August 2017

Cooling Load Prediction for Different Building Types and Room Occupancy Detection Using Accelerometers

Mengfan Yang
University of Wisconsin-Milwaukee

COOLING LOAD PREDICTION FOR DIFFERENT BUILDING TYPES

AND

ROOM OCCUPANCY DETECTION USING ACCELEROMETERS

by

Mengfan Yang

A Thesis Submitted in

Partial Fulfillment of the

Requirements for the Degree of

Master of Science

in Engineering

at

The University of Wisconsin-Milwaukee

August 2017

ABSTRACT

COOLING LOAD PREDICTION FOR DIFFERENT BUILDING TYPES
AND
ROOM OCCUPANCY DETECTION USING ACCELEROMETERS

by

Mengfan Yang

The University of Wisconsin-Milwaukee, 2017
Under the Supervision of Dr. Lingfeng Wang

There are two parts in this thesis: the first part was conducted at UWM, and the second part was conducted at Johnson Controls using the knowledge and skills that I learned throughout my time in the Master's Degree program.

The primary purpose of my time at UWM was to compare different types of buildings with two popular machine learning regression algorithms, artificial neural network (ANN), supported vector machine regression (SVR) algorithms, and lastly to provide the results of my research to better help building managers make more informed decisions in regard to electrical utilities. The major objective is to use algorithms and neural networks to detect the occupancy of a room using real-time data from accelerometers. This data could then be used to enable HVAC systems to be more efficient and intelligent.

My research at UWM consists of 6 chapters. The background and related research are shown first in chapter 1 and chapter 2. Chapter 3 focuses on analyzing different building types, which aims to provide an overlook in the feature of the data. The basic concepts of ANN and SVR are included in the Chapter 4. The last chapter is the summary of internship in Johnson Controls during the summer. The project goal, data analysis and results are presented with details. A brief occupancy detection review of the industry as well as the basic knowledge of Wavelet Transform and K-means++ algorithm are also mentioned in Chapter 7.

The result of my research at UWM shows that it is necessary to apply different models for different building types if high accuracy is required. Compared to SVR, ANN is more accurate in all the building types. However, the difference of the accuracy depends on the building features. In a hospital, SVR and ANN both show high accuracy, but in restaurants, they are both underperforming. Additionally, using vibration magnitude measured from accelerometers to detect occupancy has proved to be feasible during the first stage. However, more complicated cases and patterns need to be considered and higher resolution sensors will need to be tested in the future work.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

The year studied in UWM is a life changing experience for me. On the day of completing this thesis, I feel blessed deep in my heart. This year is full of ups and downs, during which many people provided selfless and unconditional help to enlighten me, whether it is in academic, work or life aspect. Without them, this thesis could not have been completed, and I will not be as positive, ambitious and hopeful towards my life as I am right now.

Firstly, I would like to thank Dr. Lingfeng Wang for his continuous help and support from the day I became his student. He spares no effort to give me the best guidance. When my research did not go smoothly, he has been patient to me and gave me as much useful advice as he could. When I wanted to slow down, he sent me emails to encourage and remind me of what is the important thing at this stage of my life. His rigorous scholarship, spirit of diligent study and relentless pursuit of research deeply touched me and inspired me. In terms of life besides school, he is also willing to give me the selfless help. He often said to me "My goal is to get you more prepared for your future career." He has always been adhering to this belief to all the students in our lab, not just me. I am grateful to be one of his students and have his guidance. I am very lucky to meet such a great supervisor. Without his help, this thesis could not have been finished.

Secondly, I would like to express my appreciation to my manager Tim Gamroth. There are many twists and turns during the process that Johnson Controls hired me. My student visa brought a lot of inconvenience to HR and Mr. Gamroth. In order to take me on board, he spent much time communicating with HR, the legal partner and UWM Career Center. I am very grateful for his trust in me and giving me the opportunity to start my career in a great team. Even before I was on board, he introduced me to the whole group and gave me the opportunity to participate in the team's core projects. I learned so much from my amazing co-workers and so glad to be part of the team to win the second-runner prize in the Tech Challenge Day. After I started the internship in May, he also kept up with my project and provided me with as much help as he could. His wide-ranging engineering experience, expertise in the field of building management and great leadership amazed me every time when I talked with him. He has been leading the project for 2 months, and I was deeply impressed by his innovation ability. Without his help, this thesis would not have been completed as well.

Also, I want to thank Prof. David C Yu, who provided me the opportunity to study at UWM, and supported me in various ways. I want to thank my friends in the lab, Yingmeng Xiang for the advice in my research and other friends this year for their company and support. Additionally, I want to say thanks to my boyfriend Andrew Sobczak, for the care, patience, accompany and support during the process when I wrote my thesis.

Big thanks to my thesis committee: Drs. Chiu Tai Law and Dr. Chao Zhu for the great advice on my thesis.

Last but not least, I would like to thank my parents for their support and love.

Note: The intellectual property in Chapter7 is owned by Johnson Controls. If cited, please apply for authorization from Johnson Controls, or you will be held accountable.

Chapter 1 Introduction

1.1 Research Background

According to International Energy Agency (IEA) [1], the US electricity final consumption in 2014 is 3,787,793 GWh, which has been raised by 4.2% compared to the final consumption 3,636,065 GWh in 2004. The energy demand growth brings unpredictable challenges to the electric utilities. Energy forecasting becomes one of the hotspots of today's research.

The prediction of cooling energy in commercial buildings, as one of the energy forecasting fields, has attracted more and more attention because of the large proportion of energy consumption by HVAC (Heating, Ventilation and Air Conditioning), the diversity of the commercial cooling energy loads and its significant reference for building operation in both electrical utility side and building management side.

It is necessary to predict the commercial building cooling energy accurately because it takes a large proportion of energy consumption. In 2014, the energy consumption in commercial and public services accounts for around 35.6% of all final energy use in US [1]. Many sources show that the electricity consumption from HVAC system is the largest energy load in commercial and public services. In HVAC system, electricity is generally used by ventilation and air conditioning system while natural gas is used by heating systems. Since cooling system accounts for a large part of energy consumption in HVAC system, the prediction of its electricity consumption for both short-term and long-term is important.

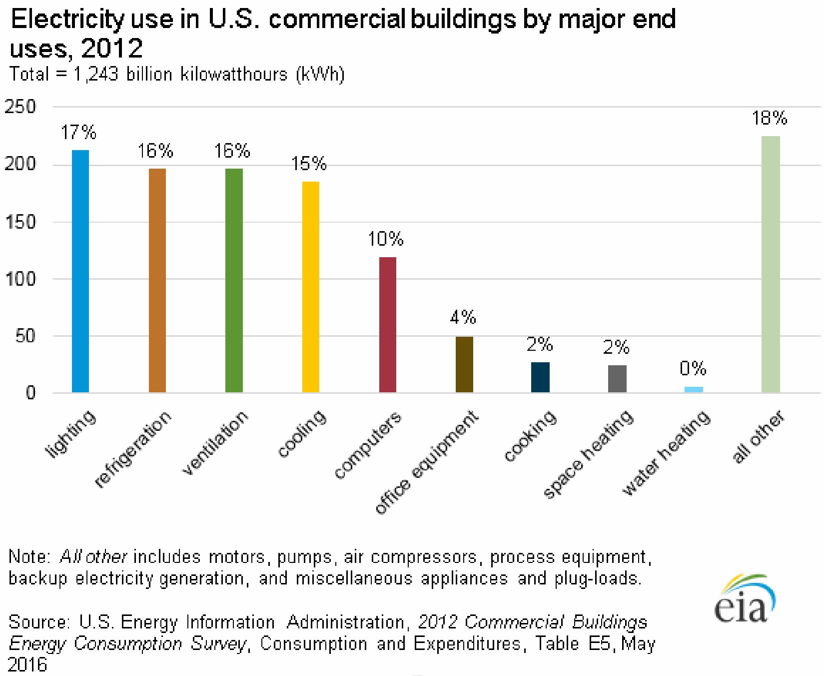
Apart from that, the cooling electricity prediction has an important impact in the HVAC design process. The prediction result can also be a significant reference for building planning and operation in both electrical utility side and building manager side. However, the diversity of the commercial cooling energy load brings much difficulty to predict the cooling load accurately. It is impossible to use one model to represent all the building types. Hence it is important to make comparisons of the performance among different building types before applying the same model to them.

1.2 Cooling electricity prediction in different types of commercial buildings

It is important to predict the cooling electricity based on the classification of commercial buildings. The reasons are discussed below from the commercial building side and the cooling load side.

On the commercial building side, it is necessary to classify the commercial buildings into several types. In the Commercial Buildings Energy Consumption Survey (CBECS), buildings are classified according to their principal activity, for example education, healthcare, food service, lodging, office, etc. [3]. Each type of commercial buildings has its own feature and pattern of electricity consumption.

On the cooling load side, predicting the cooling load has a lot of challenges. Different from lighting, refrigeration and ventilation electricity consumption, the consumption of cooling electricity relies heavily on external climatic conditions. As the external climatic conditions change from time to time, in order to maintain the customer comfort level in the building, the energy consumption of the cooling is also changed from time to time to ensure that the indoor temperature is consistent or slightly floating in a reasonable range according to the variation of occupancy. Therefore, the researches regarding predicting cooling electricity consumption,



although already exist for many years, there is still a lot of room to improve.

Figure 1-1 Electricity use in U.S. commercial buildings by major and uses

The energy consumption of buildings is greatly affected by the different types of buildings, especially the cooling electricity consumption. Three reasons are listed as follows. Firstly, different types of commercial buildings have different functions, which leads to the fact that the schedules of the buildings are not the same. Because of the different building schedules, the switching time of the HVAC system (different time per day, weekends and weekdays, statutory holidays) is different, which poses a significant challenge to the short-term prediction of cooling electricity. Secondly, different types of buildings have different architectural standards, such as the heat dissipation of external wall etc., If the electric utility operating crew lack of such information for each building and the related benchmark when they are doing short-term forecast energy forecast, it will cause big error in the final result. For example, when the operating crew in the electric utilities use the same forecasting model to predict all buildings in all types, it will result in inaccurate prediction. Thirdly, the occupants in different types of buildings have different requirements for their temperature comfort, which also results in different benchmarks for HVAC systems at runtime. Similarly, the lack of such benchmark will make it difficult to do forecasting with high accuracy. Therefore, the building types should be classified, and their forecasting models need to be established accordingly. For the electric utility managers, it is significant to be more aware of the energy consumption patterns of different types of buildings in order to carry out energy dispatching and planning for building communities. Since cooling electricity is the most difficult part to predict, this thesis will demonstrate cooling electricity consumption for different building types and present their forecast models.

1.3 Research Objective and Thesis Layout

This thesis classifies different types of buildings and uses intelligent algorithms to predict their short-term cooling energy consumption. The result could be used as a reference for electric utility to make better decision in the operation of buildings and power grids. Chapter 6, as an additional part, use accelerometers to detect occupancy. The result could be used in building management system (BMS) to help building operators save money and energy.

More specifically, Chapter 1 is the introduction of the thesis. Chapter 2 provides brief review of the related work. Chapter 3 analyzes the cooling electricity consumption features of 7 different building types. Chapter 4 elaborates on the intelligent algorithms that are presented in thesis, SVR and ANN to be specific. In Chapter 5, a case study is carried out. Chapter 6 is the additional part, which introduces the project I am working on in Johnson Controls. Chapter 7 will be the conclusion and future work of the thesis.

Chapter 2 Related Work

2.1 Cooling energy prediction method

2.1.1 ANN

ANN is known as the most widely used artificial intelligence models in the application of building energy prediction due to its ability to solve non-linear and complex problems. Since Warren McCulloch and Walter Pitts created a computational model for neural networks based on mathematics and algorithms in 1943, ANN has been rapidly developed in different areas of building intelligence and management. Especially in the past twenty years, ANN has been applied to analyze various types of building energy consumption in a variety of conditions. In this part, the previous studies about one of the applications, the cooling load prediction, is reviewed.

In Yokoyama, et al.'s work [14], building cooling demand was predicted by a back propagation (BP) neural network (NN). In this work, a new method for identifying model parameters, which is global optimization method called modal trimming, was proposed. In Kreider et al.'s work

[15], building cooling energy is predicted by a recurrent neural network. The inputs of the network are hourly energy consumption data, the external weather and the time stamp. Ben-Nakhi and Mahmoud [16] predicted the cooling load of three office buildings using the real data set of 3 years and similar with Wong et al.'s work [21] that they predicted energy consumption for office buildings. In Yan and Yao's work [17], building's heating and cooling load in different climate zones are predicted by using a BPNN. The study [18] has successfully applied NNs for predicting hourly electricity consumption for an engineering center building. In Nizami and Al-Garni's work [19], they use the occupancy and weather data as the input to predict electric energy consumption using a feed-forward neural network (FFNN). González and Zamarreno [20] also used FFNN to predict short-term electricity load.

The research work above is about the general application using ANN to predict energy consumption. To be more specific, ANN could also be used to analyze and optimize HVAC systems by the prediction result. Hou, et al. [22] predicted air-conditioning (AC) load in a building, which is a key to the optimal control of the HVAC system. In [23], general regression neural networks (GRNN) were used to optimize HVAC thermal energy storage for public and office buildings. Yalcintas, et al. [24] used NNs to predict chiller plant energy use of a building in a tropical climate. Gouda et al. [25] used a multi-layered feed-forward neural network (MLP-NN) to predict internal temperature with inputs, including external temperature, heating valve position, solar irradiance and the building internal temperature.

Previously, a variety of ANN models have been used in building cooling load forecast, for example, back propagation neural network, recurrent neural network, feed-forward neural network, general regression neural networks, etc. Most of these models are targeted for the commercial buildings, whose data is obtained from office building data. There is a lack of validation of their application with different types of commercial buildings.

2.1.2 SVR

SVR is more and more used in research and industry due to its ability to solve non-linear problems even with limited training data. SVR algorithm was invented by Vladimir N. Vapnik and Alexey Ya. Chervonenkis in 1963. In 1992, Bernhard E. Boser, Isabelle M. Guyon and Vladimir N. Vapnik applied the kernel trick to maximum-margin hyperplanes and created nonlinear classifiers [26]. After that, SVR is widely used in different areas in industry.

Dong et al. are the first researchers to apply SVR to predict the monthly electricity consumption of four buildings and the result shows good performances [27]. Later on, Lai [28] applied SVR model to a building with one year electricity consumption data record. After that the work [29] used SVRs to predict the hourly cooling load of an office building. The result shows that, based on the office building data, the performance of the SVR is better than the BPNNs. In terms of time-series prediction, Hou and Lian [30] did an experiment to show that SVRs has better performance than the ARIMA model when applying SVR to predict cooling load of the HVAC system.

All the work above seems to show that SVRs can have good performance in the application of predicting hourly and monthly building energy. However, we cannot jump to the conclusion that SVRs have better performance because the previous experiments were conducted on a small quantities buildings with limited types and the result may not be accurate enough. In paper [31], the author predicted the annual electricity consumption of buildings by BPNN, RBF neural networks, general regression neural networks and SVRs. They experiment result shows that general regression neural networks and SVRs were more applicable to this problem. Later on, in order to reduce the dimension of input variables, Researcher Lv [32] used PCA (Principal Component Analysis) to reduce variables before training SVRs for predicting building cooling load. Based on that work, paper [33] used Kernel Principal Component Analysis (KPCA), which is an improved PCA, to pre-process the data before training SVRs.

