Predicting the city foot traffic with pedestrian sensor data

Xianjing Wang
RMIT University
Melbourne, Victoria
xianjing.wang@rmit.edu.au

Jonathan Liono
RMIT University
Melbourne, Victoria
jonathan.liono@rmit.edu.au

Will McIntosh
City of Melbourne
Melbourne, Victoria
Will.Mcintosh@melbourne.vic.gov.au

Flora D. Salim
RMIT University
Melbourne, Victoria
flora.salim@rmit.edu.au

ABSTRACT

In this paper, we focus on developing a model and system for predicting the city foot traffic. We utilise historical records of pedestrian counts captured with thermal and laser-based sensors installed at multiple locations throughout the city. A robust prediction system is proposed to cope with various temporal foot traffic patterns. The empirical evaluation of our experiment shows that the proposed ARIMA model is effective in modelling both weekdays and weekend patterns, outperforming other state-of-art models for short-term prediction of pedestrian counts. The model is capable of accurately predicting pedestrian numbers up to 16 days in advance, on multiple look-ahead times. Our system is evaluated with a real-world sensor dataset supplied by the City of Melbourne.

CCS CONCEPTS

• **Human-centered computing → Ubiquitous and mobile computing**; • **Applied computing → Forecasting**; • **Computer systems organization → Sensor networks**;

KEYWORDS

prediction, pedestrian count, mobility patterns, time series

ACM Reference Format:
Xianjing Wang, Jonathan Liono, Will McIntosh, and Flora D. Salim. 2017. Predicting the city foot traffic with pedestrian sensor data. In *MobiQuitous 2017: the 14th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services, November 7–10, 2017, Melbourne, VIC, Australia.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3144457.3152355

1 INTRODUCTION

With the rapid growth of human population in modern cities, the facilities provided in sustainable environments have brought the citizens to be more engaged in moving within the pedestrian friendly urban spaces. The vibrancy and vitality of a city are often indicated by its high pedestrian activities in a walking-friendly environment.

Furthermore, there is also a direct link between the economic prosperity and safety index of a city, and the convenience of the pedestrian experience [11]. For example, 2.9 percent population growth in Melbourne suggests that it lives up to its reputation as a great place to work, visit, socialise and reside. The numbers add up to more than 38.3 percent increase in daily population to the city by the year 2030. The vast majority of this daily population would move as pedestrians within the dynamic urban spaces.

Consequently, a high number of pedestrians indicates the crowdedness of its environment, which is also influenced by various urban factors (e.g. weather and reliability of public transportation services). Such situation where high human density is prominent would require immediate attention (in terms of resource planning and allocation) to cope with the growing pedestrian demands. Hence, there is a tremendous need for both internal and external stakeholders to continue adopting a long-term strategy for urban planning and management; improving city walkability and transport; measuring the impacts of events on pedestrian activity; developing marking strategies to maximise their exposure and identify staffing and resource requirements.

Modelling modern city pedestrian counts is difficult as the level of pedestrian crowdedness varies by locations and changes over time significantly. Although the level of crowdedness can be derived from the counts, the mobility of pedestrians also depends on the spatio-temporal aspects (such as different time of day and location). For example, fewer pedestrians are noticed in the evening (in comparison to the counts in the daytime) for a specific location, while other areas may have a dramatic increase of pedestrian movements where there are prominent bars, restaurants, and night parties on the corresponding streets. Certain groups of people are more interested in the nightlife activities, which results in higher counts of pedestrians on those streets. Even if sensors can be deployed ubiquitously to detect the pedestrians, the tasks for diagnosing and forecasting pedestrian movements solely based on the pedestrian sensor data are inherently challenging. Furthermore, a modern city such as Melbourne has many special events that may change the overall mobility patterns of the pedestrians.

As sensors are becoming smarter to detect the presence of passing pedestrians, modelling pedestrian mobility is a non-trivial task. Many of the previous works have only attempted to predict pedestrian movements in a closed area (i.e. controlled environment).

MobiQuitous 2017, November 7–10, 2017, Melbourne, VIC, Australia

ACM ISBN 978-1-4503-5368-7/17/11…$15.00
https://doi.org/10.1145/3144457.3152355

For example, several methodologies are used to predict the pedestrian movement in a shopping mall based on the previous status of visitors [3]. In other cases, a planning-based prediction for the pedestrians is achieved by determining the robot movements in a closed environment (e. g. kitchen, secretary desk, and lounge areas [23]). Hence, the challenge that we faced during this study is related to the dynamic human movement within urban spaces (i.e. public service areas in the city). Moreover, there is no suggestion about a long-term strategy for urban planning and management. Other methods such as [7, 13] leverage image based estimation approach of pedestrian orientation to improve travel path and traffic predictions. In this paper, the proposed system is tailored to tackle the issue of pedestrian movements in public locations, given various temporal foot traffic patterns that can emerge from pedestrian sensor data. Therefore, the aim of our study is to predict the city foot traffic, given pedestrian sensor data that are continuously streamed by non-vision based sensors.

To tackle the problems of analysing pedestrian mobility patterns in a dynamic urban environment, the main contributions in this paper are as the following:

(1) A robust system for processing time series data (embedded with various temporal trends) and interactive report (including visualisation) for the prediction of city foot traffic.
(2) Accurate short-term predictions for pedestrian counts based on the sensor locations and their intrinsic temporal foot traffic patterns.
(3) Empirical evaluation of various state-of-art models for future prediction of city foot traffic.

Given the proposed system and the main contributions above, the outputs of such predictions allow us to have a better understanding of how the pedestrians move dynamically within the urban spaces, especially when the data are sourced from non-intrusive devices (non-vision based sensors). By having the insights of how people utilise the city, it could lead to the improved coordination and planning for satisfying future citizenship needs. Furthermore, it allows the authorities to improve pedestrian walking service and in turn increase user's satisfaction with the city life experience.

Throughout this work, there are several key motivations for knowing more about current and future pedestrian movement. The corresponding authorities identify them as the following:

- The city has a focus to effectively plan for the future to ensure the city remains a pedestrian friendly and easy access city.
- Planning requires data and an understanding of current trends and future demand.
- The understanding of future state allows the city to plan for additional infrastructure (such as footpath construction that eases congestion).

The rest of the paper is organised as follows. Section 2 provides the background and related work. In Section 3, the analysis of pedestrian counts data is elaborated extensively. Section 4 includes our proposed system for predicting the city foot traffic with pedestrian sensor data. The forecasting models used in our experiment are then described in Section 5. Hence, Section 6 presents the performance evaluation of these forecasting models. Finally, we conclude our studies in Section 7, including the future work.

