "Historical review of ocr research and development," *Proceedings of the IEEE*, vol. 80, no. 7, pp. 1029–1058, 1992.
- [40] B. Roberto, *Template Matching Techniques in Computer Vision: Theory and Practice*. John Wiley & Sons., 2009.
- [41] I. Talmi, R. Mechrez, and L. Zelnik-Manor, "Template matching with deformable diversity similarity," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 175–183.
- [42] S. Mattoccia, F. Tombari, and L. Di Stefano, "Fast full-search equivalent template matching by enhanced bounded correlation," *IEEE Transactions on Image Processing*, vol. 17, no. 4, pp. 528–538, April 2008.
- [43] L. Di Stefano and S. Mattoccia, "A sufficient condition based on the cauchy-schwarz inequality for efficient template matching," in *Proceedings 2003 International Conference on Image Processing (Cat. No.03CH37429)*, vol. 1, Sep. 2003, pp. I–269.
- [44] M. Sonka, V. Hlavac, and R. Boyle., *Image Processing, Analysis, and Machine Vision*. Cengage Learning., 2014.
- [45] W. Yaoming, "The geometric moment and its invariants," *Journal of Shanghai Dianji University*, vol. 9, no. 2, pp. 7–10, 2006.
- [46] M. Nixon and A. S. Aguado, *Feature extraction and image processing for computer vision*. Academic Press, 2012.

## REFERENCES

---

- [47] M.-K. Hu, "Visual pattern recognition by moment invariants," *IRE transactions on information theory*, vol. 8, no. 2, pp. 179–187, 1962.
- [48] X. Yuan, P. He, Q. Zhu, and X. Li, "Adversarial examples: Attacks and defenses for deep learning," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–20, 2019.
- [49] S. Zhang, S. M. H. Bamakan, Q. Qu, and S. Li, "Learning for personalized medicine: A comprehensive review from a deep learning perspective," *IEEE Reviews in Biomedical Engineering*, vol. 12, pp. 194–208, 2019.
- [50] H. Yanagisawa, T. Yamashita, and H. Watanabe, "A study on object detection method from manga images using cnn," in *2018 International Workshop on Advanced Image Technology (IWAIT)*, Jan 2018, pp. 1–4.
- [51] K. Dutta, P. Krishnan, M. Mathew, and C. V. Jawahar, "Improving cnn-rnn hybrid networks for handwriting recognition," in *2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR)*, Aug 2018, pp. 80–85.
- [52] Watercare, "Find and read your meter - read your meter to check water use or confirm a leak," 2019. [Online]. Available: <https://www.watercare.co.nz/Water-and-wastewater/Water-meters/Find-and-read-your-meter>
- [53] R. Garmabdari, S. Shafie, and M. M. Isa, "Sensory system for the electronic water meter," in *2012 IEEE International Conference on Circuits and Systems (ICCAS)*, Oct 2012, pp. 223–226.
- [54] L. Ma, D. McCann, and A. Hunt, "Combining magnetic induction tomography and electromagnetic velocity tomography for water continuous multiphase flows," *IEEE Sensors Journal*, vol. 17, no. 24, pp. 8271–8281, Dec 2017.
- [55] S. . Hsia, S. . Hsu, and Y. . Chang, "Remote monitoring and smart sensing for water meter system and leakage detection," *IET Wireless Sensor Systems*, vol. 2, no. 4, pp. 402–408, December 2012.
- [56] D. J. Cristaldi, S. Pennisi, and F. Pulvirenti., *Liquid crystal display drivers: Techniques and circuits*. Springer Science & Business Media, 2009.
- [57] P. Yin, C. Lu, Y. Chen, H. Liang, and S. Tseng, "A 10-bit low-power high-color-depth column driver with two-stage multi-channel rdacs for small-format tft-lcd driver ics," *Journal of Display Technology*, vol. 11, no. 12, pp. 1061–1068, Dec 2015.
- [58] A. Anagha and M. Mathurakani, "Prototyping of dual master i2c bus controller," in *2016 International Conference on Communication and Signal Processing (ICCSP)*, April 2016, pp. 2124–2129.

## REFERENCES

---

- [59] P. Schinagl and A. Sharp, “Algorithmic analysis and hardware implementation of a two-wire-interface communication analyser,” in *2017 Internet Technologies and Applications (ITA)*, Sep. 2017, pp. 189–193.
- [60] B. Eswari, N. Ponmagal, K. Preethi, and S. G. Sreejeesh, “Implementation of i2c master bus controller on fpga,” in *2013 International Conference on Communication and Signal Processing*, April 2013, pp. 1113–1116.
- [61] A. Malvino and D. J. Bates, *Electronic Principles*, 7th ed. McGraw-Hill, 2007.
- [62] I. The MathWorks, “Imshow – display image,” 2019. [Online]. Available: <https://au.mathworks.com/help/images/ref/imshow.html>
- [63] G. Gu and G. Peng, “The survey of gsm wireless communication system,” in *2010 International Conference on Computer and Information Application*, Dec 2010, pp. 121–124.
- [64] J. Liu and S. Hui, “The study of zigbee networking with wireless sensor,” in *2012 2nd International Conference on Consumer Electronics, Communications and Networks (CECNet)*, April 2012, pp. 1488–1492.
- [65] Y. Arakawa, Y. Sonoda, K. Tomoshige, S. Tagashira, and A. Fukuda, “Implementation of wifi/bluetooth-based smart narrow field communication,” in *2014 Seventh International Conference on Mobile Computing and Ubiquitous Networking (ICMU)*, Jan 2014, pp. 91–92.
- [66] Y. Li, X. Yan, L. Zeng, and H. Wu, “Research on water meter reading system based on lora communication,” in *2017 IEEE International Conference on Smart Grid and Smart Cities (ICSGSC)*, July 2017, pp. 248–251.
- [67] [Online]. Available: <https://xiaoi.mi.com>
- [68] [Online]. Available: <https://bot.tmall.com/equipment>