Chapter 3 Building Type Analysis

In this chapter, the analysis of building type will be addressed and several indices regarding building type will be proposed. CBECS divides all commercial buildings into 14 categories, as

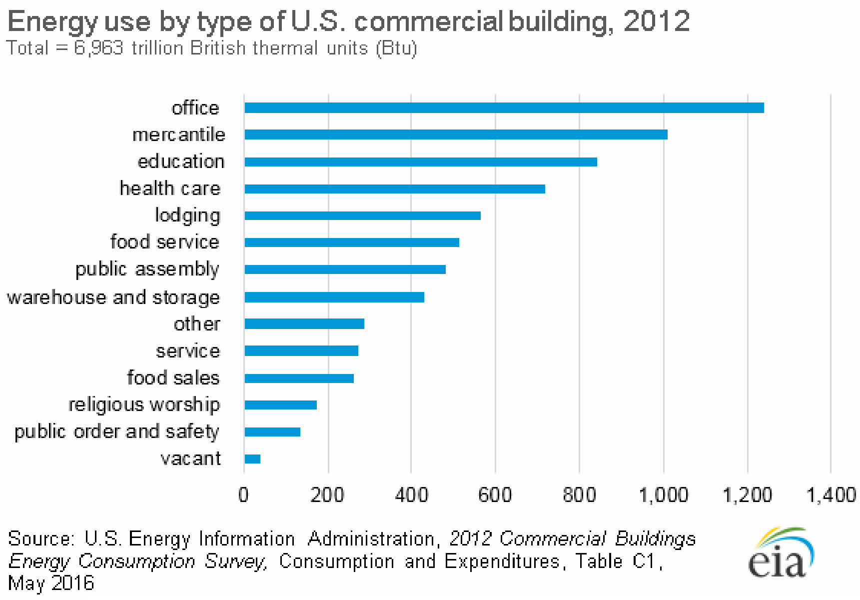


Figure 3-1 Energy use by type of U.S. commercial building

showed in Figure 3-1. Among the 14 categories, office building consumes most energy and the vacant buildings consumes the least energy. In view of our familiarity with the building type and the ranking of energy consumption, I have selected a few types of buildings for analysis. They are office, mercantile, education, health care, lodging and food service. In the next two sections of this chapter, I will analyze these types of buildings one by one.

3.1 Building type feature analysis

A benchmark metric named national median energy use intensity (EUI) [5] is used in all commercial buildings. The reason why it uses he median value instead of mean value is because the median value could accurately reflect the energy use midpoint for most property types. In the table 1 below, two metrics are listed, median Source EUI and median Site EUI.

Table 1 U.S. National Median Reference Values for All Types (Extracted from [5])

Building type	Further breakdown	Source EUI (kBtu/ft2)	Site EUI (kBtu/ft2)	Reference Data Source – Peer Group Comparison
Health care	Hospital	389.8	196.9	CBECS - Inpatient Healthcare
Lodging	Hotel	162.1	73.4	CBECS - Hotel & Motel/Inn
Office	Large Office	148.1	67.3	CBECS - Office & Bank/Financial
Education	Elementary school	141.4	58.2	CBECS - Elementary/Middle & High School
Mercantile	Strip mall	237.6	94.2	CBECS - Strip Shopping Mall
Lodging	Apartment	114.9	73.9	CBECS - Dormitory
Food sales	Full service restaurant	432.0	223.8	CBECS - Restaurant/Cafeteria

Site EUI is the metric that we are all familiar with on the utility bills. Site EUI contains a mixture of primary energy and secondary energy. While Source EUI combines primary and secondary energy types into a single common unit to provide an equitable way to compare buildings in different types. Thus, Source EUI will be used in this thesis.

Apart from the national Median Source EUI that almost all the buildings have, some buildings also have a 1-100 ENERGY STAR Score [6]. Similar to the EUI values, the score evaluates a building comparing to its peers and also adjusts for climate and business activity.

There are 7 different building types in Chapter 4's case study. Here I describe the details of each building for future reference. The building types that used in the case study are office, mercantile, education, health care, lodging and food service. Figure 3-2 shows the cooling

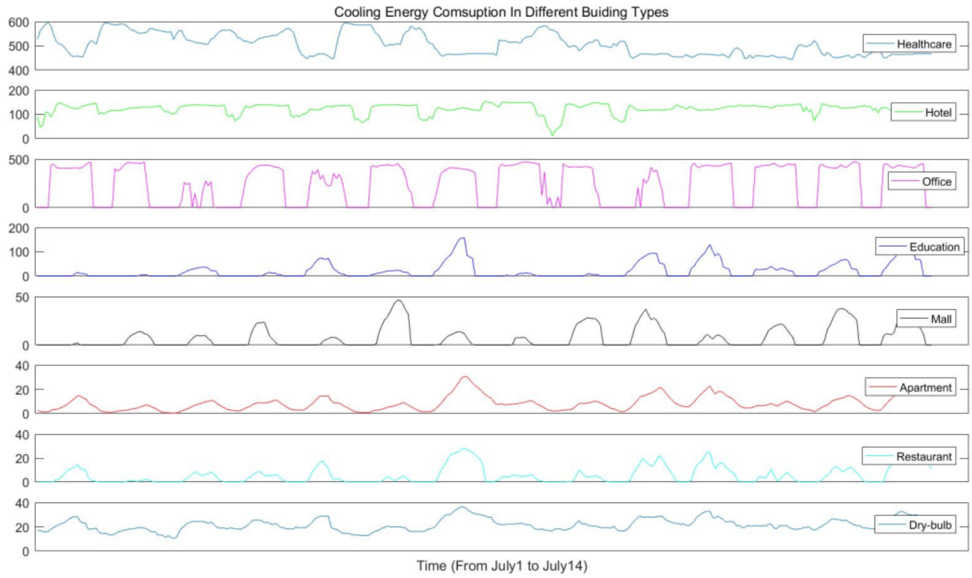


Figure 3-2 Cooling electricity load in different building types

energy consumption, with seven different buildings during the time period July 1^{st} to July 14^{th}.

I use hospital building cooling electricity consumption to represent the category of Health care. Hospital here represents the general medical and surgical hospital. From Figure 3-2, we can see that the cooling electricity load during the time period July 1^{st} to July 14^{th} keeps a high level and does not vary with the change of time/date and weather. This is because the hospital's energy consumption is mainly concentrated in (more than 50%) long-term care. These energy consumption is relatively stable. Hospital load will change slightly with the fluent external temperature. In order to maintain the indoor temperature for the user's comfort, the cooling load will fluctuate slightly with the temperature, but the overall impact is not significant.

Hotel provides overnight short-term accommodations. Office refers to buildings that has a different schedule among weekdays, Saturday and Sunday. Education Building is the type of building which is used for academic or technical classroom instruction. The dormitory building in campus should be classified as “lodging” building type, which is not included in this category. In this example, the data is extracted from elementary school buildings. Within a week, the feature of cooling electricity consumption has changed significantly due to the difference between the weekend and weekdays. From Monday to Friday, the energy consumption is relatively stable following a same pattern. However, the daily peak energy consumption will vary due to the changes in temperature. On the weekend, the cooling electricity consumption is almost 0 because there is almost no occupancy during that time. We could easily identify which period is weekdays and which is weekends by education lines from in figure3-2. Within a day, the feature of cooling electricity consumption has changed significantly due to the difference of time. Typically, there is cooling electricity consumed from 9 am to 5 pm during weekdays, at noon due to the temperature rise, its electricity consumption will reach its peak. It is noteworthy that the cooling electricity consumption of restaurant type will have a relatively low trough at noon every day, which is the most obvious feature of this type.

Different from hotel building type which offers multiple accommodations for short-term residents, the apartment building offers for long-term residents. The energy dissipation curve of apartment building type varies with the changes of temperature curve. The main reason is that the users of this type of building are long-term residence, in order to ensure the comfort level, the cooling system must be kept running from time to time, and to change with the external temperature changes.

Restaurant building is classified as the building which is used for retail or wholesale of food. The example showed in figure3-2 uses the full-service restaurant data. The most important feature of hotel type building energy consumption, from this map, is that there will be energy consumption of the trough at noon every day.

Chapter 4 Methodology

4.1 SVM Regression (or SVR)

Support vector machine (SVM) analysis is a machine learning tool for classification and regression. SVR stands for support vector machines for regression models. SVR is considered to be a nonparametric technique because it relies on kernel functions.

The SVR calculating goal is to seek and optimize the generalization bounds. Usually it will define a function called loss function, which ignores errors. Those errors are situated within the certain distance of the true value. Generally, there are two types of SVM Regression, linear and nonlinear as showed on Figure 4-1 and Figure 4-2.

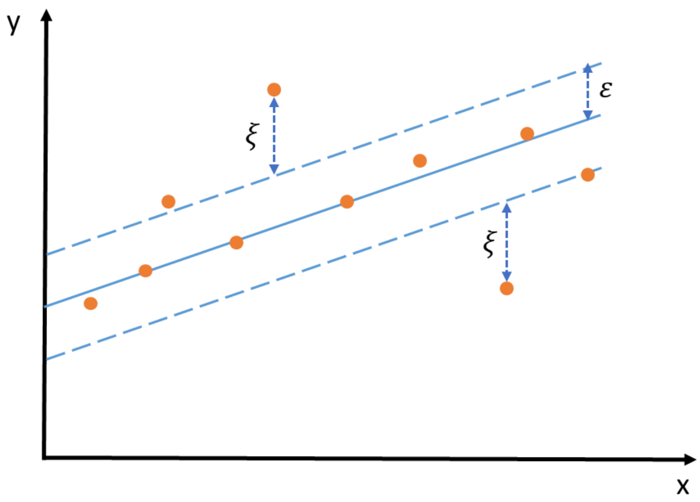


Figure 4-1 One-dimensional linear regression with epsilon intensive band.

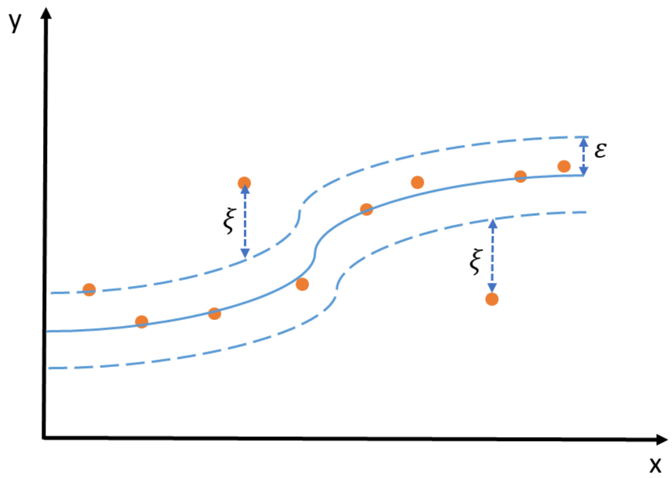


Figure 4-2 One-dimensional nonlinear regression with epsilon intensive band.

4.1.1 Linear SVM Regression: Primal Formula

In order to find the linear function

$$f(x) = x'\beta + b \tag{4.1}$$

We suppose we have a set of training data, where x_n is a multivariate set of N observations with target value y_n.

To make the linear model as flat as possible, we should minimal norm value $(\beta'\beta)$ to make it formulated as a convex optimization problem to minimize.

$$J(\beta) = \frac{1}{2}\beta'\beta \tag{4.2}$$

This optimized problem should subject to all residuals having a value less than ε, showed in equation form as (4.3).

$$\forall n : \left| y_n - (x_n{}'\beta + b) \right| \le \varepsilon \tag{4.3}$$

Ideally, there is such a function $f(x)$ existing to satisfy these constraints for all points. In most of the case, we have to introduce slack variables ξ_n and $\xi_n{}^*$ for each point, to deal with other infeasible constraints. The slack variables allow regression errors to exist up to the value of ξ_n and $\xi_n{}^*$, yet still satisfy the required conditions. Including slack variables leads to the objective function, also known as the primal formula [9]:

$$J(\beta) = \frac{1}{2}\beta'\beta + C\sum_{n=1}^{N}(\xi_n + \xi_n{}^*) \tag{4.4}$$

Subject to:

$$\begin{aligned}
&\forall n : y_n - (x_n{}'\beta + b) \le \varepsilon + \xi_n \\
&\forall n : (x_n{}'\beta + b) - y_n \le \varepsilon + \xi_n{}^* \\
&\forall n : \xi_n{}^* \ge 0 \\
&\forall n : \xi_n \ge 0
\end{aligned} \tag{4.5}$$

The constant C is a positive numeric value that controls the penalty imposed on observations that lie outside the epsilon margin ε and helps to prevent overfitting. This value determines the trade-off between the flatness of $f(x)$ and the amount up to which deviations larger than ε are tolerated.

The linear ε-insensitive loss function ignores errors that are within ε distance of the observed value by treating them as equal to zero. The loss is measured by the distance between observed value y and the ε boundary. This is formally described by:

$$L_{\varepsilon} = \begin{cases} 0, |y - f(x)| \leq \varepsilon \\ |y - f(x)| - \varepsilon, otherwise \end{cases} \tag{4.6}$$

4.1.2 Linear SVM Regression: Dual Formula

The difference between dual problem and primal problem is that the dual problem provides a lower bound to the solution of the primal problem. The optimal values of the primal and dual problems should not be equal. The difference between the primal and dual optimal value is defined as the "duality gap." However, when it is a convex problem and satisfies a constraint qualification condition, the primal problem’s optimal solution value to is given by the solution of the dual problem. By constructing a Lagrangian function from the primal function with introducing nonnegative multipliers α_n and α_n^* for each observation x_n, we could obtain the dual formula as (4.7), where we minimize

$$L(a) = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)x_i' x_j + \varepsilon\sum_{i=1}^{N}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{N} y_i(\alpha_i - \alpha_i^*) \tag{4.7}$$

Subject to the constraints:

$$\begin{aligned} &\sum_{n=1}^{N}(\alpha_n - \alpha_n^*)x_n = 0 \\ &\forall n: 0 \leq \alpha_n \leq C \\ &\forall n: 0 \leq \alpha_n^* \leq C \end{aligned} \tag{4.8}$$

Using the equation (4.9), the β parameter can be completely described as a linear combination of the training observations and function is equal to (4.10).

$$\beta = \sum_{n=1}^{N} (\alpha_n - \alpha_n^{*}) x_n \tag{4.9}$$

$$f(x) = \sum_{n=1}^{N} (\alpha_n - \alpha_n^{*})(x_n' x) + b \tag{4.10}$$

For linear SVM regression, Karush-Kuhn-Tucker (KKT) complementarity conditions, which are optimization constraints required to obtain optimal solutions, is showed as (4.11).

$$\begin{aligned} &\forall n: \alpha_n(\varepsilon + \xi_n - y_n + x_n'\beta + b) = 0 \\ &\forall n: \alpha_n^{*}(\varepsilon + \xi_n^{*} + y_n - x_n'\beta - b) = 0 \\ &\forall n: \xi_n(C - \alpha_n) = 0 \\ &\forall n: \xi_n(C - \alpha_n^{*}) = 0 \end{aligned} \tag{4.11}$$

4.1.3 Nonlinear SVM Regression: Primal Formula

In some cases, the problem is too complicated to be described as a linear problem. In order to get a better accuracy, linear functions can be extended to nonlinear functions by using the Lagrange dual formulation. More specifically, nonlinear SVM regression model is obtained by replacing $x_1' x_2$ with a nonlinear kernel function G(x1,x2) = <φ(x1),φ(x2)>, where φ(x) is a transformation that maps x to a high-dimensional space. Some kernels are showed below.

Table 2 Kernel Functions of nonlinear SVM Regression

Kernel Name	Kernel Function
Linear (dot product)	$G(x_1, x_2) = x_1' x_2$
Gaussian	$G(x_1, x_2) = \exp(-\lVert x_1 - x_2 \rVert^2)$
Polynomial	$G(x_1, x_2) = (1 + x_1' x_2)^p$, where p is in the set {2,3,...}.

4.1.4 Nonlinear SVM Regression: Dual Formula

By replacing $x_i{}'x_j$ in (4.7) with $G(x_i, x_j)$, which is the corresponding element of the Gram matrix, the dual formula for minimizing nonlinear SVM Regression will be

$$L(\alpha) = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)G(x_i, x_j) + \varepsilon\sum_{i=1}^{N}(\alpha_i + \alpha_i^*) - \sum_{i=1}^{N} y_i(\alpha_i - \alpha_i^*) \quad (4.12)$$

subject to

$$\begin{aligned} &\sum_{n=1}^{N}(\alpha_n - \alpha_n^*) = 0 \\ &\forall n: 0 \le \alpha_n \le C \\ &\forall n: 0 \le \alpha_n^* \le C \end{aligned} \quad (4.13)$$

The KKT complementarity conditions are

$$\begin{aligned} &\forall n: \alpha_n(\varepsilon + \xi_n - y_n + f(x_n)) = 0 \\ &\forall n: \alpha_n^*(\varepsilon + \xi_n^* + y_n - f(x_n)) = 0 \\ &\forall n: \xi_n(C - \alpha_n) = 0 \\ &\forall n: \xi_n^*(C - \alpha_n^*) = 0 \end{aligned} \quad (4.14)$$

4.1.5 Solving the SVM Regression Optimization Problem

SVM Regression optimization problem is essentially a minimization problem, which could be solved by expressing in standard quadratic programming form and then being solved using common quadratic programming techniques. However, one shortcoming of quadratic programming algorithms is that it might be computationally expensive and the Gram matrix might be too large to be stored in memory. In order to solve that problem, we introduce a decomposition method. Decomposition methods detached all observations into two disjoint sets: the working set and the remaining set. A decomposition method modifies only the elements in the working set in each iteration. Therefore, only some columns of the Gram matrix are needed in each iteration, which reduces the amount of storage needed for each iteration.