2 RELATED WORKS

Doan et al. in 2015 used the City of Melbourne pedestrian movements dataset for their research [6]. They tested whether the accuracy of an Ensemble Switching outweighs its complexity compared to their static HyCARCE clustering model for modelling pedestrian movements. Doan et al. [6] focused on anomalous events and analysing the HyCARCE model. The experiments show that pedestrian distributions can be clustered into meaningful profiles characterising major activities throughout the day. HyCARCE detects the crowd time periods for different major events throughout the event day, which is for a specific day. This research does not provide a general solution for the normal pedestrian activity pattern. Compared to Doan et al.'s research work [6], we concentrate on the forecasting of pedestrian counts for the future time periods instead of discovering meaningful clustering patterns.

Many of previous works on forecasting pedestrian flow have only attempted to predict pedestrian's movement in a closed area. For example, some method to predict the pedestrian movement in a shopping mall from people's previous status [3], or a planning-based prediction for pedestrian for determining robot movements in a closed environment such as a kitchen area, secretary desk area, and lounge area [23]. In fact, these studies are more focused on private area prediction instead of the pedestrian activity in public service area in the city. In other domain such as human occupancy counting, the human presence is detected by modelling time series data from ambient sensors in a closed office space environment [2].

Furthermore, there is no suggestion about a long-term strategy for urban planning and management. The other approaches are related to the image-based estimation of pedestrian orientation for improving path prediction and traffic [7, 13]. In [14], the pedestrian counts in groups are estimated and tracked using kalman filtering techniques.

Another related work such as [22] focused on sensors in public areas. However, they required the data from vision-based sensors (i.e. surveillance video cameras). Thus, high dimensional features are extracted and be used as the inputs for image processing techniques and non-linear regression models, in order to quantify the pedestrian counts in crowded areas. However, the solution for non-vision based sensors is different, due to the fact of having significantly less number of dimensions. In our case, a sensor only captures the pedestrian count in last hour at a given time. Moreover, their work focused on quantifying the pedestrian counts for the real-time application instead of a future prediction of city foot traffic, which is the main problem we discuss in this paper. Thus, it is important to model city foot traffic patterns based on time-series analysis, for the purpose of forecasting report (i.e. future prediction of pedestrian counts).

There is a significant amount of analysis about short term analysis of human mobility using different prediction methodologies. Moreover, several related works are using the number of available bikes in the stations of the community bicycle program [12]. It should be noted that these works are not based on pedestrian data. In [12, 18], only next 60 minutes and 2 hours are estimated.

Bezuglov et al. [4] proposed the grey system theory models for short term traffic speed prediction study. They also compared the nonlinear time series models. Their grey models demonstrated better accuracy performance in comparison to other tested nonlinear models using Root Mean Squared Errors and Mean Absolute Percent Errors. In [8], the article presents a multi-agent system model for virtual power plants. The proposed model manages the different elements including a set of agents embedded with artificial neural networks for collaborative forecasting of disaggregated energy demand of domestic end users.

3 PEDESTRIAN COUNTS IN MELBOURNE

3.1 Pedestrian Counts Dataset

Nowadays, location-based urban data are highly available such as point of interests, real-time traffic and user check-in data. However, the pedestrian counts used in this paper are derived from non-vision based sensors in Melbourne, Australia. The open dataset is widely accessible from City of Melbourne's 24-hour pedestrian counting system. There are more than 42 sensors installed within range of Melbourne Central Business District (CBD). In [16], visual representation of pedestrian volume is provided for each location from year 2009 to 2017. We leveraged their open dataset for our study in this paper.

The types of sensors deployed in City of Melbourne's Pedestrian Counting System are thermal and laser based sensors. Typically, a sensor is installed under an awning or on a street pole forming a counting zone on the footpath below. It records all multi-directional pedestrian movements passing through the zone. The data collected from a wireless data transmission system is stored in the onsite data logger and be transferred to the central server every 10-15 minutes, which then be integrated to the data visualisation website every hour. The target locations were selected based on following three criteria:

- Main pedestrian thoroughfares
- Retail and event activity
- Egress and entry flow to these areas

As mentioned previously, these non-vision based sensors are set to only record human movements. Hence, no personal information is collected since no image is captured. Each sensor is labelled by its street or station name. Particular streets or stations have multiple sensors for different directions, which are labelled according to their directions.

The Pedestrian Counting System provides an information service for users through the City of Melbourne Council website (refers to Figure 1). It shows a dynamic map of Melbourne CBD area overlaid with a small vertical bar for each sensor location, which includes the value of pedestrian counts and its detailed daily summary. In each daily summary of a particular sensor, we can perform a direct hourly comparison between real-time pedestrian counts of the day and the average pedestrian counts of the previous four weeks and fifty-two weeks. Each data point in the visualisation provides the pedestrian count information for each hour. Our observation also revealed that the pedestrian dataset contains missing data. Moreover, the hourly count represents repetitive zero values when the sensor is unavailable due to a technical reason.

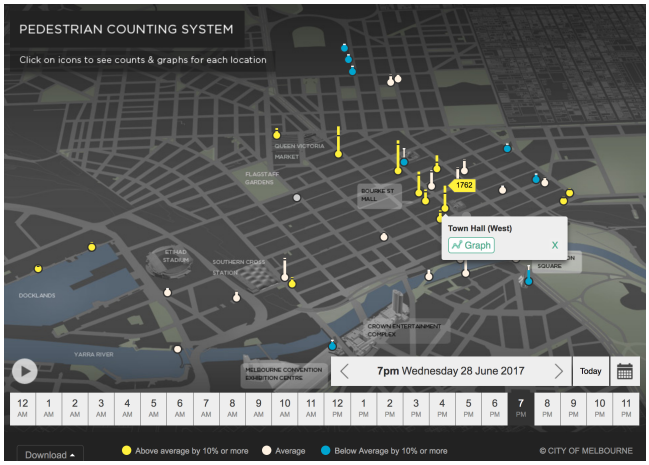


Figure 1: City of Melbourne's 24-hour pedestrian counting system [16]

Their open data is structured with the following information:

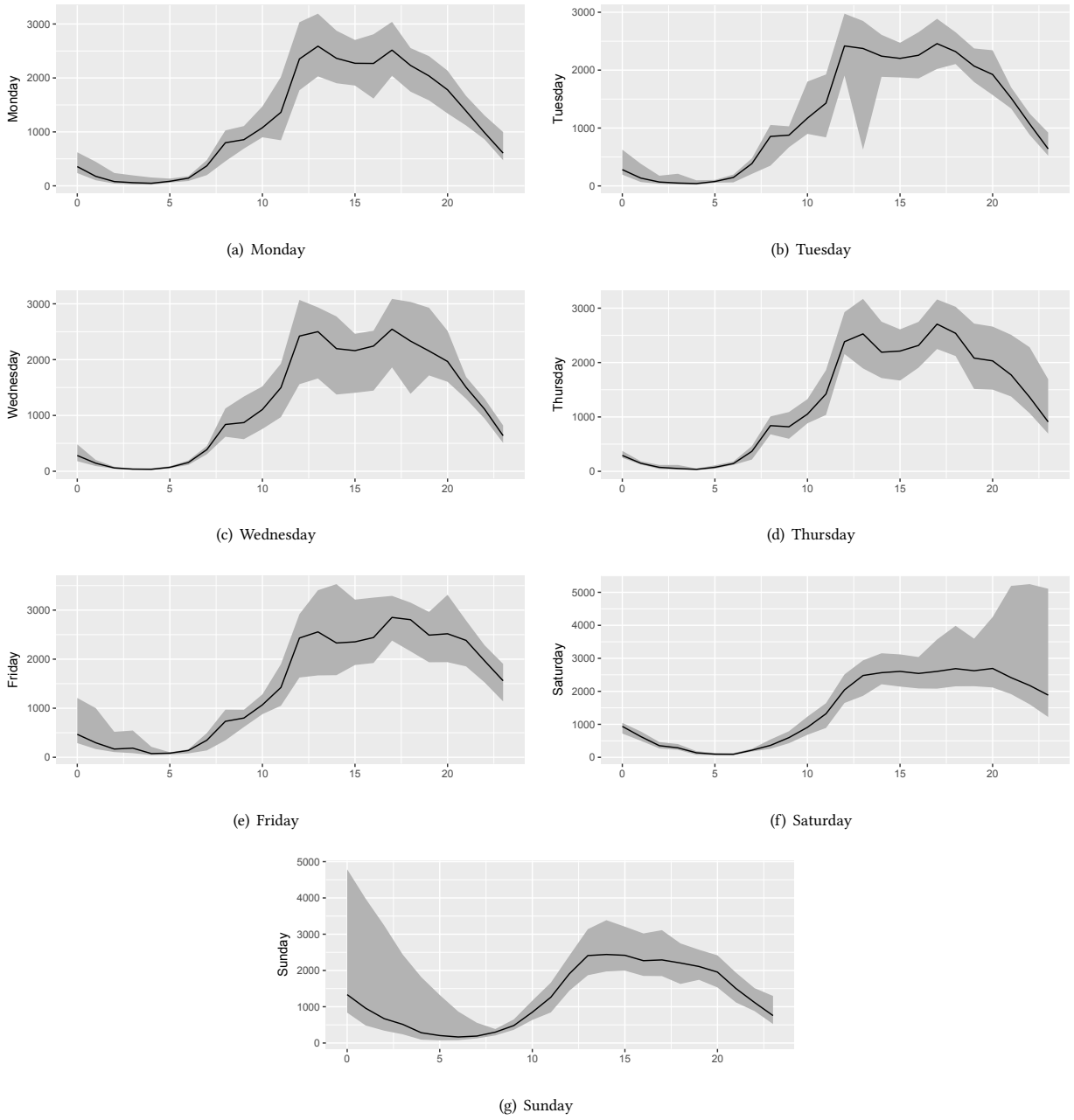
(1) Station name
(2) Street name
(3) Sensor location name and sensor id
(4) Latitude and longitude of each sensor
(5) Number of pedestrian present in the counting zone
(6) Detailed timestamps of each day

In order to analyse the dynamics of station or street loads, we have been collecting these CSV documents since March 2017 and stored all the relevant information, such as the station name, localisation, pedestrian counts and timestamps. As the Pedestrian data are updated from time to time, the dimension of the data increases as new sensors are also added (corresponding to certain stations). In overall, the deployment has grown from 34 to 42 sensors, ever since the information of pedestrian counts data were made publicly accessible through the Pedestrian Counting website.

3.2 Analysis of Pedestrian Counts Data

Before we begin calculating pedestrian activity, we take a closer look at the data collected from City of Melbourne Council website. The plot in Figure 2 shows an example of the collected time series data from a railway train station that is close to large shopping centre, office, public transport and university campus. The data collection started from January to March 2016. In other words, the observation of total three months time series data was applied to one sensor as an example. The black line indicates the average number of pedestrian counts captured by this particular sensor within three months. The figures also show the overview of average pedestrians by the weekly patterns for a particular train station (a sensor in a given location). Furthermore, the grey areas correspond to minimum and maximum bounds of the pedestrian counts.

From our observation, the data is relatively noisy with several sudden increases and declines in the minimum and maximum pedestrian counts. These trends can be caused by special events or extreme weather, which could affect the local pedestrian activities. The average weekly activity patterns are shown in Figure 2 from



(a) Monday (b) Tuesday

(c) Wednesday (d) Thursday

(e) Friday (f) Saturday

(g) Sunday

Figure 2: Summary of Pedestrian Mobility Week Patterns (The data collection started from January to March 2016).

Monday to Sunday, which allows us to average out those fluctuations. Therefore, unpredictable events such as traffic accidents and natural disasters are excluded in the rest of this study. Tuesday, 26 January 2016 is Labour Day, which may be the reason to cause the drop in Figure 2(b) for its minimum number of the pedestrian. Monday, 14 March 2016 is an Australia Day. Such festivals are also excluded and be filtered out for our analysis in this paper. It should be noted that the foot traffic patterns associated with these events will be included in our future work, in order to improve our proposed system.

The greater standard deviation of the weekend patterns are primarily caused by the public holiday. There are two distinct patterns that we have observed: weekdays and weekend foot traffic patterns. This differentiation is verified by more detailed analysis of the weekdays (Monday - Friday) and weekend patterns (Saturday and Sunday) in the given Figure 2. The weekdays and weekend patterns

across three month time period are compared. We first concentrate on the weekdays foot traffic patterns. In fact, the weekdays patterns shown in Figure 2 are focused on one of Melbourne's major railway train stations. Relative low activity pattern is noticed between 00:00 and 7:00 am, the average number of pedestrian under this sensor is less than 500 in weekdays patterns. There is an upward trend during 7:00 to 11:00 as people normally start the daytime activities within this time range. Not until 12:00 to 1:00 the pedestrian counts reaches a peak point. There are two reasons that may explain this intrinsic behaviour. Firstly, several people started their activities in the middle of the day. Secondly, there are many restaurants near this station in the shopping centre. Consequently, people would be seeking for food in the lunchtime. Furthermore, this station experiences the second peak starting at 17:00 and reaches its maximum at 18:00 in the afternoon. This may either be caused by people leaving the office or university to off from work or study. Finally, this station also experienced a high number of pedestrian activity around 20:00 caused by the popular bars and restaurants near or within the shopping centre.