For all the approaches to solve SVM problems, sequential minimal optimization (SMO) is the most popular one [12]. SMO does a series of two-point optimizations. In each iteration, a working set of two points are chosen based on a selection rule that uses second-order data. Then the Lagrange multipliers for this working set are solved analytically using the approach described in [10] and [11].

In SVM regression, the gradient vector ∇L for the active set is updated after each iteration. The decomposed equation for the gradient vector is

$$(\nabla L)_n = \begin{cases} \sum_{i=1}^{N} (\alpha_i - \alpha_i^*) G(x_i, x_n) + \varepsilon - y_n, n \le N \\ -\sum_{i=1}^{N} (\alpha_i - \alpha_i^*) G(x_i, x_n) + \varepsilon + y_n, n > N \end{cases} \tag{4.15}$$

The program is kept running until the specified convergence criterion is met. There are several options for convergence criteria that made the program stop running. First of all, the feasibility gap, defined as (4.16) could be evaluated by the software at each iteration. If it is less than the value specified by GapTolerance, then the algorithm met the convergence criterion and the software returns a solution.

$$\Delta = \frac{J(\beta) + L(\alpha)}{J(\beta) + 1} \tag{4.16}$$

Secondly, the gradient difference, gradient vector ∇L between current iteration and the previous iteration specifically, could be used to evaluate the convergence criterion. If the difference in gradient vector values for the current iteration and the previous iteration is less than the value specified by Delta Gradient Tolerance, then the algorithm met the convergence criterion and the software returns a solution. Thirdly, largest KKT violation could be used as a convergence criterion. After each iteration, the software evaluates the KKT violation for all the α_n and α_n^*values. If the largest violation is less than the value specified by KKT

Tolerance, then the algorithm met the convergence criterion and the software returns a solution.

4.2 ANN

A basic neuron with R inputs is shown in Figure 4-3. Each input is weighted with an appropriate w. The sum of the weighted inputs and the bias forms the input to the transfer function f. Different neurons can have different transfer functions to generate their output.

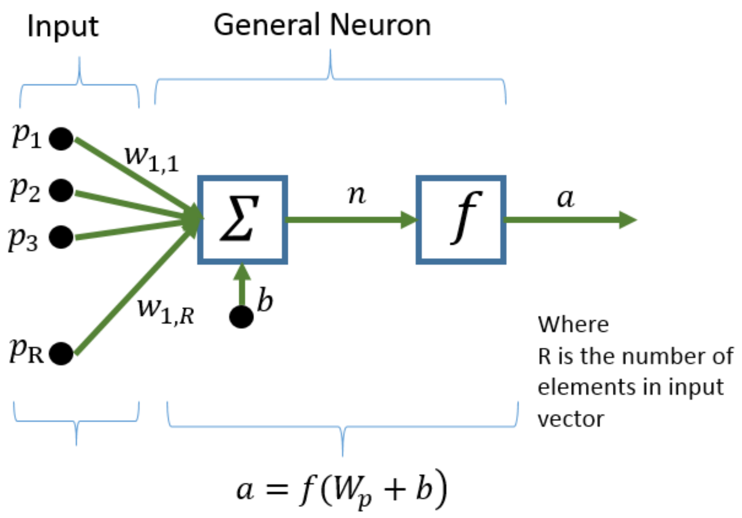


Figure 4-3 General Neural Network Architecture

Several transfer functions could be used in multilayer neural network, for example log-sigmoid transfer function, tan-sigmoid transfer function and linear transfer function. Among them, log-sigmoid transfer function is the most widely used one.

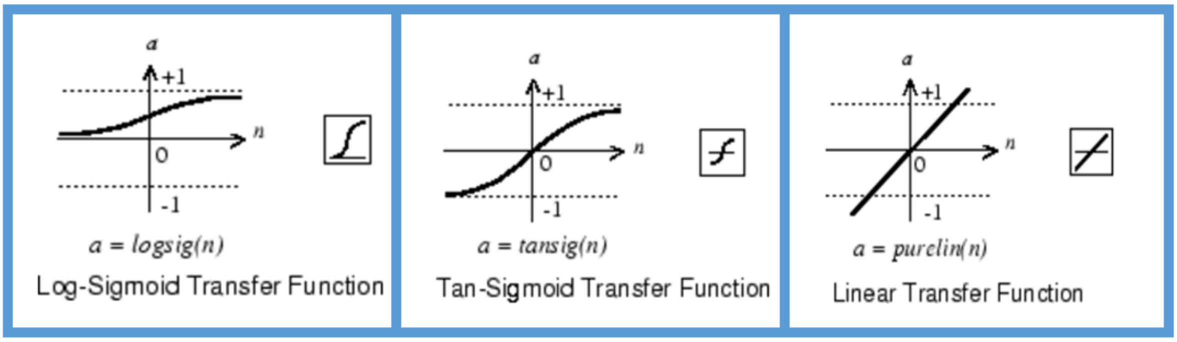


Figure 4-4 3 types of transfer functions in neural network

Feedforward networks often have one or more hidden layers of sigmoid neurons followed by an output layer of linear neurons. The network could learn nonlinear relationships between input and output vectors through multiple layers of neurons with nonlinear transfer functions.

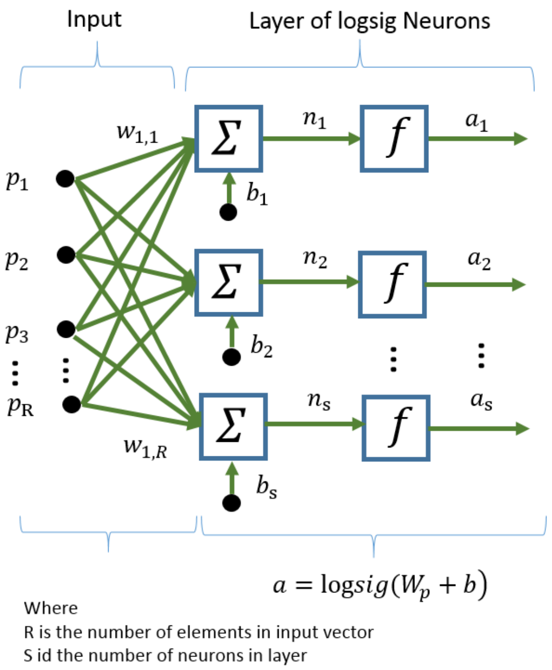


Figure 4-5 A single-layer network of S logsig neurons having R inputs in full detail

The network is ready for training when the network weights and biases are initialized. The training process entails network inputs and target outputs. The process of training a neural network is to optimize network performance by tuning the values of the weights and biases. The performance function that used in the feedforward networks experiment later is the mean square error (MSE) —the average squared error between the network outputs a and the target outputs t. It is defined as follows:

$$F = mse = \frac{1}{N}\sum_{i=1}^{N}(e_i)^2 = \frac{1}{N}\sum_{i=1}^{N}(t_i - a_i)^2 \tag{4.17}$$

There are two modes in neural network training, incremental mode and batch mode. Incremental mode is to apply input to network first and then to update the weights and computing the gradient. Batch mode is to apply all the inputs to the network and then update the weights. For most problems, when using the Matlab Neural Network Toolbox™ software, batch training is significantly faster and produces smaller errors than incremental training.

When training small networks, the fastest training function in Matlab is generally trainlm() function, and it is the default training function for feedforward neural network. Additionally, trainlm() performs better on nonlinear regression problems than on pattern recognition problems. The trainbfg() function, is also quite fast. Both trainlm and trainbfg require more memory and more computation time for these cases and that is why they fit better for the small network applications. When training large networks, and when training pattern recognition networks, function trainscg() and trainrp() are good choices. Their memory requirements are relatively small, and yet they are much faster than standard gradient descent algorithms.

Chapter 5 Application Studies

5.1 Data description

The dataset comes from the online resource of U.S. Department of Energy [6]. The original dataset is the commercial and residential hourly load profiles for all typical meteorological year 3 (TMY3) locations in the United States.

TMYs [41] are data sets of hourly values of solar radiation and meteorological elements for a 1-year period. TMY dataset represents typical rather than extreme conditions, so they are not suited for simulation of building system rather than designing systems to meet the worst-case conditions. The new TMY3 contains more locations than previous TMY2 so it is recommended for use instead of TMY2. In this case, the data that is extracted from 7 commercial building types in 17 Wisconsin locations is used.

5.2 Evaluation indices

The performance evaluation indices to ANN and SVR models adopted throughout this thesis are the root mean square error (RMSE), equivalent root mean square error (here marked as RMSEp) and the coefficient of variation (CV), which are defined as follows:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\hat{Y}_i - Y_i\right)^2} \tag{5.1}$$

$$RMSEp = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\hat{Y}_i - Y_i\right)^2/\left(\hat{Y}_i\right)^2} \tag{5.2}$$

$$CV = \frac{std\left(\frac{\hat{Y}_i - Y_i}{\hat{Y}_i}\right)}{mean\left(\frac{\hat{Y}_i - Y_i}{\hat{Y}_i}\right)} \tag{5.3}$$

Where n is the number of sample, $\hat{Y}_i$ is the predicted value of Y_i at the sample point i. $std()$ is the function of standard deviation, $mean()$ is the function of getting the mean value. The three Indices are used to indicate the result in different aspects.

RMSE is a measure of accuracy, which shows the differences between sample values predicted by an ANN/SVR model and the actual values. In this case, RMSE can compare forecasting errors of different models for a particular variable within a same dataset.

RMSEp is a measure of accuracy as RMSE but with the additional ability to compare the accuracy of different models for a particular variable among different datasets.

CV shows the extent of variability in relation to the mean of the population. The larger CV value is, the more validation of the data would be.

5.3 Sample collection

The summer of Wisconsin is mainly from June to August. In this study, the hourly weather data and building cooling load during the 3 months in 14 cities (Appleton, Eau Claire, Ephraim, Janesville, La Crosse, Lone Rock, Manitowac, Marshfield, Mosinee, Rice Lake, Sturgeon, Waterton, Wausau, Wittman) is the training sample and those data in Green Bay, Madison, Milwaukee is the testing sample. The 7 building type that is used in this study is Hospital, Large Hotel, Midrise Apartment, Full Service Restaurant, Large Office, Primary School and Strip Mall.

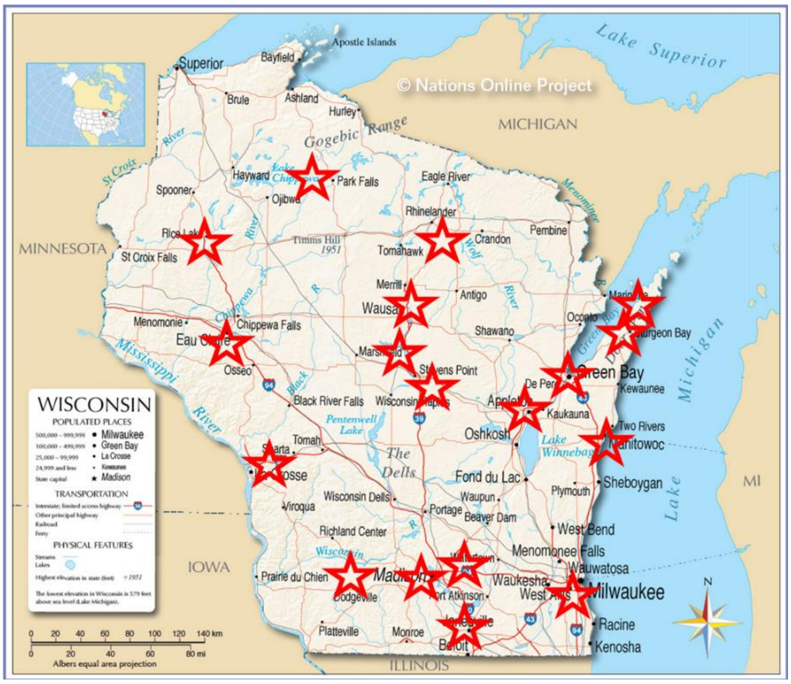


Figure 5-1 17 locations in Wisconsin map (Map graphic is from [8])

Table 3 Information for the 7 selected building type

Building Type Name	Floor Area (ft^2)	Number of Floors
Large Office	498,588	12
Strip Mall	22,500	1
Primary School	73,960	1
Full Service Restaurant	5,500	1
Hospital	241,351	5
Large Hotel	122,120	6
Midrise Apartment	33,740	4

There are 4 input parameters of ANN models and SVR models, which are dry-bulb temperature, relative humidity, solar radiation intensity and the cooling load consumption of one hour ago. The weather data is from the climate database of Wisconsin in the TMY3. The cooling load data is from the database [6]. There are 1 output for this study, which is the predicted cooling load that is calculated by the SVR and ANN models. The cooling load, which is extracted from the same database with the input data, is taken as the basic values to be compared with the predicted value. The calculated building cooling load in 14 cities is used for the output of training sample. The calculated building cooling loads in Green Bay, Madison and Milwaukee are used to be compared with the value of building cooling loads predicted by SVR model and ANN models.

5.4 Parameters setting

5.4.1 ANN

The Matlab 7.0/Neural Network Toolbox is adopted to train and develop the artificial neural networks for the building cooling load estimation.

The training function used for this exercise is Levenberg-Marquardt. According to [13], the number of neurons in a hidden layer affects the outcome of the network training. If the number of the hidden neurons is too small, then it might be not enough for correcting the modelling problem. If the number is too large, it will cost more computation time. With that being said, there is no general ways to determine the number of neurons in one hidden layer. After several experiments, we set the hidden layer as 15.

5.4.2 SVR

The Matlab 7.0/ Statistics and Machine Learning Toolbox™ is adopted to train and develop the SVM for regression in the application of building cooling load estimation.

Linear epsilon-insensitive SVM regression is implemented in the toolbox and Gaussian kernel function with an automatic kernel scale are used in this case. The set of training data includes predictor variables and observed response values. The goal is to find a function $f(x)$ that deviates from y_n by a value no greater than ε for each training point x, and at the same time the function is as flat as possible.

5.5 Prediction results and analysis

There are 7 cases based on 7 different types of buildings that are being conducted in this part. Each case will be applied ANN and SVR models. Testing data will be the summer (June, July and August) cooling load in three main cities of Wisconsin, Green Bay, Madison and Milwaukee. While the training data will be the same dataset for 14 cities in Wisconsin.

The first case is conducted with hospital cooling load. Figure 5-2 shows that both ANN and

SVR model have good accuracy by looking at the overlap ratio among ANN line, SVR line and the actual data line. Figure 5-3 presents data with Figure 5-2 but in a more quantitative way by the index called relative error (RE). Both ANN line and SVR line is near 0, which means that they both have a high accuracy. However, SVR line generally has a higher value and more undulate than ANN model, which means that ANN is more accurate than SVR in this case.

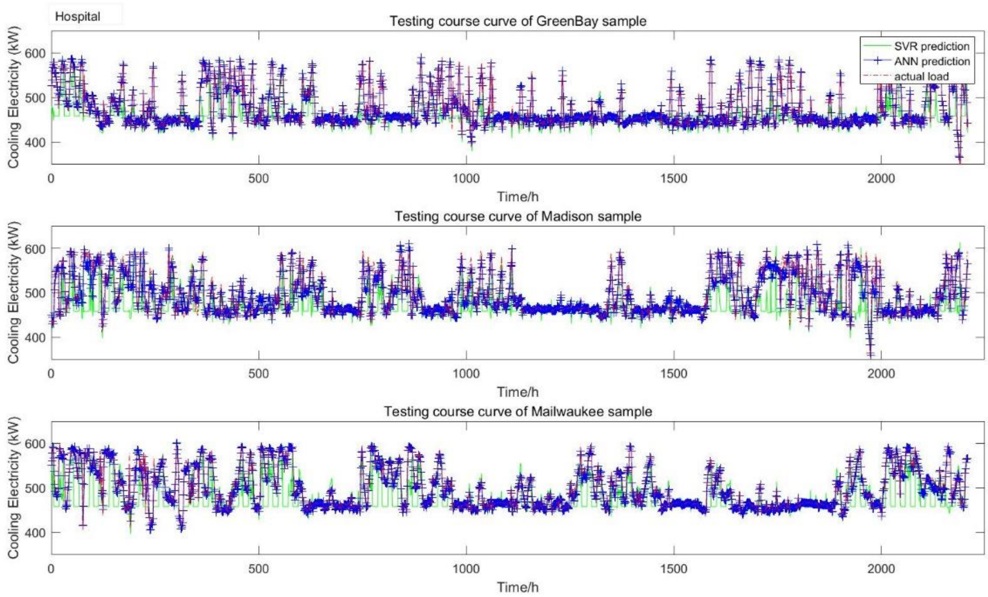


Figure 5-2 Testing curves of Green Bay, Madison and Milwaukee's sample, hospital.

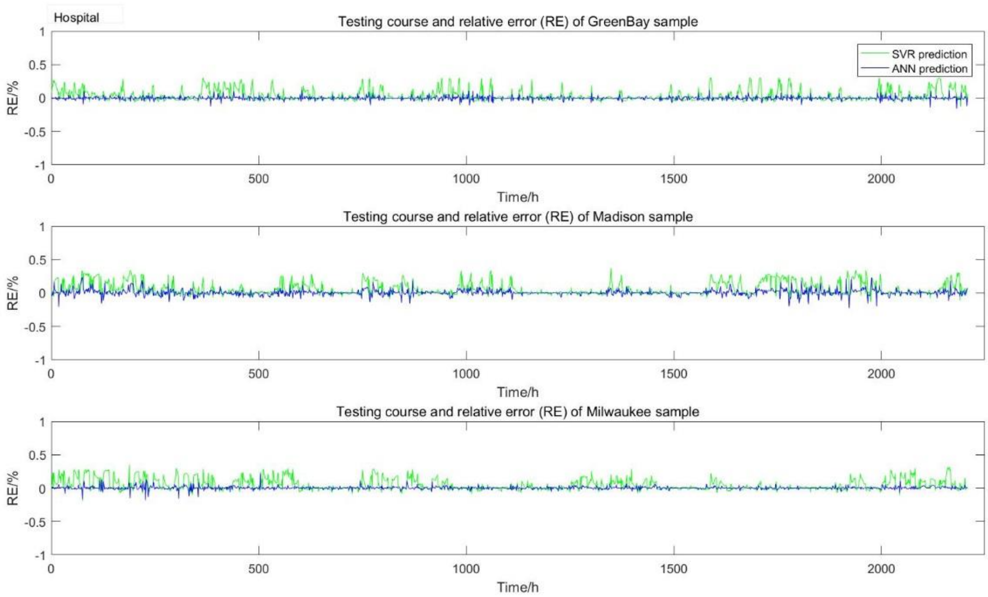


Figure 5-3 Testing relative error (RE) curves, hospital.

The second case is conducted with hotel cooling load. Figure 5-4 shows that ANN model is more accurate than SVR because the ANN prediction line follows more closely with the actual data line. However, as shown in Figure 5-5, both ANN and SVR line are more fluctuant around 0 value than the first case. It indicates that the accuracy of applying ANN and SVR to hotel data is not as good as applying those to hospital data.

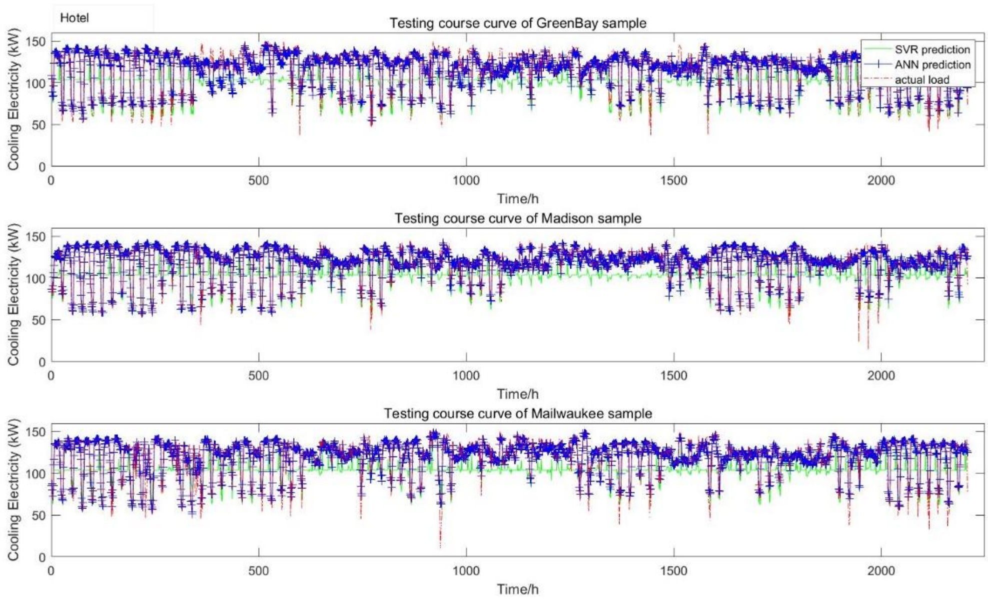


Figure 5-4 Testing curves of Green Bay, Madison and Milwaukee's sample, hotel.

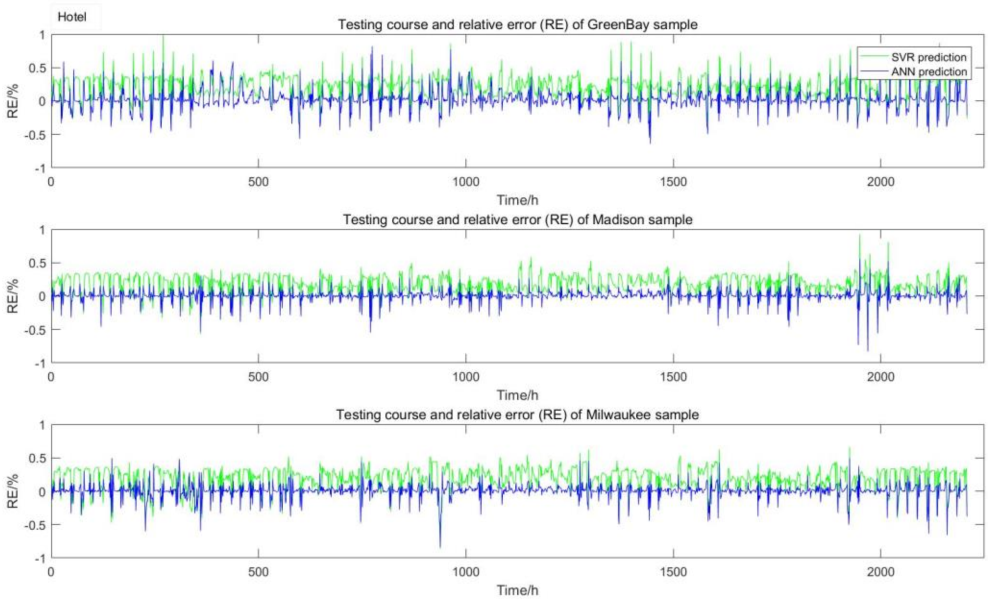


Figure 5-5 Testing relative error (RE) curves, hotel.

The third case is about large office building. This type of building has a very important feature which is that the cooling load has a clear schedule. In other words, the building manager will set a certain period of time to turn on and turn off the cooling load. Therefore, when the input variables are the external environment information and the electricity of the previous minute, the accuracy will be insufficient. Because the effect of human activities (cooling load schedule) to the electricity consumption is not being considered into the model (Schedule is not one of the input variables). As can be seen from Figure 5-4, the ANN model has better adaptability and can respond in a timely manner according to the changes in the cooling schedule of the building. In contrast, the accuracy of the SVR model is far less accurate than ANN. The SVR model predicts the maximum limit, resulting in most of the predicted value remaining near a fixed value.

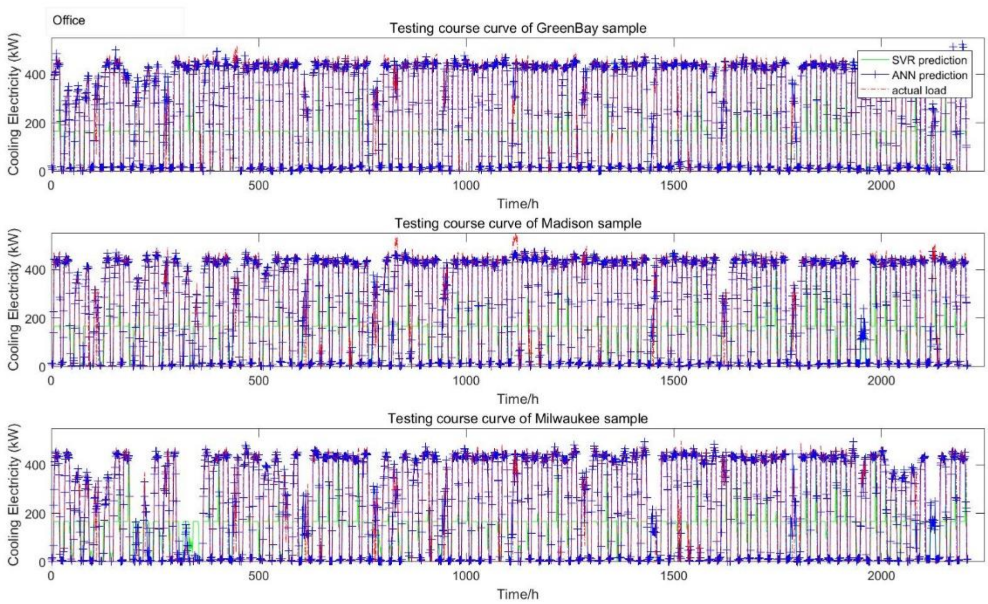


Figure 5-6 Testing curves of Green Bay, Madison and Milwaukee's sample, office.

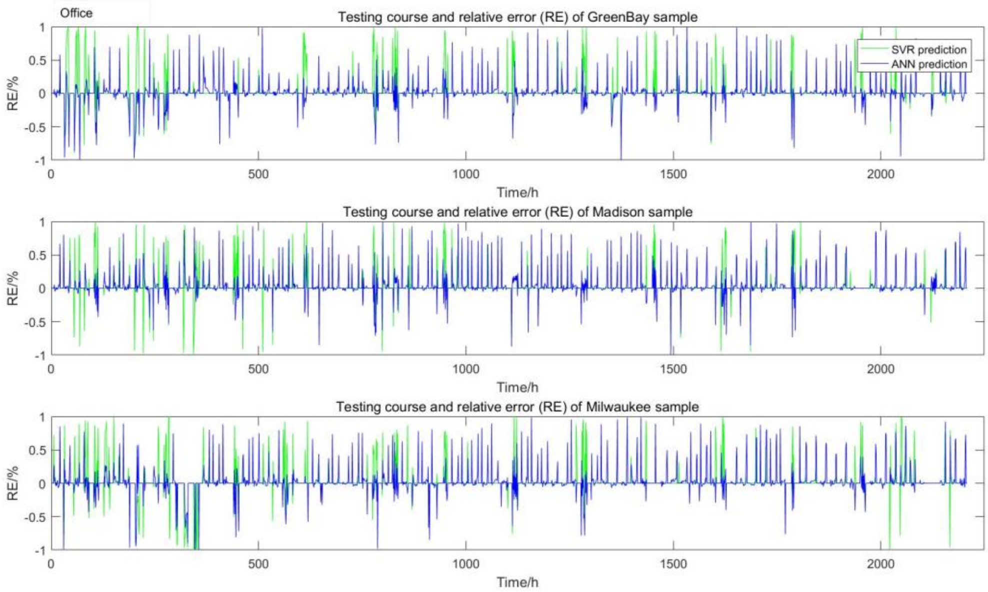


Figure 5-7 Testing relative error (RE) curves, office.

The forth case is about education building. Similar to office building type, ANN model is more accurate than SVR model. 5th case to 7th case is conducted on strip mall, apartment and restaurant building type separately and the accuracy is worse and worse. However, ANN still maintain a better performance than SVR.

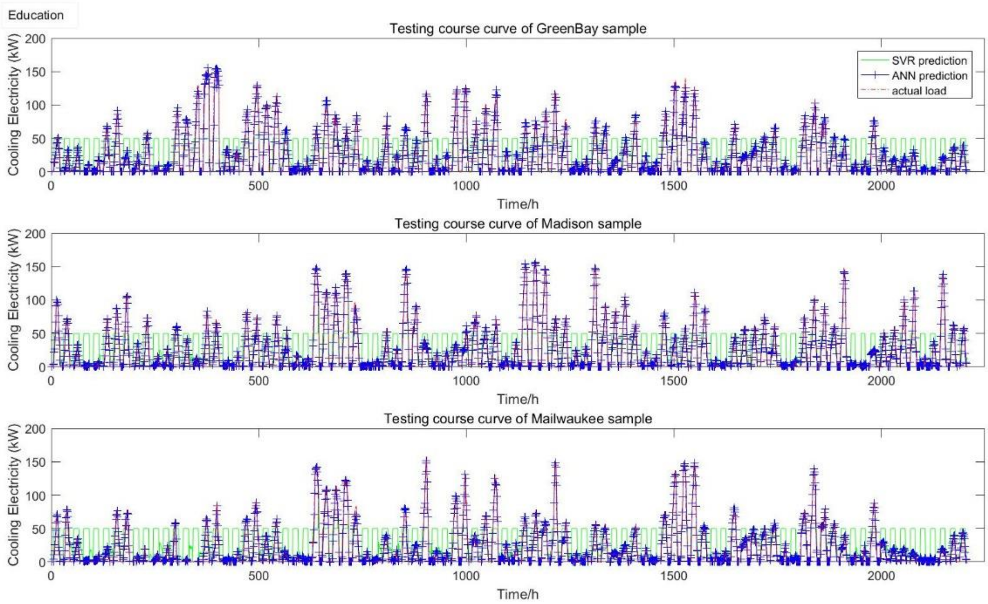


Figure 5-8 Testing curves of Green Bay, Madison and Milwaukee's sample, education.

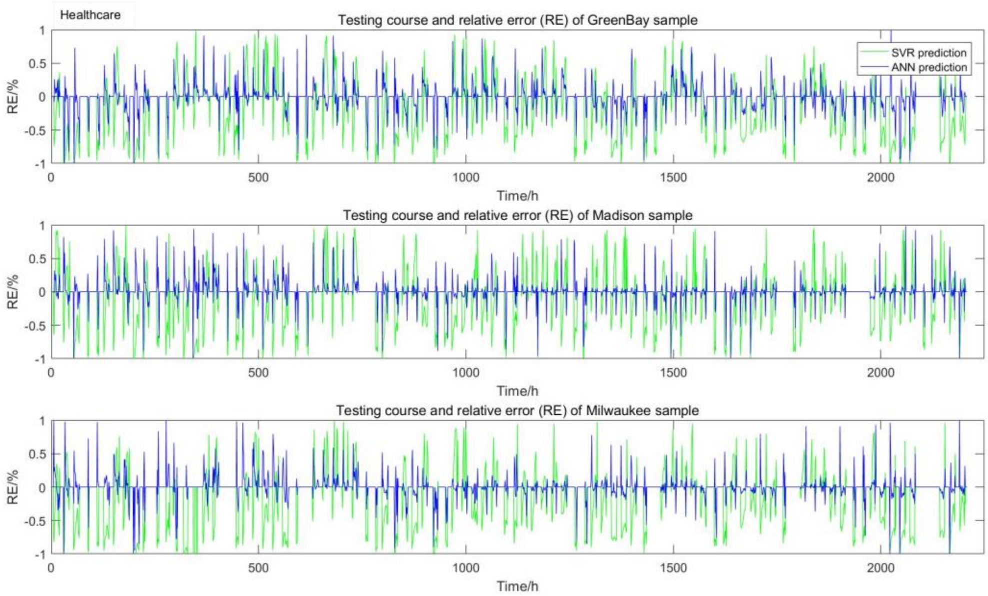


Figure 5-9 Testing relative error (RE) curves, education.