Compared with weekdays patterns, the weekend patterns are distinctively different, thus having greater standard deviation. We can observe that the boundary of the maximum pedestrian count is about 5000 in the midnight of Saturday. This change can be explained that people are more active and likely to have more outdoor activities during the evening in the weekend. In the daytime, the average number of pedestrian under this sensor is about 2500 at peak time from 13:00 to 20:00, which is significantly lesser than the peak of weekdays foot traffic patterns.

4 CITY FOOT TRAFFIC PREDICTION

4.1 System Architecture

In order to handle the variability and inconsistency of temporal foot traffic patterns emerging from the time series of pedestrian sensor data, we present the design of City Foot Traffic Prediction system in this section.

The general system architecture is described in Figure 3. City Foot Traffic Prediction system consists of the following three core components:

4.1.1 Construction of Forecasting Model. The objective of this component is to produce the forecasting models that can be used for prediction tasks. First, the **historical records** are retrieved from the Pedestrian Counting System. Essentially, these historical records require substantial **data cleaning** process, which includes the selection and extraction of temporal foot traffic patterns. In this case, we extract both weekdays and weekend foot traffic patterns. Once these patterns have been extracted from the time-series data, the **model building** process is initiated, given a set of predefined predictive algorithms. Subsequently, the **model selection** process is applied to the **forecasting models** that are produced from the previous step. The criteria for selecting the forecasting model depend on the evaluation metrics that are used to measure the predictive performance (refers to Section 6).

4.1.2 Prediction of Future Pedestrian Counts. This component is mainly responsible for executing the predictive tasks. Previous n-weeks time series data of pedestrian counts are required as the input. Consequently, data **pre-processing** is required to produce the data in the desired format, which will be the input for the **predict** function. In this function, the **short-term forecasting model** from output of the model selection process is used. Inherently, there are two possible outcomes of the predict function: success and failure. When the predict function fails to execute, it would then be forwarded to **error handling** component as a fall-back mechanism. For example, default parameters for the ARIMA model would be applied when the predict function fails to execute. Otherwise, the **forecasting result** for future pedestrian counts is produced as the output of a successful prediction task for a given sensor location.

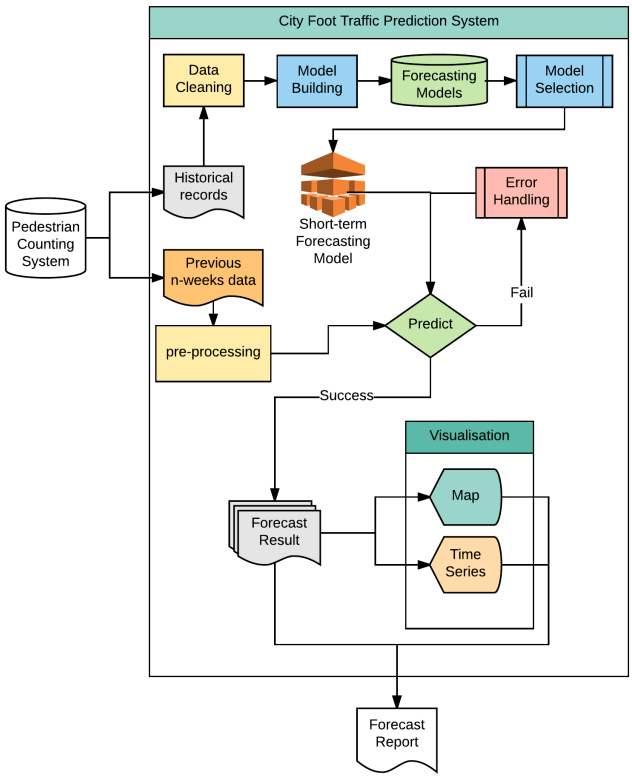


Figure 3: General System Architecture of City Foot Traffic Prediction System

4.1.3 Visualisation and Report Generation. Once the **forecasting result** has been produced from the previous step, meaningful representation is generated through this component. Consequently, the report generation requires the consolidation of the forecasting result. In this case, data summarisation and visualisation are used in the **forecast report** generation.

4.2 System Implementation

City Foot Traffic Prediction system was implemented in R programming language. For the reporting module, Knitr [20, 21] is used as the engine for dynamic report generation with R. In order to perform both weekdays and weekend prediction for the next two weeks, the system requires a CSV input file that contains at least previous four weeks of pedestrian counts data. These data can be

extracted from the Pedestrian Counting System. Moreover, the system is specifically developed to be executed on a standard Windows Operating System.

5 SHORT-TERM FORECASTING MODELS

5.1 ARIMA Model

The ARIMA algorithm is a statistical method for analysing and building a forecasting model which best represents a time series by modelling the correlations in the data. ARIMA model is one of the most widely-used approaches to time series forecasting, and provide complementary approaches to the problem [19] [10].

The aim of ARIMA model is to describe the autocorrelations in the data.The acronym ARIMA stands for Auto-Regressive Integrated Moving Average. Lags of the stationarised series in the forecasting equation are called "autoregressive (AR)" terms, lags of the forecast errors are called "moving average (MA)" terms, and a time series which needs to be differenced to be made stationary is said to be an "integrated" version of a stationary series [19] [9]. In an autoregression model, we forecast the variable of interest using a linear combination of past values of the variable. The term autoregression indicates that it is a regression of the variable against itself [10]. Thus, an autoregressive model of order p can be written as

$$y_t = c + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \ldots + \phi_p y_{t-p} + e_t, \quad (1)$$

where c is a constant and e_t is white noise. $y_{t-1}, y_{t-2} \ldots y_{t-p}$ are the past series values named as lags. We refer to this as $AR(p)$ model. A moving average model applies past forecast errors in a regression-like model. The q order moving average model denoted by $MA(q)$ is

$$y_t = c + e_t + \theta_1 e_{t-1} + \theta_2 e_{t-2} + \ldots + \theta_q e_{t-q}, \quad (2)$$

where e_t is white noise, and $\theta_1 \ldots \theta_q$ are constants. If combining differencing with autoregression and a moving average model, we obtain a non-seasonal ARIMA model. The full model can be written as

$$y'_t = c + \phi_1 y'_{t-1} + \ldots + \phi_p y'_{t-p} + \ldots + \phi_q e_{t-q} + e_t, \quad (3)$$

where y'_t is the differenced series that include both lagged values and lagged errors. A non-seasonal ARIMA model is usually specified as an $ARIMA(p, d, q)$ model, where:

- p is the number of autoregressive terms,
- d is the number of non-seasonal differences needed for stationarity, and
- q is the number of lagged forecast errors in the prediction equation.