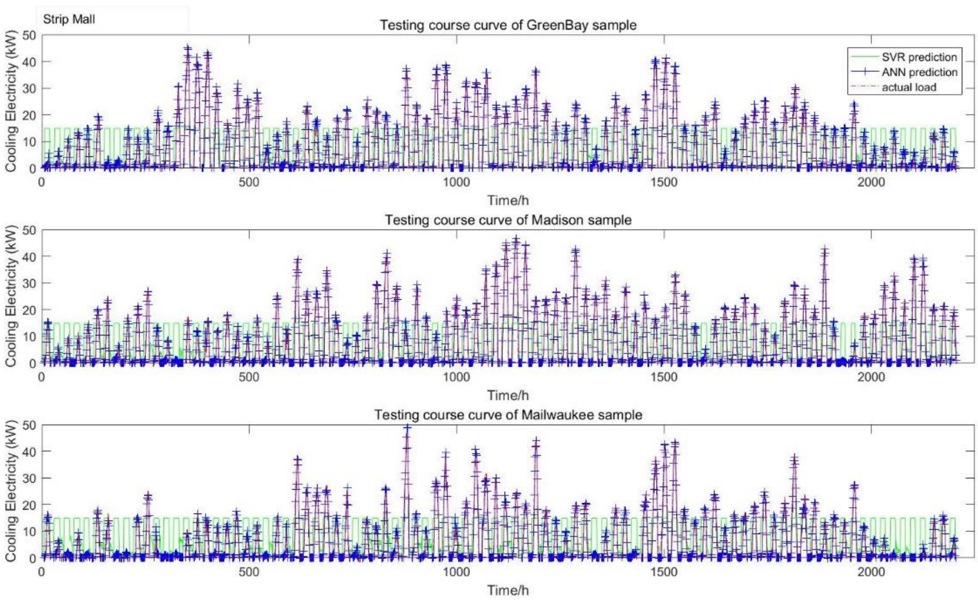


Figure 5-10 Testing curves of Green Bay, Madison and Milwaukee's sample, strip mall.

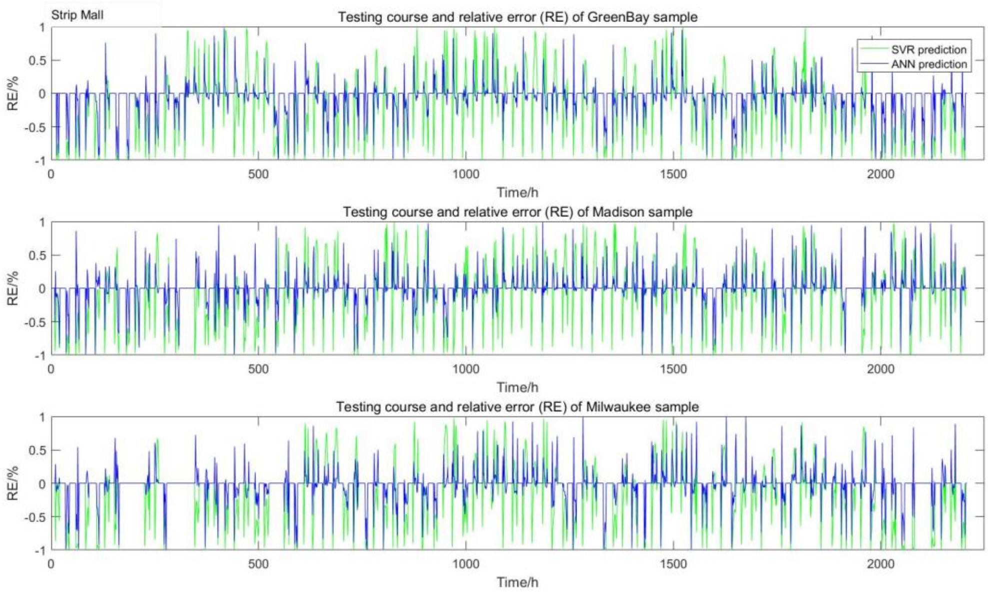


Figure 5-11 Testing relative error (RE) curves, strip mall.

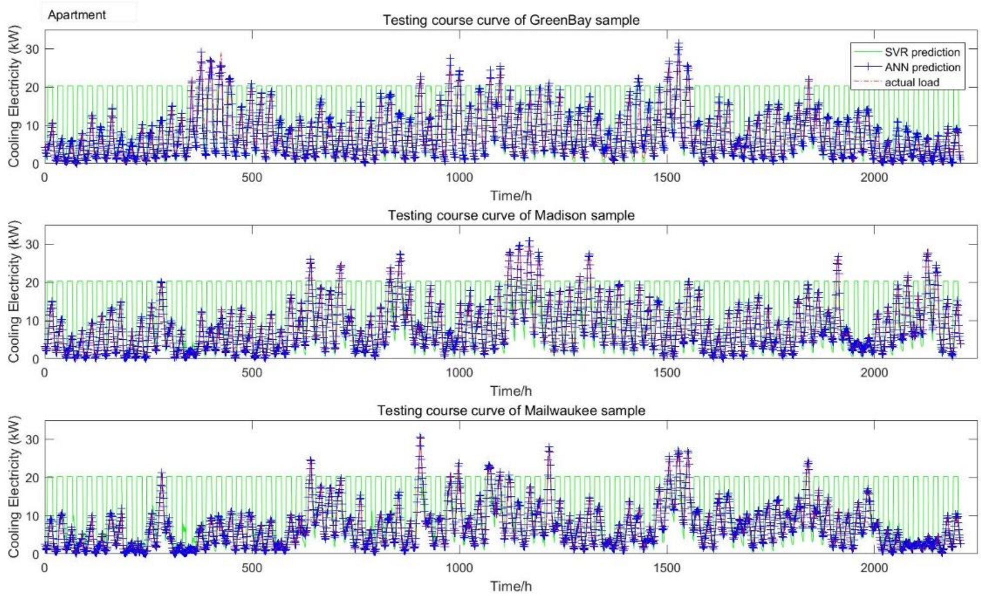


Figure 5-12 Testing curves of Green Bay, Madison and Milwaukee's sample, apartment.

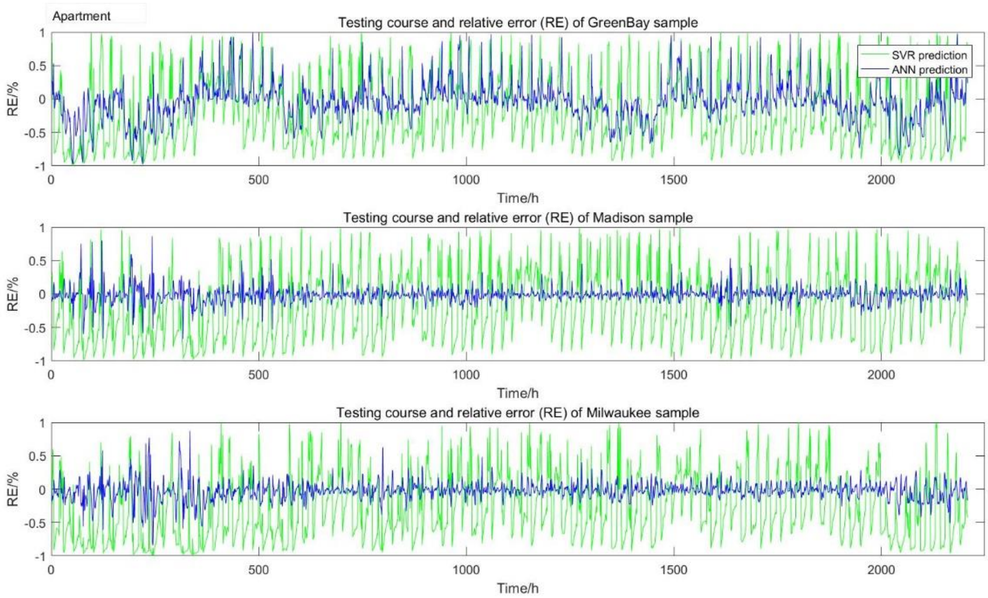


Figure 5-13 Testing relative error (RE), apartment.

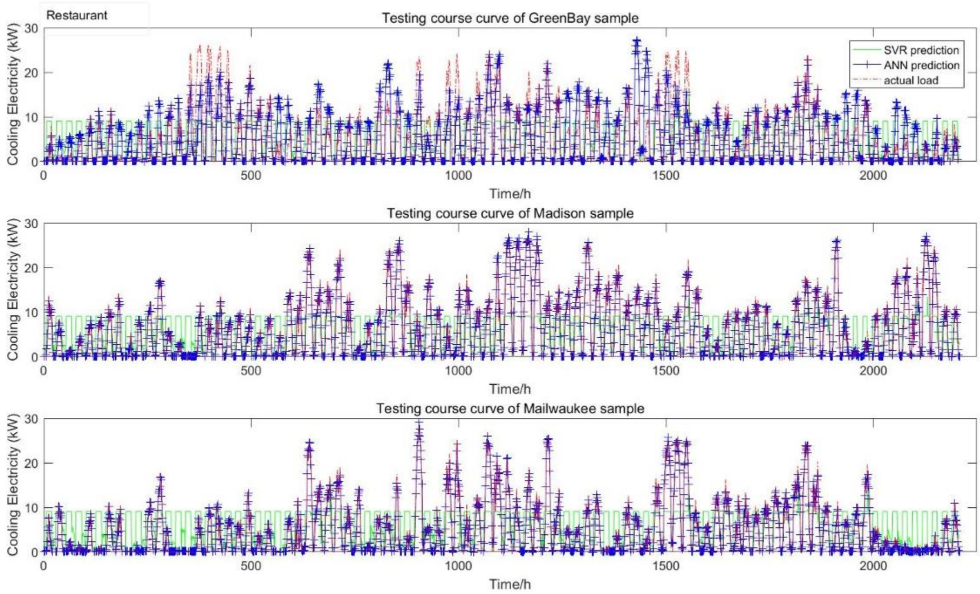


Figure 5-14 Testing curves of Green Bay, Madison and Milwaukee's sample, restaurant.

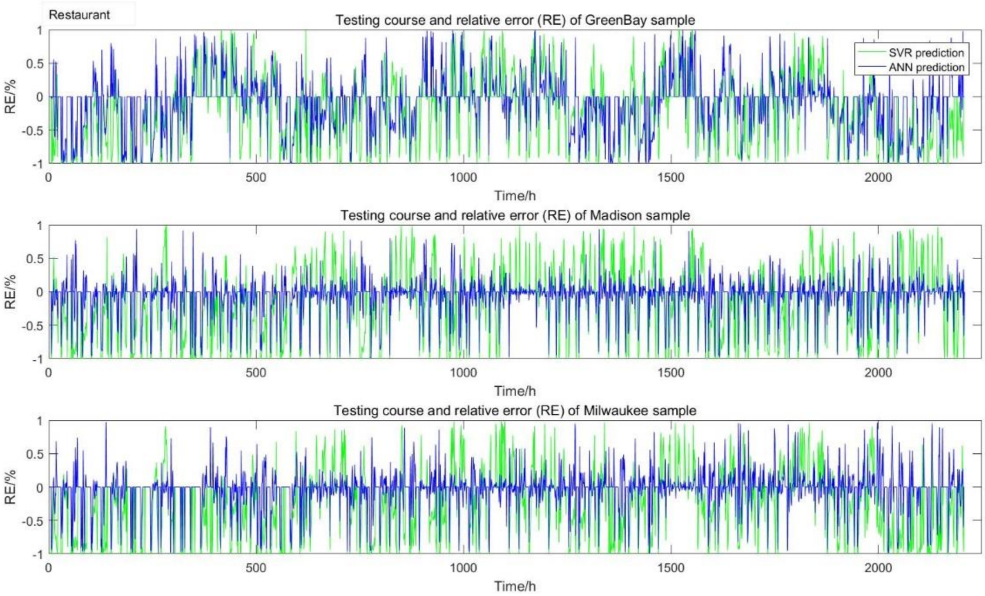


Figure 5-15 Testing relative error (RE) curve, restaurant.

Table 4 Experiment result

Healthcare		GreenBay			Madison			Milwaukee		
		RMSE	RMSEp(%)	CV	RMSE	RMSEp(%)	CV	RMSE	RMSEp(%)	CV
	ANN	11.67	0.02407	-8.81816	21.32289	0.041871	5.876622	10.29981	0.020696	5.988855
	SVM	40.16305	0.087899	2.152739	52.10187	0.1126	1.440877	45.40264	0.097616	1.602709

Hotel		GreenBay			Madison			Milwaukee		
		RMSE	RMSEp(%)	CV	RMSE	RMSEp(%)	CV	RMSE	RMSEp(%)	CV
	ANN	13.52924	0.1414	5.773506	8.788212	0.084168	14.79669	10.56752	0.102808	67.83332
	SVM	26.26	0.274784	0.733969	23.10993	0.230201	0.767647	25.54621	0.252552	0.934584

Office		GreenBay			Madison			Milwaukee		
		RMSE	RMSEp(%)	CV	RMSE	RMSEp(%)	CV	RMSE	RMSEp(%)	CV
	ANN	100.6065	0.171982	12.15384	81.28658	0.185768	4.681679	81.95287	0.190919	7.117116
	SVM	207.7987	0.189001	6.256537	204.8965	0.172316	8.903864	194.6012	0.202063	12.73752

Education		GreenBay			Madison			Milwaukee		
		RMSE	RMSEp(%)	CV	RMSE	RMSEp(%)	CV	RMSE	RMSEp(%)	CV
	ANN	8.347896	0.190368	-22.4683	6.364884	0.167977	17.0877	6.202979	0.166357	110.9441
	SVM	28.73151	2.413775	-9.47283	29.27471	1.427978	-3.62312	29.93108	2.402737	-4.53001

Mall		GreenBay			Madison			Milwaukee		
		RMSE	RMSEp(%)	CV	RMSE	RMSEp(%)	CV	RMSE	RMSEp(%)	CV
	ANN	2.257883	0.229157	-4.30951	2.215527	0.213705	94.04561	2.058177	0.219191	-10.4268
	SVM	8.536496	0.379468	-3.98034	8.915747	0.35432	-6.18143	9.049739	0.35863	-3.32444

Apartment		GreenBay			Madison			Milwaukee		
		RMSE	RMSEp(%)	CV	RMSE	RMSEp(%)	CV	RMSE	RMSEp(%)	CV
	ANN	1.792114	0.291937	-5.79195	0.541804	0.117649	-13.9381	0.586651	0.139225	-5.12061
	SVM	9.060247	0.537246	-2.47923	9.087285	0.524011	-2.33047	9.934005	0.572756	-1.50923

Restaurant		GreenBay			Madison			Milwaukee		
		RMSE	RMSEp(%)	CV	RMSE	RMSEp(%)	CV	RMSE	RMSEp(%)	CV
	ANN	4.292714	0.40025	-4.63992	1.007151	0.282703	-5.21162	1.002018	0.305763	-7.58447
	SVM	5.170799	0.507466	-2.25529	5.53259	0.495576	-2.98249	5.479701	0.502993	-2.12268

Chapter 6 Room Occupancy Detection Using Accelerometers

6.1 Project goal

The project goal is to design a system with accelerometers with the real-time detection of occupancy in a conference room for a reasonable accuracy.

To complete the goal, I break it into the following small goals:

1. When there is only one person walking in/out of the conference room, detect the walking direction (The difficulty lies in judging the direction).
2. When there are three people walking in/out of the conference room, detect every direction of the three people. (The difficulty is to identify each person's entry and exit time and the time interval between each person)
3. Use real-time data stream for detection. (The difficulty is how to synchronize real-time data to the algorithm)
4. Design a simple user interface for display.
5. Adjusting its installation to ensure accuracy. (The difficulty is that it is related to mechanical and material engineering)

Till the date of completion of this thesis, most of the above small targets have been completed, which will be described in details. However, the accuracy is still needed to be improved, especially for some special cases. So the next step requires to design with a more accurate and robust model. This content will also be mentioned later in this chapter.

6.2 Occupancy Detection Method in Industry

It is important to determine when people occupy a commercial building because the result could be used to improve building operation in terms of energy management, security and indoor air quality. However, the widely used sensors in industry are movement sensor [35] and passive infrared sensor (PIR) sensor [36] which sometimes is costly. Paper[35] describes an algorithm to simulate occupant presence, to be later used as an input for occupant behavior models within building simulation tools. In paper [36], the researchers developed and installed a network of PIR sensors in two private offices, before applying Bayesian probability theory to determine occupancy. Researchers in [37] used PIR sensor to predict and simulate occupancy in single office by examining the statistical properties of occupancy with a proposed model. Researchers in [38] described a complex sensor network with a wireless ambient-sensing system, a wired camera network, a wired carbon dioxide sensing system, and a wired indoor air quality sensing system, to solve the problem of the current sensing techniques that it is hard to determine the actual number of people in a room.