We select the three parameters based on Hyndman's method presented in [10]. In order to identify the numbers of AR or MA terms in an ARIMA model, a time series needs to be stationarised by the difference in the first step. The next step in fitting an ARIMA model is to determine whether AR or MA terms are needed to correct any autocorrelation that remains in the differenced series. We use autocorrelation function (ACF) plot, and the closely related partial autocorrelation (PACF) plot to determine appropriate values for p and q. ACF plot shows the autocorrelation which measure

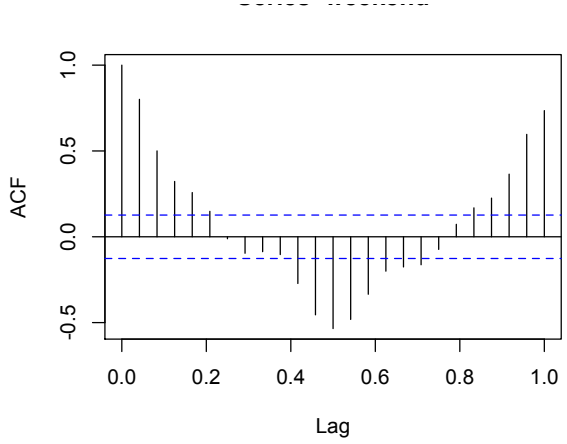


Figure 4: An estimate of the autocorrelation function of a pedestrian counts time series

the relationship between y_t and y_{t-k} for different values of k. If y_t and y_{t-1} are correlated, then y_{t-1} and y_{t-2} must also be correlated. Then y_t and y_{t-2} might be correlated. This is because both y_t and y_{t-2} are connected to y_{t-1}, which can be applied in forecasting y_t. The measurement of relationship between y_t and y_{t-k} after removing the effects of other time lags from $1, 2, 3, \ldots, k-1$ can be done by using PACF. The first partial autocorrelation is identical to the first autocorrelation. The partial autocorrelation for lags 2, 3 and greater are calculated as a_k equal to the kth partial autocorrelation coefficient, and a_k also equal to the estimate of θ_k in the autoregression model as follows:

$$y_t = c + \theta_1 y_{t-1} + \theta_2 y_{t-2} + \ldots + \theta_k y_{t-k} + e_t. \quad (4)$$

Three factors that should be considered to determine the first guess at an ARIMA model are: a time series plot of pedestrian data, the autocorrelation and the partial autocorrelation for pedestrian count data. The ACF and PACF should be considered together. The following charts in Figure 4 and Figure 5 demonstrate the ACF and PACF plots for the weekend pedestrian count data from January to April 2017. The partial autocorrelations have the same critical values of $| z | > 1.96$ as for ordinary autocorrelations. In Figure 4, there are three spikes decreasing with the lag in the ACF and then no significant spikes thereafter. In PACF shown in Figure 5, there are two spikes and no other significant spikes apart from one just few just outside the bounds 0.4. The ignored spikes in each plot are the ones outside the limits, and not in the first few lags. The probability of a spike being significant by chance is about one in twenty four. In total, there are 24 spikes in each plot ACF and PACF. The pattern in the first two spikes in PACF specifies the expectation from an $AR(2)$.

Let us consider the selection best parameters by comparing the value of: Akaike's Information Criterion (AIC) on a set of models and investigate the models with the lowest AIC values. Table 1 demonstrates all the AIC values comparison between different parameters applied on the pedestrian forecasting model. We obtain $ARIMA(2, 0, 1)$ as the best model for predicting pedestrian counts

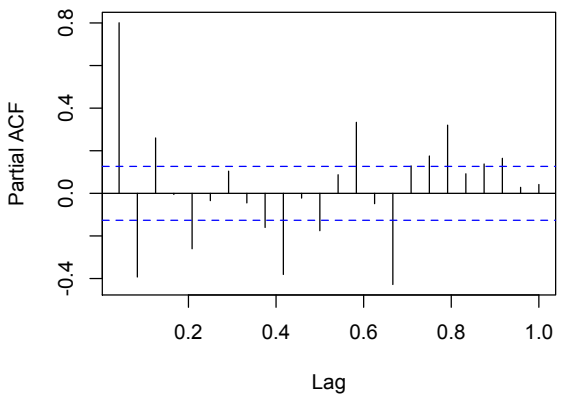


Figure 5: An estimate of the partial autocorrelation function of a pedestrian counts time series

from the lowest AIC comparison. We test various ARIMA models on the pedestrian counts dataset in 2015–2017. There are more than 16 models compared for choosing the best parameters: $ARIMA(0,0,1)$, $ARIMA(0,0,2)$, ..., $ARIMA(3,0,3)$. The table below illustrates the AIC comparison among p in $AR(p)$ and q in $MA(q)$. The lowest AICs are highlighted in bold in three different ARIMA models. The worst two models are also highlighted in Italic. The three bold highlighted ARIMA models has the lowest AICs in the table, such as $ARIMA(2,0,2)$ and $ARIMA(2,0,3)$ may offer better fit than $ARIMA(2,0,1)$, however that fit is not worth the loss in parsimony imposed by the addition of increased AR and MA lags.

	p=**0**	p=**1**	p=**2**	p=**3**
q=**0**	*2929.71*	2578.40	2479.25	2455.02
q=**1**	*2709.98*	2507.54	**2462.24**	2452.84
q=**2**	2597.26	2485.62	**2443.05**	2448.70
q=**3**	2594.13	2481.83	**2441.66**	2451.54

Table 1: AIC values with respect to different values of parameters p in $AR(p)$ and q in $MA(q)$

5.2 Baseline models

In practice, we compare the forecasting performance of three different advanced linear regression models: autoregressive integrated moving average, support vector regression (SVR), and multiple linear regression (MLR) models. We use SVR and MLR models as our baseline models.