6.3 Method

6.3.1 Wavelet Transform

Wavelet transform is one of the most widely used signal processing tool today. Similar to Fourier transform, Wavelet transform breaks a signal down into its constituent parts for analysis. However, the difference is that Fourier transform breaks the signal into a series of sine waves of different frequencies, the wavelet transform breaks the signal into its "wavelets", which is the scaled and shifted versions of the "mother wavelet". Compared to the smooth sine wave with infinite length, the wavelet is densely supported with an irregular shape, which makes it an ideal tool for analyzing signals of a non-stationary nature. The reason is because the irregular shape lends them to analyzing signals with cutoff's or sharp changes, while their compactly supported nature enables temporal localization of a signals features.

For non-stationary signals generated by accelerometers in time domain, Wavelet Transform fits better than Fourier Transform. It is beneficial for researchers to acquire a correlation between the time and frequency domains of a signal. The Fourier transform, where time localized information is essentially lost in the process, however, only provides information about the frequency domain. Fourier Transform has problem in associating features in the frequency domain with their location in time. Compared to the Fourier Transform, the Wavelet Transform allows exceptional localization in both the time domain via translations of the mother wavelet, and in the scale (frequency) domain via dilations. The translation and dilation operations applied to the mother wavelet are performed to calculate the wavelet coefficients, which represent the correlation between the wavelet and a localized section of the signal. The wavelet coefficients are calculated for each wavelet segment, giving a time-scale function relating the wavelets correlation to the signal. Generally, the wavelet transform can be expressed by the following equation:

$$F(a,b) = \int_{-\infty}^{\infty} f(x)\psi^{*}{}_{(a,b)}(x)dx \tag{6.1}$$

where * is the complex conjugate symbol, ψ is some function, a is scale and b is time. The function ψ can be chosen arbitrarily provided that it obeys certain rules. [38]

6.3.2 K-means++

K-means++ is proposed by David Arthur and Sergei Vassilvitskii [39] with the aim to improving the performance of K-means clustering algorithm.

K-means clustering is an unsupervised learning, which could be used when there is unlabeled data that needed to be grouped. The aim of this algorithm is to cluster the given data into several groups based on its features and the number of groups is represented by the variable K. By iteratively assigning each data point to one of K groups based on the features that are provided, the algorithm could make data points clustered based on feature similarity. However, K-means picks the initial centers randomly. The K-means++ algorithm tries to solve this problem, by spreading the initial centers evenly.

The first cluster center of K-means++ is chosen randomly from the data, after which each subsequent cluster center is chosen from the remaining data with probability proportional to its squared distance from the data point's closest existing cluster center. To be more specific, K-means++ follows the steps below [40].

First of all, choose one data point randomly as the centers among all the data points. For each data point x, calculate the distance between x and the nearest center that has already been chosen (Here we mark the distance as D(x)). Secondly, Choose a new data point randomly as an update center, using a weighted probability distribution where the point x is chosen with probability proportional to the square of D(x). By repeating the previous steps, K centers will be chosen, as the initial centers. Then the rest of the steps will be processing the same as standard K-means clustering.

Although it takes extra time to do the initial selection in K-means++, the k-means part converges very quickly after this “pre-processing” and thus K-means actually decreses the computation time. The authors in Ref [40] tested their method with real and synthetic datasets and found out that K-means++ almost always performed at least as well as K-means in both speed and error.

6.4 Experiment and Result

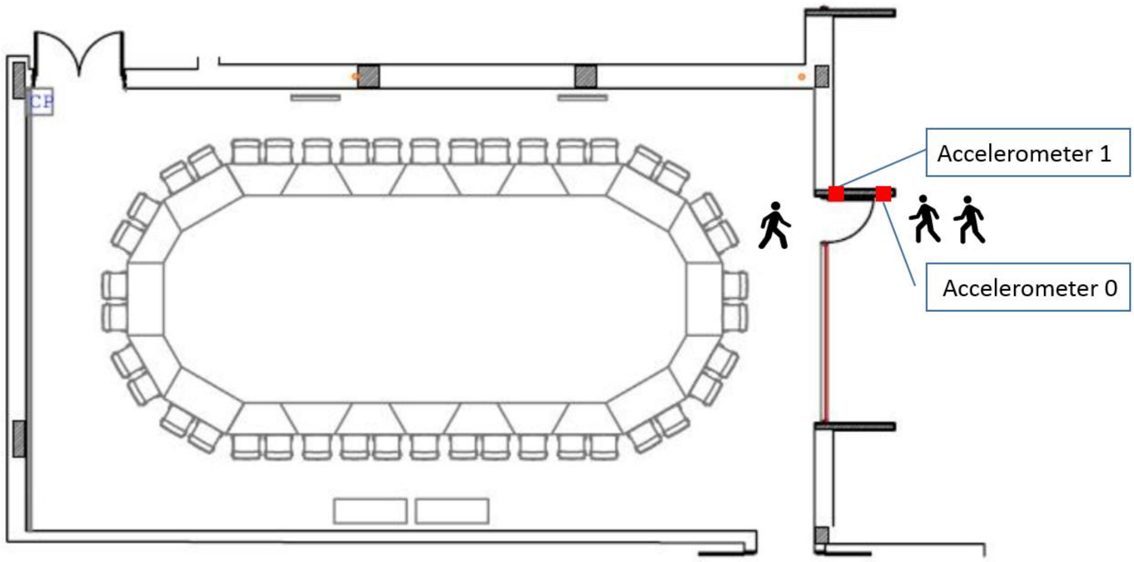


Figure 6-1 Layout

To begin the experiment, two accelerators are mounted on the carpet floor by the entrance of the conference room, as shown in Figure 6.1. The microcontroller is Arduino Mini and the two accelerometers are LIS2HH12. After collecting data from Arduino, the result is analyzed in Matlab.

I carried out the experiment with the following hypothesis: First, due to the restrictions on the width of the door, only one person allowed to pass. Secondly, due to the workplace courtesy, the person in the back will not follow the previous person too closely. There is at least 30 cm distance between them. Thirdly, the person should walk naturally without actions like stomping.

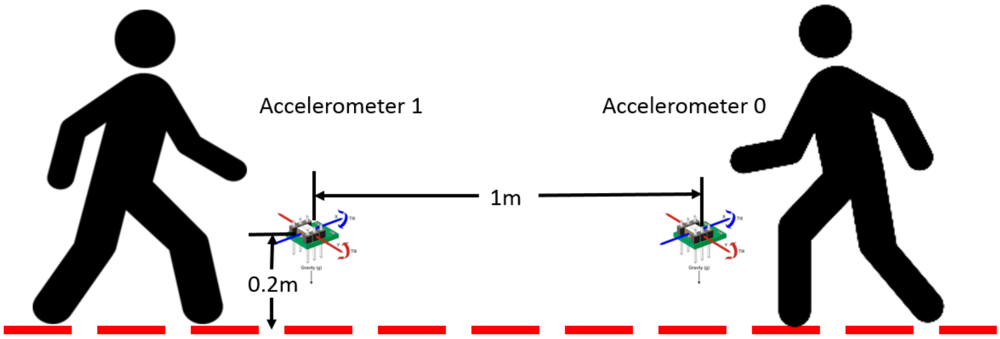


Figure 6-2 JCI office experimental environment.

Figure 6.2 shows the experimental environment I have in JCI 4^{th} floor office. The distance between the two accelerometers is 1m and the distance between each accelerometer and the walking line is roughly 0.2m.

The target walked from left to the right, passing accelerometer1 first and then accelerometer0. At the same time, the experimenter used Matlab to control the serial port and capture 2000 samples during the process. Figure 6.3 shows the original figure with noise. Although we can clearly identify peaks from the graph, some small peaks are not easily distinguishable, such as the third peak of accelerator 1. This peak is easily confused with noise and is not easy to distinguish. Therefore, we have to deal with the noise.

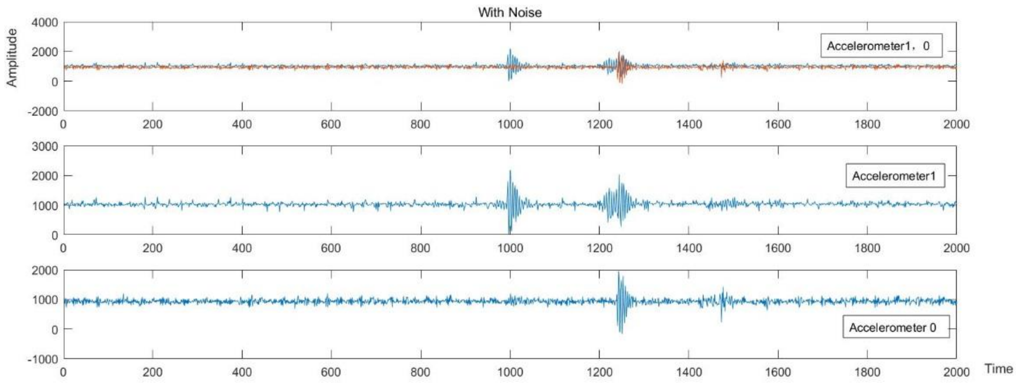


Figure 6-3 Figure with noise

There are many methods to filter the noise from a signal. Among all the methods, Wavelet transform and Fourier transform are the most popular ones. However, the Fourier has some drawbacks. For example, when transforming to the frequency domain, time information is lost. We don’t know when an event happened. In this case, it is important to know when an event happened and then remove noise from time series. So Wavelet transform is more suitable for this case. Figure 6.4 is the result after filtering the noise using Wavelet transform.

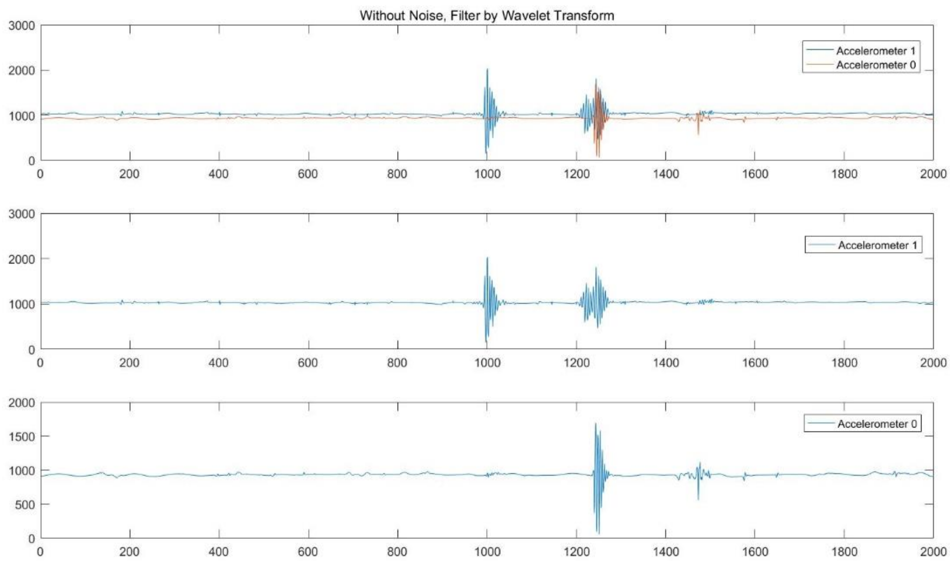


Figure 6-4 Figure without noise

From Figure6.4 we can see that the base values of the two accelerometers are slightly different. Accelerometer 1 is around 1000 and accelerometer 0 is slightly below 1000. In order to compare the differences between the two on the same basis. I have processed the filtered data as following procedures.

Take Accelerometer 1 as an example, the data after being filtered by Wavelet transform could be expressed as (6.2).

$$f(x_i) = y_i, i = 1, 2, ..., 2000. \tag{6.2}$$

Where x_i is the time at sample i, y_i is the amplitude at sample i. Apply (6.3) to each data point, and plot the data as Figure 6.5. The base value is tune to 0.

$$f_1(x_i) = f(x_{i+1}) - f(x_i), i = 1, 2, ..., 1999 \tag{6.3}$$

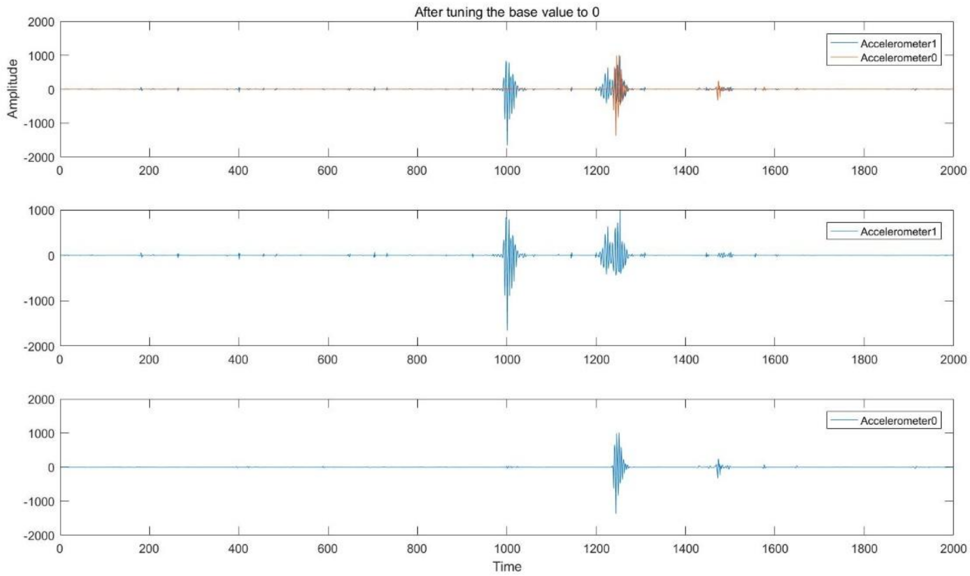


Figure 6-5 Figure of tuning base value to 0

Now it is time to find a way to detect the walking distance. By using the command in Matlab as showed in Figure 6.6, it is easy to find the peak in each line.

```
[pks1,locs1] = findpeaks(abs(y1_dev),x,'MinPeakHeight',20,'MinPeakDistance',100)
[pks2,locs2] = findpeaks(abs(y2_dev),x,'MinPeakHeight',20,'MinPeakDistance',100)
```

Figure 6-6 Matlab command to find the peak

Assuming that there are m peaks in accelerometer1 and n peaks in accelerometer0. After conducting the command, the result is showed as (6.4), where $p_m{}^1$ means the m peak value of accelerometer 1 and $q_m{}^1$ means the time when the m peak happens.

$$\begin{aligned} &Peak1:(q_1{}^1, p_1{}^1),(q_2{}^1, p_2{}^1),...(q_m{}^1, p_m{}^1) \\ &Peak0:(q_1{}^0, p_1{}^0),(q_2{}^0, p_2{}^0),...(q_n{}^0, p_n{}^0) \end{aligned} \tag{6.4}$$

Apply (6.5) to (6.4), find the locations of maximum and minimum peak in both accelerometers.

$$\begin{aligned} &(q_{\max}{}^1, p_{\max}{}^1) = FindMax(p_1{}^1, p_2{}^1,..., p_m{}^1) \\ &(q_{\max}{}^0, p_{\max}{}^0) = FindMax(p_1{}^0, p_2{}^0,..., p_m{}^0) \\ &(q_{\min}{}^1, p_{\min}{}^1) = FindMin(p_1{}^1, p_2{}^1,..., p_m{}^1) \\ &(q_{\min}{}^0, p_{\min}{}^0) = FindMin(p_1{}^0, p_2{}^0,..., p_m{}^0) \end{aligned} \tag{6.5}$$

Pseudo-code to detect the walking direction is showed in Figure 6.7 and the result is showed in Figure 6.8. This method has high accuracy.

$$\boxed{\begin{aligned} &if\,(q_{\max}{}^1 > q_{\max}{}^0)\,\&\&(q_{\min}{}^1 < q_{\min}{}^0) \\ &print('right\ to\ the\ left') \\ &elseif\,(q_{\max}{}^1 < q_{\max}{}^0)\,\&\&(q_{\min}{}^1 > q_{\min}{}^0) \\ &print('left\ to\ the\ right') \\ &else \\ &print('unknown') \\ &end \end{aligned}}$$

Figure 6-7 pseudo-code

```
ans =

left to the right

fx >>
```

Figure 6-8 Matlab Result

Up till now, the first small goal that mentioned in 6.1 section is complete. The next step is how to detect accurately when there are up to 3 experimenters. The original figure shows in Figure 6.9.

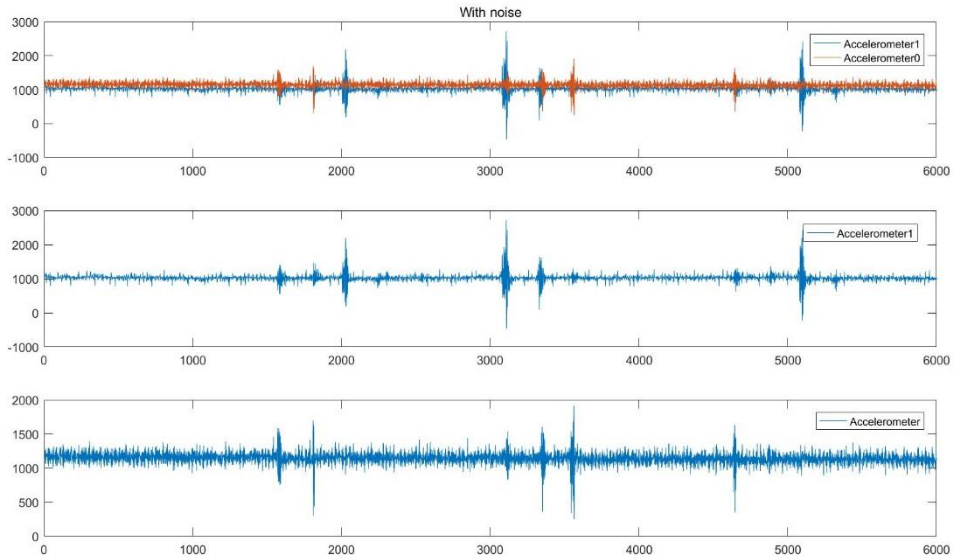


Figure 6-9 Figure with noise, 3 experimenters

After processing the data with the same method, the figure shows as Figure 6-10. From Figure 6-10 we could tell that there are 3 experimenters in total as showed in Figure 6-11.

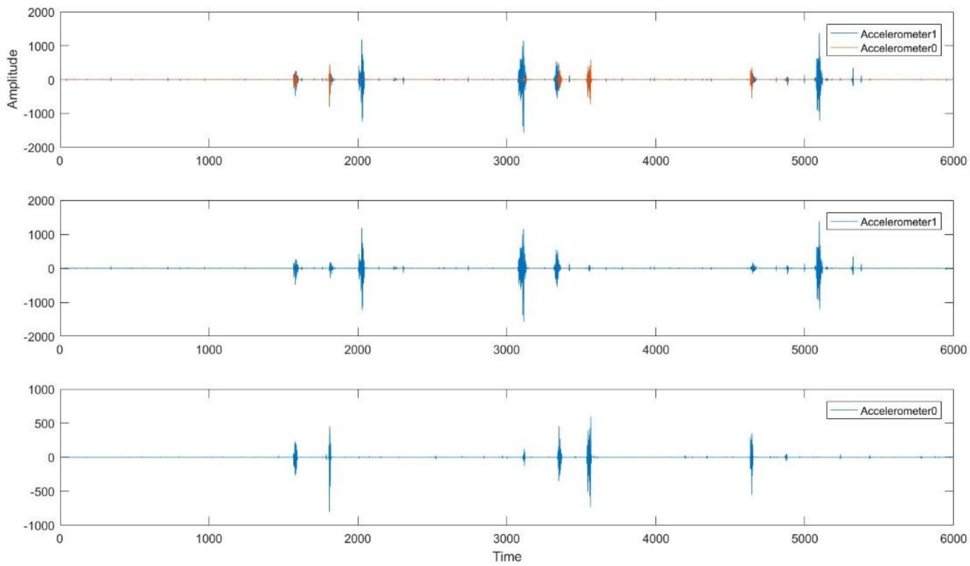


Figure 6-10 Figure of tuning base value to 0, 3 experimenters

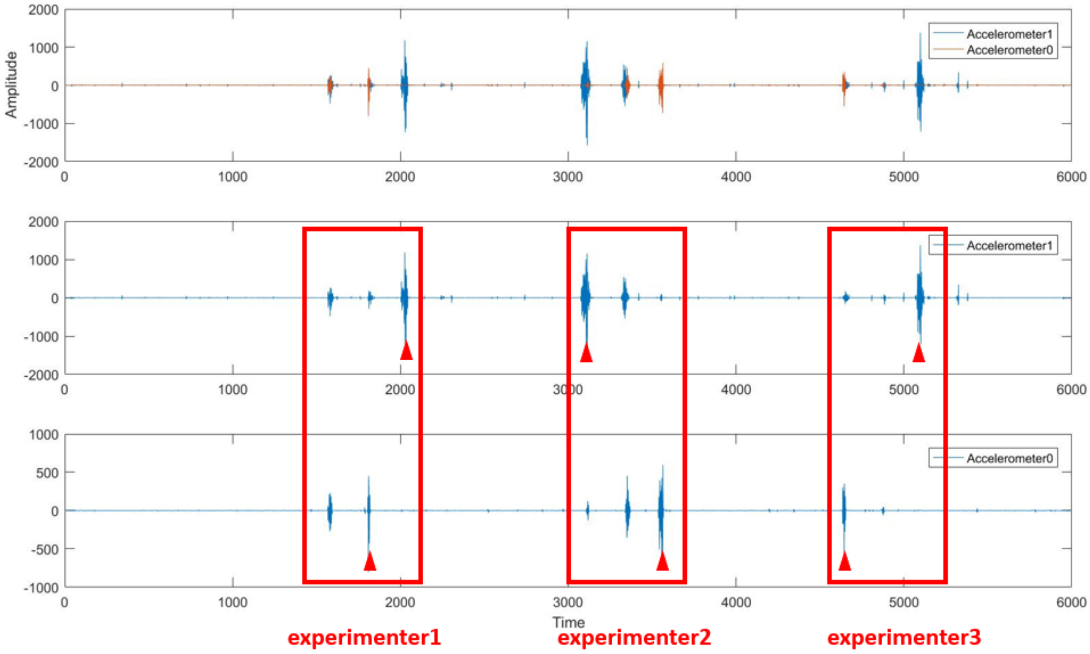


Figure 6-11 Figure of tuning base value to 0, 3 experimenters with explanation

If we plot the data into a 3D environment, z axis is the timeline, x and y is the data of accelerometer1 and accelerometer0 separately, the figure shows as Figure 6-12. There are 3 layers in the figure, every layer represents a person walking by the two accelerometers. By comparing the time that peak value occurs in both x and y axis, we could detect the direction of each person.

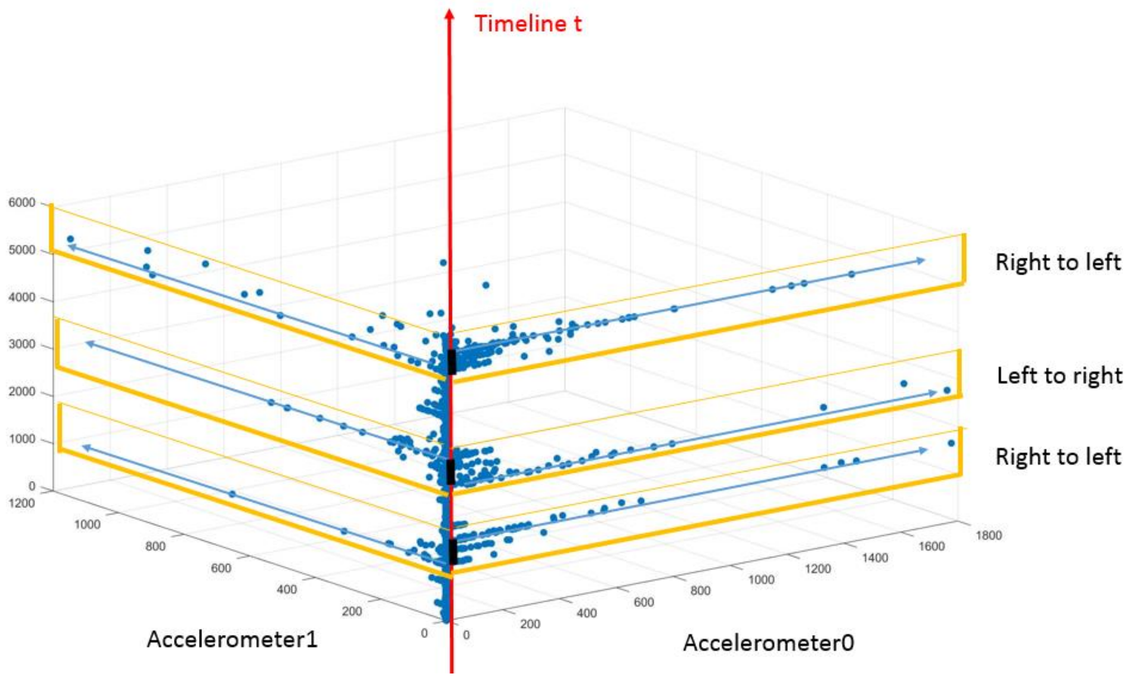


Figure 6-12 3D figure

What should be accentuated is that the pattern when a person is walking through the two accelerometers is very clear from Figure 6-11. This pattern has 3 classes, which are ‘Right to left’, ’ Left to right’ and ’No one passing by’. When ‘Right to left’ happens, the peak in Accelerometer1 happens first and then follows with the peak in Accelerometer0. When ‘Left to right’ happens, the peak in Accelerometer0 happens first and then follows with the peak in Accelerometer1. When ‘No one passing by’ happens, no peak is detected in both accelerometers. The method to detect the three patterns is similar with the method to detect 1 person. However, some values that are used for detection might be modified if the experimental environment changes.

After classifying the patterns, it is important to capture the walking time interval accurately and apply the pattern recognition method to each interval. Now it is the time to use the clusting algorithm, called K-means++. K-means++ could separate data into certain number of groups based on its feature. In this case, I separate the 6000 samples into 3 groups and the K-means algorithm will help to distinguish the 3 groups by its feature. With that being said, the maximum number of people that we are able to detect in this case is 3 and the minmum is 0.

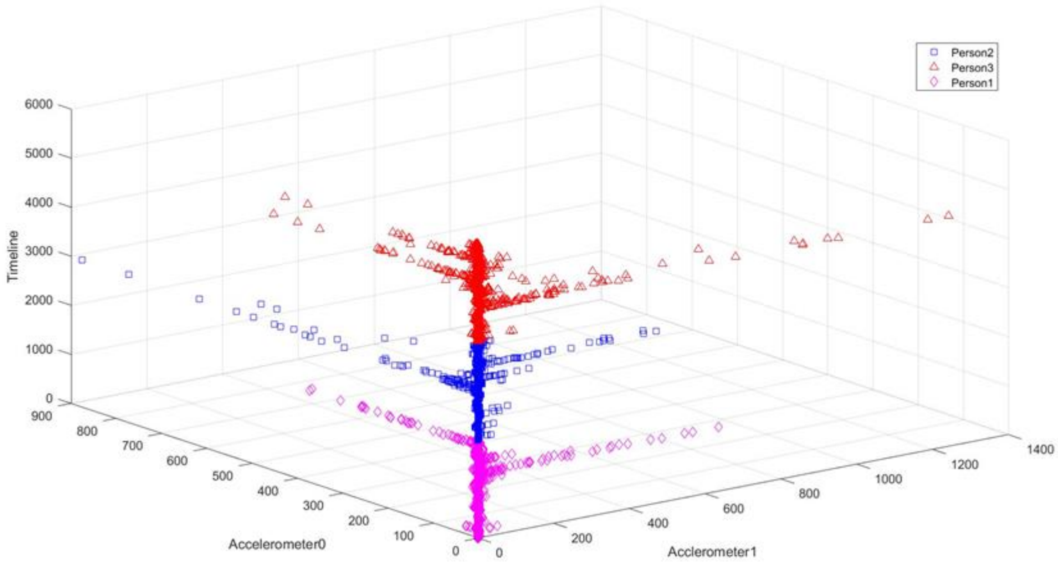


Figure 6-13 3D figure after clustering with k-mean++

After applying k-means++ into the dataset with the 3 group settings, the result is showed in Figure 6-13. Clearly to see that 3 people's movement is detected and clustered. An UI for this case is designed in Figure 6-14.

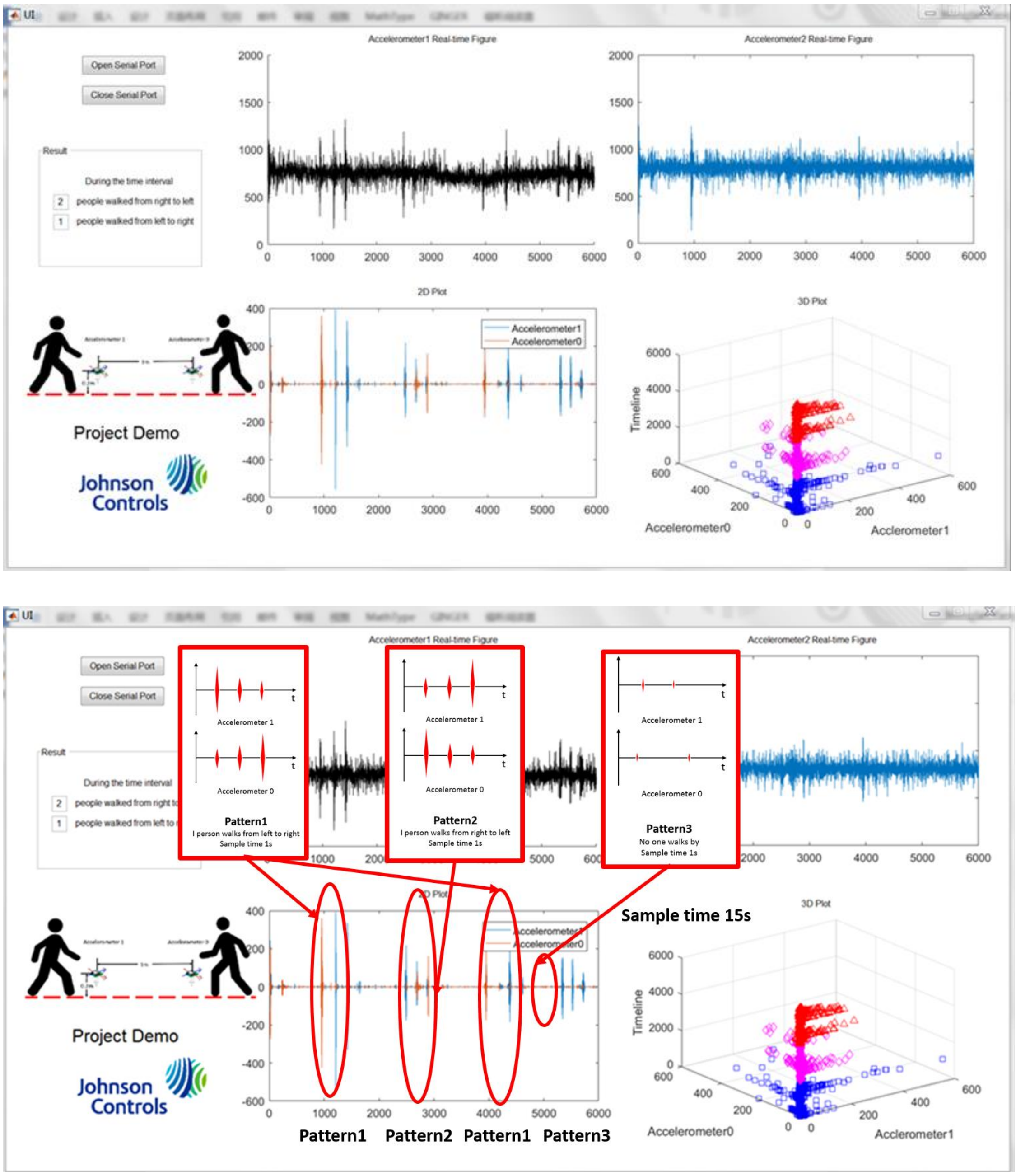


Figure 6-14 User Interface

Note: Some follow-up experiments to expand the scope of this project is needed in the future, depending on the performance of this experiment. For example, how to successfully identify two people walking at the same time, how to detect some special case like people standing in the detection area for a chat. These experiments require higher resolution accelerator, more accelerators probably, further study of the mounting locations and more advanced recognition algorithms.

The method that using accelerometers to detect occupancy shows to be promising and is worth to dig deeper for higher accuracy. In the future, more patterns need to be involved and recognized in the detection system. In table 5, it lists 6 patterns.