5.2.1 Support vector machine regression model. Support vector machine (SVM) analysis is a popular machine learning tool for classification and regression, first identified by Vladimir Vapnik and his colleagues in 1992 [5]. SVM can also be applied as a regression method, maintaining all the main features that characterise the algorithm (maximal margin) [17]. When it is applied to a regression problem, it is termed as support vector regression.

Suppose we have given training data $\{(x_1, y_1), ..., (x_l, y_l)\} \subset X \times \mathbb{R}$, where X denotes the space of the input patterns. When there is a linearly separable set of points of two different classes, the objective of a SVM is to find where that particular hyperplane in which separates these two classes with minimum error. While it also making sure that the perpendicular distance between the two closes points from either of these two classes is maximized. That is the method how the hyperplane is determined. A separating hyperplane in canonical form must satisfy the following constrains,

$$y_i[\langle w, x_i \rangle + b]\ 1, \quad i = 1, \ldots, l, \tag{5}$$

where $\langle ., . \rangle$ denotes the dot product in X. Hyperplane in the case of Eq. 5 means that one seeks small w. There is one way to minimize the Euclidean norm. The problem as a convex optimization problem by requiring minimize $\frac{1}{2} \parallel w \parallel^2$ which is subject to:

$$\begin{aligned} y_i - \langle w, x_i \rangle - b &\leq \varepsilon \\ \langle w, x_i \rangle + b - y_i &\leq \varepsilon \end{aligned} \tag{6}$$

The assumption in Eq. 7 is that such a function y actually exists that approximates all pairs (x_i, y_i) with ε precision [17], support vector regression attempts to minimize the generalization error bound to achieve generalized performance. The idea of SVR is based on the computation of a linear regression function in a high dimensional feature space where the input data are mapped through a non linear function.

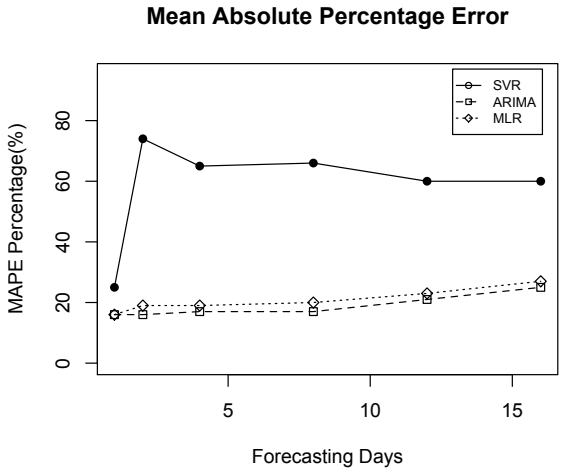


Figure 6: The forecasting performance of three different models evaluated via the mean absolute percentage error.

We use R interface to `libsvm` in package *e*1071, `svm()`, is designed to be as intuitive as possible [15]. The models are fitted and new data can be predicted. Both matrix and the formula interface are implemented in the R's statistical functions. The engine uses the dependent variable's type (y): if y is a factor, the engine switched to classification mode, otherwise, it behaves as a regression machine. If y is omitted, the engine assumes a novelty detection task.

5.2.2 Multiple linear regression model. A multiple linear regression analysis attempts to model the relationship between two or more explanatory variables and a response variable by fitting a linear equation to observed data [1]. Each value of the independent variable x is associated with a value of the dependent variable y. This carries out to predict the value of a dependent variable, Y, given a set of p explanatory variables $(x_1, x_2, ..., x_p)$ is defined to $\mu_y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_p x_p$. The equation describes how the mean response μ_y changes with the explanatory variables. The observed values for y vary about their means μ_y and are assumed to have the same standard deviation. The formal model for multiple linear regression, given n observations, is:

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + ... + \beta_p x_{ip} + \varepsilon_i, \quad i = 1, 2, \ldots, n. \quad (7)$$

We use R interface `lm()` is used to fit linear models with addition of a one or more predictors. It can be used to perform multiple linear regression, and single stratum analysis of variance and analysis of covariance. As a predictive analysis, we model the relationship between three explanatory variables — Spencer St-Collins St (North), Spencer St-Collins St (South) and Southern Cross Station sensors. We used Spencer St-Collins St (North) as one continuous dependent variable and the other two sensors as independent variables. In next section, we evaluate the forecasting performance of the relationship between three explanatory variables for pedestrian count data.

6 PERFORMANCE EVALUATION

In this section, we present initial results on the prediction of pedestrian counts at a given time and location. We compare several linear regression models and establish evaluation measurements, including the baseline with which other models can be compared. The evaluation that we present in this section would be used as the inclusive process (refers to **model selection** process in Section 4.1.1) in our proposed system. The framework leveraged by the system can essentially be used not only for forecasting the pedestrian counts, but also for other forecasting problems.