Table 5 6 Patterns of 1 or 2 people walking

Pattern	Description
Pattern1	I person walks from left to right
Pattern2	I person walks from right to left
Pattern3	No one walks by
Pattern4	2 people walks closely to the right
Pattern5	2 people walks closely to the left
Pattern6	2 people walks toward each other

The ideal pattern with 1 experimenter and 2 experimenters are showed in Figure 6-15 and Figure 6-16 separately. And the difficulty lies in the pattern recognition between pattern1 and pattern4, and between pattern2 and pattern5, because their shape look very similar.

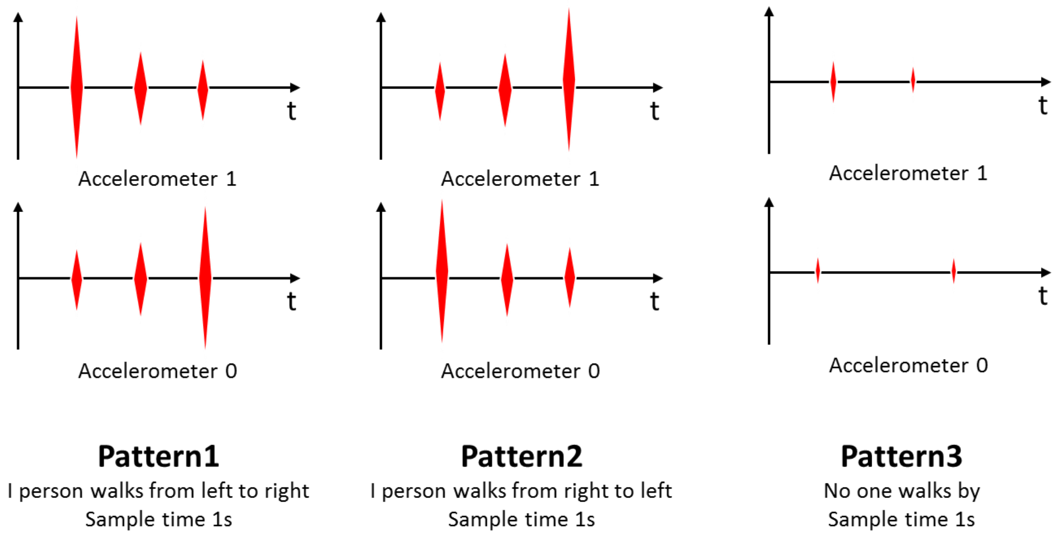


Figure 6-15 Ideal pattern with 1 experimenters

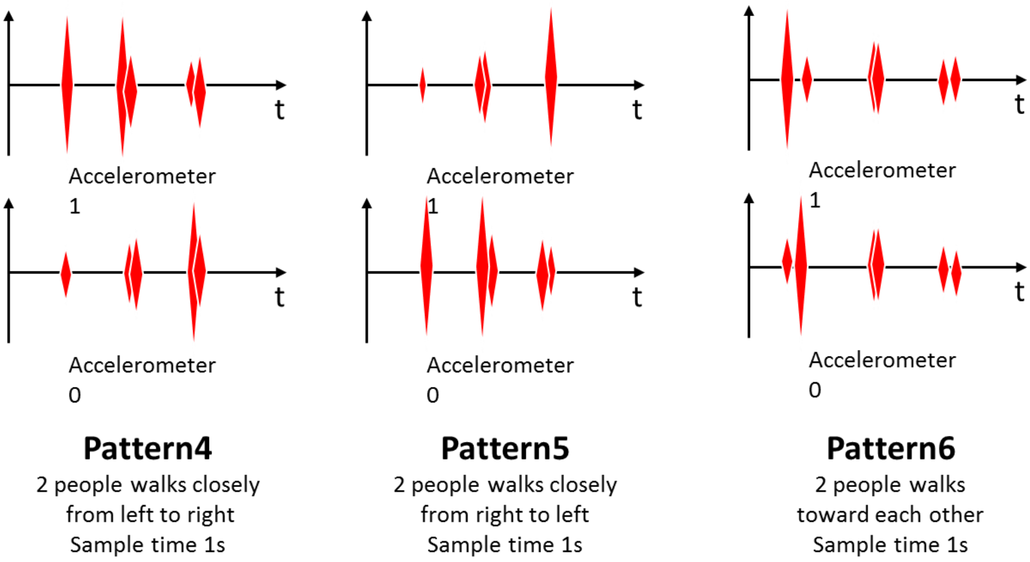


Figure 6-16 Ideal pattern with 2 experimenters

At the same time, a sensor with higher resolution and less noise should also be tested, in order to find an alternative solution to solve the problem. In figure 6.17, it shows the connection of digital and analog accelerometers in the experiment later on. Because of security reason, the details will not be presented in this thesis.

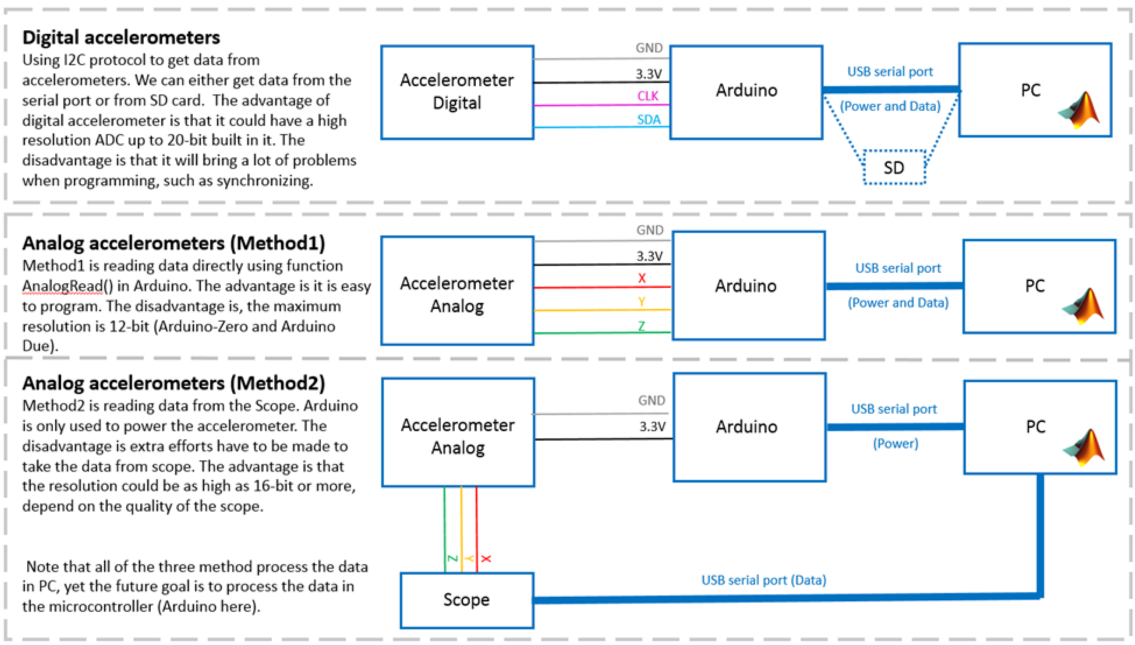


Figure 6-17 Digital accelerometer and analog accelerometer

Chapter 7 Conclusions and Future Work

In this thesis, there are two research areas that are being explored. Cooling load prediction of different building types is studied in the first part, and the second part is to explore the method of using accelerometers to detect occupancy in a room.

In the first part, the pattern of cooling load consumption in different building types is being analyzed. Two algorithms, ANN and SVR are applied to the cooling load data for short-term prediction. A performance comparison is made for various building types and several conclusions are obtained. Firstly, it is necessary to apply different models to different building types if high accuracy is required. Secondly, compared to SVR, ANN is more accurate in all the building types. Lastly, the difference of the accuracy depends on the building features. In hospital buildings, SVR and ANN both show high accuracy level, but in restaurant buildings, they are both underperforming.

In the second part, the method of using accelerometers to detect occupancy is explored. Walking direction pattern is classified and clustered, and the result shows decent accuracy in limited conditions. Wavelet transform is used for signal processing and K-means++ is used for pattern recognition. The results show that this method is promising and is worth looking deeper into in the future for use in efficient building management.

Future work in cooling load prediction could be focused on the following aspects:

• Selection of parameters in SVR and ANN should be further studied. In this thesis, the selection of parameters is based on experience. In the future, more work should be done to deeply understand the two algorithms to optimize the performance with reasonable parameter selection.

• SVR algorithm and other novel algorithms could be tested in order to provide more options for electric utilities. In this thesis, I choose the most popular prediction algorithm not basing my choice on their best prediction accuracy. In the future, more artificial regression algorithms should be tested in different building types to provide more options for electric utilities to optimize performance.

• More building types could be involved in the future work, to provide a wider range of cooling load consumption profiles for electric utilities. In this thesis, there are only 7 commercial building types being listed. More commercial building types, for example, warehouse, museum and residential buildings could also be tested for the next step.

Future work in occupancy detection could be focused on the following aspects:

• More patterns should be considered to enhance the robustness of the performance in this detection system. In this thesis, there are only 3 patterns being considered. When more complicated scenarios happen, the prediction results could become inadequate.

• Real-time data processing should be explored. To detect the occupancy in real time, it is important to grasp the real-time data processing technology. The selection of sample time as well as the length of update time is essential for this case.

• The alternative accelerometer board should be tested to get data with better quality. The quality of the vibration data strongly influences the prediction accuracy. An accelerometer with higher resolution and lower noise will probably bring more benefit in terms of prediction rather than the current one.

References

[1]http://www.iea.org/statistics/statisticssearch/report/?year=2014&country=USA&product=ElectricityandHeat

[2] Pérez-Lombard L, Ortiz J, Pout C. A review on buildings energy consumption information[J]. Energy and buildings, 2008, 40(3): 394-398.

[3] https://www.eia.gov

[4] Energy Information Administration, International Energy Outlook 2016, U.S. Department of Energy, June 2016.

[5]https://portfoliomanager.energystar.gov/pdf/reference/US%20National%20Median%20Table.pdf

[6]https://www.energystar.gov/buildings/tools-and-resources/portfolio-manager-technical-reference-energy-star-score

[7]http://en.openei.org/doe-opendata/dataset/commercial-and-residential-hourly-load-profiles-for-all-tmy3-locations-in-the-united-states

[8] http://www.nationsonline.org/oneworld/map/USA/wisconsin_map.htm

[9] Vapnik, V. The Nature of Statistical Learning Theory. Springer, New York, 1995.GB/T 7714

[10] Fan, R.E. , P.H. Chen, and C.J. Lin. A Study on SMO-Type Decomposition Methods for Support Vector Machines. IEEE Transactions on Neural Networks, 17:893–908, 2006.

[11] Fan, R.E. , P.H. Chen, and C.J. Lin. Working Set Selection Using Second Order Information for Training Support Vector Machines. The Journal of machine Learning Research, 6:1871–1918, 2005.

[12] Platt, J. Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines. Technical Report MSR-TR-98–14, 1999.

[13] Yasar I, Akif K, Cem P. Performance prediction for non-adiabatic capillary tube suction line heat exchanger: an artificial neural network approach. Energy Convers Manage 2005;46(2):223-10.

[14] Yokoyama R, Wakui T, Satake R. Prediction of energy demands using neural network with model identification by global optimization. Energy Conversion and Management 2009;50(2):319–27.

[15] Kreider JF, Claridge DE, Curtiss P, Dodier R, Haberl JS, Krarti M. Building energy use prediction and system identification using recurrent neural networks. Journal of Solar Energy Engineering 1995;117(3):161–6.

[16] Ben-Nakhi AE, Mahmoud MA. Cooling load prediction for buildings using general regression neural networks. Energy Conversion and Management 2004;45(13–14):2127–41.

[17] Yan C-w, Yao J. Application of ANN for the prediction of building energy consumption at different climate zones with HDD and CDD. In: Proceedings of the 2nd international conference on future computer and communication, vol. 3. 2010. p. 286–9.

[18] Joint Center for Energy Management (JCEM). Final report: Artificial neural networks applied to loan STAR data. Tech. Rep. TR/92/15; 1992.

[19] Javeed Nizami S, Al-Garni AZ. Forecasting electric energy consumption using neural networks. Energy Policy 1995;23(12):1097–104.

[20] González PA, Zamarreno JM. Prediction of hourly energy consumption in buildings based on a feedback artificial neural network. Energy and Buildings 2005;37(6):595–601.

[21] Wong SL, Wan KKW, Lam TNT. Artificial neural networks for energy analysis of office buildings with daylighting. Applied Energy 2010;87(2):551–7.

[22] Hou Z, Lian Z, Yao Y, Yuan X. Cooling-load prediction by the combination of rough set theory and an artificial neural-network based on data-fusion technique. Applied Energy 2006;83(9):1033–46.

[23] Ben-Nakhi AE, Mahmoud MA. Energy conservation in buildings through efficient A/C control using neural networks. Applied Energy 2002;73(1):5–23.

[24] Yalcintas M, Akkurt S. Artificial neural networks applications in building energy predictions and a case study for tropical climates. International Journal of Energy Research 2005;29(10):891–901.

[25] Gouda MM, Danaher S, Underwood CP. Application of an artificial neural network for modelling the thermal dynamics of a building's space and its heating system. Mathematical and Computer Modelling of Dynamical Systems: Methods, Tools and Applications in Engineering and Related Sciences 2002;8(3):333–44.

[26] Boser, B. E.; Guyon, I. M.; Vapnik, V. N. (1992). "A training algorithm for optimal margin classifiers". Proceedings of the fifth annual workshop on Computational learning theory – COLT '92. p. 144.

[27] Dong B, Cao C, Lee SE. Applying support vector machines to predict building energy

consumption in tropical region. Energy and Buildings 2005;37(5):545–53.

[28] Lai F, Magoulès F, Lherminier F. Vapnik's learning theory applied to energy consumption forecasts in residential buildings. International Journal of Computer Mathematics 2008;85(10):1563–88.

[29] Li Q, Meng QL, Cai JJ, Hiroshi Y, Akashi M. Applying support vector machine to predict hourly cooling load in the building. Applied Energy 2009;86(10):2249–56.

[30] Hou Z, Lian Z. An application of support vector machines in cooling load prediction. In: Proceedings of international workshop on intelligent systems and applications. 2009. p. 1–4.

[31] Li Q, Ren P, Meng Q. Prediction model of annual energy consumption of residential buildings. In: Proceedings of 2010 international conference on advances in energy engineering. 2010. p. 223–6.

[32] Lv J, Li X, Ding L, Jiang L. Applying principal component analysis and weighted support vector machine in building cooling load forecasting. In: Proceedings of 2010 international conference on computer and communication technologies in agriculture engineering. 2010. p. 434–7.

[33] LiX, Ding L, Lv J,Xu G, Li J.Anovel hybrid approach ofKPCAand SVM for building cooling load prediction. In: Proceedings of the third international conference on knowledge discovery and data mining. 2010. p. 522–6.

[34] Bing Dong, Burton Andrews, Khee Poh Lam. An information technology enabled sustainability test-bed (ITEST) for occupancy detection through an environmental sensing network. In: Energy and Buildings2010; 42(7), p. 1038–46

[35] J. Page , D. Robinson, N. Morel, J.-L. Scartezzini. A generalised stochastic model for the simulation of occupant presence. In: Energy and Buildings 2008; 40(2), p. 83-98.

[36] Robert H. Dodiera, Gregor P. Henzeb, Dale K. Tillerb, Xin Guob. Building occupancy detection through sensor belief networks. In: Energy and Buildings 2006; 38(9), p. 1033-43

[37] Danni Wanga,, Clifford C. Federspiela, Francis Rubinsteinb. Modeling occupancy in single person offices. In: Energy and Buildings 2005; 37(2), p. 121-6

[38] http://disp.ee.ntu.edu.tw/tutorial/WaveletTutorial.pdf

[39] Arthur, D.; Vassilvitskii, S. (2007). "k-means++: the advantages of careful seeding" (PDF).

Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms. Society for Industrial and Applied Mathematics Philadelphia, PA, USA. pp. 1027–1035.

[40] Kanungo, T.; Mount, D.; Netanyahu, N.; Piatko, C.; Silverman, R.; Wu, A. (2004), "A Local Search Approximation Algorithm for k-Means Clustering" (PDF), Computational Geometry: Theory and Applications.

[41] http://rredc.nrel.gov/solar/old_data/nsrdb/1991-2005/tmy3/