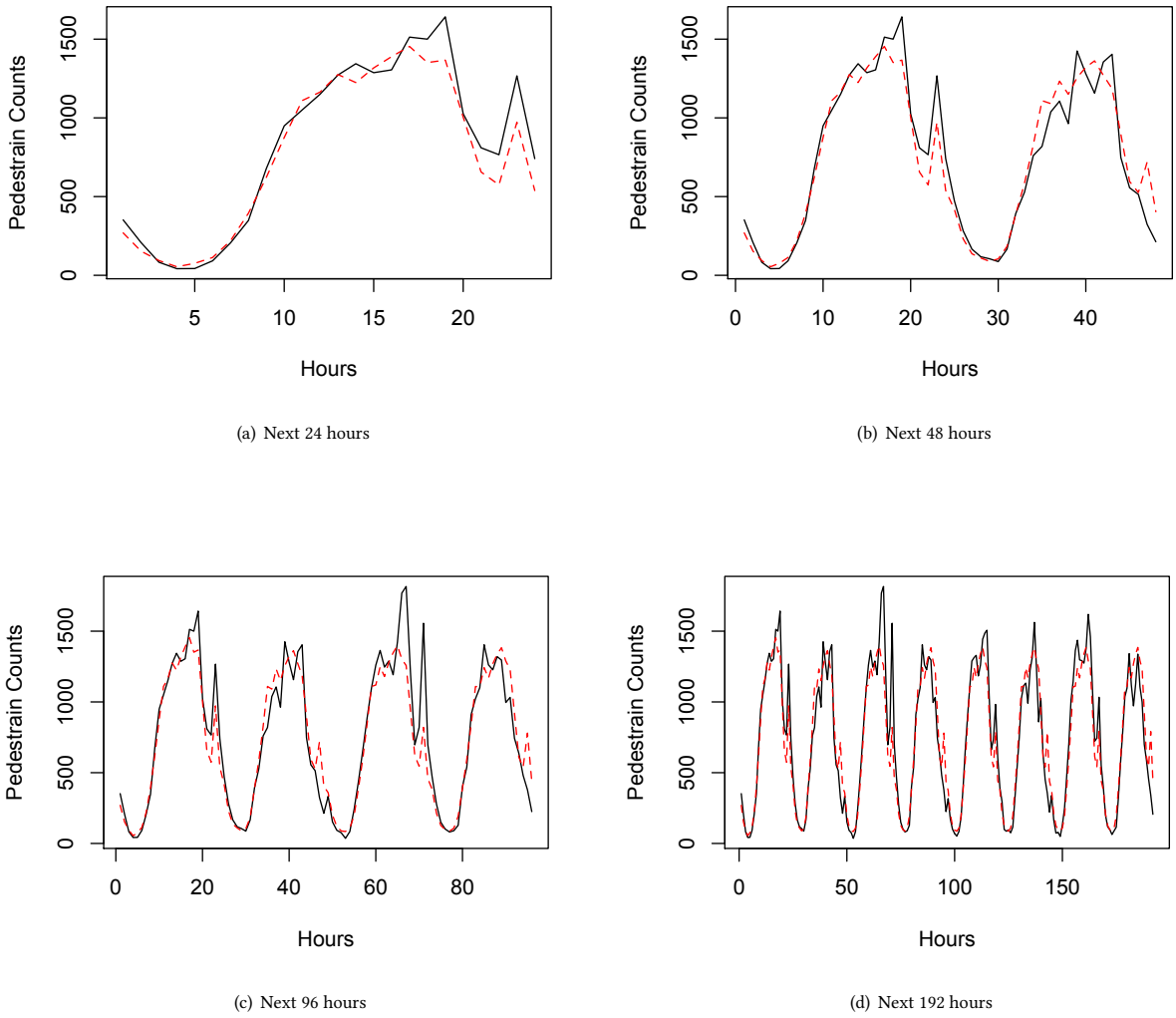
In our experiment, we compare the forecasting performance of three different advanced linear regression models: autoregressive integrated moving average (ARIMA), support vector regression (SVR), and multiple linear regression (MLR) models. Empirical evaluation is performed on these models by measuring the mean absolute percentage error (MAPE) over all available dates and all available sensors. Consequently, the evaluation covers various time periods, i.e. one day, two days, four days, ... and more than 16 days into the future.

Figure 6 shows an example of overall forecasting performance of ARIMA, SVR, and MLR models evaluated via the mean absolute percentage error for one particular sensor and different time periods. The black curve corresponds to the mean absolute percentage error value for SVR model, while the dotted line with diamond and dashed line with square indicate the mean absolute percentage error value for MLR and ARIMA models. In this example, the ARIMA model with best-selected parameters achieves a much lower MAPE (indicated be the red line in Figure 6) using the pedestrian counts of August 2016 dataset. The lower MAPE shows a better performance in term of accurate forecasting method. Low MAPE result for SVR model (dotted line with square in Figure 6) indicates a better forecasting result in the next 24 hours prediction. However, the mean absolute percentage error result increases dramatically from next 24 hours. For next 48, 96, and 192 hours prediction, SVR models are not optimal given the MAPE performance evaluation. The mean absolute percentage error of MLR model is slightly higher than ARIMA model across different prediction time periods. For the overall forecasting performance, the ARIMA model has the lowest MAPE and the error keeps low within 16 days' forecasting result.

From the application input perspective, both SVR and ARIMA models request only one sensor as input string for one pipeline, while the MLR model needs two or more than two input pipeline to build the predictive model. We modelled the relationship between three explanatory variables — Spencer St-Collins St (North), Spencer St-Collins St (South) and Southern Cross Station sensors in MLR. We used Spencer St-Collins St (North) as one continuous dependent variable and the other two sensors as independent variables. This increase the complexity of building a predictive application. Given the overall ARIMA performance is better than the MLR for its lowest mean absolute percentage error, and the simple input pipeline system. The forecasting performance of ARIMA model with best-selected parameters beats the other two linear regression models: SVR and MLR.

The ARIMA models are trained by pedestrian count dataset from 2015 to 2017 March for all the sensors in Southern Cross area. We selected a continuous section of about 300 hours of historical data for each sensor for training the models. In all the experiments presented here, we used a history dataset, the same sensor or location the predictions are generated for (AR component) and the surrounding sensor or location (MA component) to generate the predictions. Hence, the corresponding orders of our ARIMA model are $p = 2$, $d = 0$, and $q = 1$ in $ARIMA(2, 0, 1)$.

Figure 7(a), Figure 7(b), Figure 7(c) and Figure 7(d) show the forecasting result for the next 24, 48, 96 and 192 hours for one specific sensor in Southern Cross area, using the weekend dataset for forecasting (August and September in the year 2016). The solid black line represents the ground truth of hourly pedestrian count in the day. The dashed red line represents the forecasting result for ARIMA model in different time periods in the four figures. As shown in Figure 7(a), the forecasting result is very close to the ground truth. Two lines are noticed to have similar city foot traffic pattern from the first hour to twenty-four hours in the future. The pedestrian count reaches the lowest number at midnight around 4am and starts to increase dramatically from 5am in the morning to 3pm in the afternoon, then following with some noise around 6pm. Similar to the rest three figures (Figure 7(b), Figure 7(c) and Figure 7(d)), there are some differences between the forecasting results and ground truth from the 15th to the 20th hours of the day. The performance of ARIMA models shows the lowest MAPE. Moreover, its forecasting results are closer to the ground truth within the limit of 192 hours for future prediction.



(a) Next 24 hours

(b) Next 48 hours

(c) Next 96 hours

(d) Next 192 hours

Figure 7: ARIMA forecasting results

7 CONCLUSIONS AND FUTURE WORK

In this paper, a robust pedestrian forecasting model is presented for the case study of pedestrian sensor locations (the Southern Cross area) in Melbourne, Australia. We choose a time series ARIMA model for its best forecasting performance via the analysis of both weekdays and weekend foot traffic patterns among three different linear regression models. By investigating the lowest AIC value, the best parameters for ARIMA model is selected and tested on ground truth. From the MAPE comparison between different time period, the ARIMA model provides accurate and stable forecasting result in next 192 hours. The best time period for the weekend model is next two weekends. The best time period for the weekdays model is next two weekdays. Most importantly, the forecasting model is mainly leveraged by our proposed system to predict city foot traffic. Both City Foot Traffic Prediction system and forecasting models are implemented in R programming language. The code is designed to execute on a standard Windows desktop environment. A visualisation of forecasted pedestrian counts will be presented in the PDF file while executing the model.

For the larger scale of our future work, there are several improvements that could be studied and applied to enhance the current system. First, the modelling of different patterns such as public holidays and important events. Currently, only weekend and weekdays pedestrian counts are predicted effectively. Different patterns could emerge from such abnormal patterns that can be extracted from special events (e.g. public holidays or major football league events).

Secondly, a robust and integrated pedestrian count forecasting system needs to be built for deep analysis of pedestrian mobility patterns between sets of sensors. In other words, the association between sensors (including auxiliary data) would be critical for the improvement of the current forecasting system. Such predictions would allow us to give City of Melbourne a better understanding of how the people use the city at future times of the day so that we can manage the way it functions and plan for future needs. In addition, other sources of information can be leveraged, such as weather data, road networks data and public transportation schedules.

Furthermore, the exploration and application of cluster analysis (an unsupervised machine learning methodology) are needed. This approach would let us distinguish sets of sensors into many groups/clusters for similar pedestrian mobility patterns by calculating the small distances among the cluster members.

8 ACKNOWLEDGMENT

This research project is funded and supported by City of Melbourne.

REFERENCES

[1] David F Andrews. 1974. A robust method for multiple linear regression. *Technometrics* 16, 4 (1974), 523–531.

[2] Irvan Bastian Arief Ang, Flora Dilys Salim, and Margaret Hamilton. 2016. Human occupancy recognition with multivariate ambient sensors. In *Pervasive Computing and Communication Workshops (PerCom Workshops), 2016 IEEE International Conference on*. IEEE, 1–6.

[3] Akinori Asahara, Kishiko Maruyama, Akiko Sato, and Kouichi Seto. 2011. Pedestrian-movement prediction based on mixed Markov-chain model. In *Proceedings of the 19th ACM SIGSPATIAL international conference on advances in geographic information systems*. ACM, 25–33.

[4] Anton Bezuglov and Gurcan Comert. 2016. Short-term freeway traffic parameter prediction: Application of grey system theory models. *Expert Systems with Applications* 62 (2016), 284–292.

[5] Bernhard E. Boser, Isabelle M. Guyon, and Vladimir N. Vapnik. 1992. A Training Algorithm for Optimal Margin Classifiers. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory (COLT '92)*. ACM, New York, NY, USA, 144–152.

[6] Minh Tuan Doan, Sutharshan Rajasegarar, Mahsa Salehi, Masud Moshtaghi, and Christopher Leckie. 2015. Profiling pedestrian distribution and anomaly detection in a dynamic environment. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*. ACM, 1827–1830.

[7] Tarak Gandhi and Mohan Manubhai Trivedi. 2008. Image based estimation of pedestrian orientation for improving path prediction. In *Intelligent Vehicles Symposium*. IEEE, 506–511.

[8] Luis Hernández, Carlos Baladron, Javier M Aguiar, Belen Carro, Antonio Sanchez-Esguevillas, Jaime Lloret, David Chinarro, Jorge J Gomez-Sanz, and Diane Cook. 2013. A multi-agent system architecture for smart grid management and forecasting of energy demand in virtual power plants. *IEEE Communications Magazine* 51, 1 (2013), 106–113.

[9] SL Ho, M Xie, and TN Goh. 2002. A comparative study of neural network and Box-Jenkins ARIMA modeling in time series prediction. *Computers & Industrial Engineering* 42, 2 (2002), 371–375.

[10] Rob J Hyndman and George Athanasopoulos. 2014. *Forecasting: principles and practice*. OTexts.

[11] Frank Jaskiewicz. 2000. Pedestrian level of service based on trip quality. *Transportation Research Circular, TRB* (2000).

[12] Andreas Kaltenbrunner, Rodrigo Meza, Jens Grivolla, Joan Codina, and Rafael Banchs. 2010. Urban cycles and mobility patterns: Exploring and predicting trends in a bicycle-based public transport system. *Pervasive and Mobile Computing* 6, 4 (2010), 455–466.

[13] C. G. Keller and D. M. Gavrila. 2014. Will the Pedestrian Cross? A Study on Pedestrian Path Prediction. *IEEE Transactions on Intelligent Transportation Systems* 15, 2 (April 2014), 494–506.

[14] Prahlad Kilambi, Evan Ribnick, Ajay J Joshi, Osama Masoud, and Nikolaos Papanikolopoulos. 2008. Estimating pedestrian counts in groups. *Computer Vision and Image Understanding* 110, 1 (2008), 43–59.

[15] David Meyer, Evgenia Dimitriadou, Kurt Hornik, Andreas Weingessel, Friedrich Leisch, Chih-Chung Chang, Chih-Chen Lin, and Maintainer David Meyer. 2017. Package 'e1071'. (2017).

[16] City of Melbourne. 2017. Pedestrian Counting System. (2017). http://www.pedestrian.melbourne.vic.gov.au/ [Online; accessed 28-June-2017].

[17] Alex J Smola and Bernhard Schölkopf. 2004. A tutorial on support vector regression. *Statistics and computing* 14, 3 (2004), 199–222.

[18] A Tascikaraoglu and M Uzunoglu. 2014. A review of combined approaches for prediction of short-term wind speed and power. *Renewable and Sustainable Energy Reviews* 34 (2014), 243–254.

[19] JP Wu and Shuony Wei. 1989. *Time series analysis*. Hunan Science and Technology Press, ChangSha.

[20] Yihui Xie. 2013. knitr: A general-purpose package for dynamic report generation in R. *R package version* 1, 7 (2013), 1.

[21] Yihui Xie. 2014. knitr: a comprehensive tool for reproducible research in R. *Implement Reprod Res* 1 (2014), 20.

[22] Junping Zhang, Ben Tan, Fei Sha, and Li He. 2011. Predicting pedestrian counts in crowded scenes with rich and high-dimensional features. *IEEE Transactions on Intelligent Transportation Systems* 12, 4 (2011), 1037–1046.

[23] Brian D Ziebart, Nathan Ratliff, Garratt Gallagher, Christoph Mertz, Kevin Peterson, J Andrew Bagnell, Martial Hebert, Anind K Dey, and Siddhartha Srinivasa. 2009. Planning-based prediction for pedestrians. In *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*. IEEE, 3931–3936.