5-2015

Self-Learning License Plate Matching Algorithm: Some Enhancements and Its Role in Travel Time Ground Truth Measurements

Stephanie Raven Ann Hargrove

University of Tennessee - Knoxville, shargro1@utk.edu

Recommended Citation

To the Graduate Council:

I am submitting herewith a dissertation written by Stephanie Raven Ann Hargrove entitled "Self-Learning License Plate Matching Algorithm: Some Enhancements and Its Role in Travel Time Ground Truth Measurements." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Civil Engineering.

Lee D. Han, Major Professor

We have read this dissertation and recommend its acceptance:

Christopher Cherry, Xueping Li, William Seaver

Accepted for the Council:

Dixie L. Thompson

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)
Self-Learning License Plate Matching Algorithm: Some Enhancements and Its Role in Travel Time Ground Truth Measurements

A Dissertation Presented for the Doctor of Philosophy Degree The University of Tennessee, Knoxville

Stephanie Raven Ann Hargrove May 2015
ACKNOWLEDGMENTS

I would like to thank everyone who has encouraged and supported me during my time in the Doctorate program at the University of Tennessee. In special, I would like to thank my advisor Lee Han for his guidance and support. I also want to thank Dr. Christopher Cherry, Dr. William Seaver and Dr. Xueping Li for serving on my committee.

While some persons helped me with ideas and insight others spent their time to provide valuable insight for my research. In special, I am thankful to Hyeonsup Lim for assisting in field experiments and data management, spending countless hours with code, and providing valuable data which enriched my work and guided me to new directions. I am also grateful to Moraes de Oliveira-Neto for his hard work establishing the license plate matching algorithm and continued support of my research.

I was fortunate to work on a number of research projects sponsored by Tennessee Department of Transportation, Oak Ridge National Laboratory, USDOT’s Southeastern Transportation Program at the University of Tennessee, and FHWA’s Dwight D. Eisenhower Fellowship. I am very grateful for their support and recognition.

I would like to thank everyone in the transportation program: Dr. Fred Wegman, Dr. David Clarke, Deanna Flinchum, Dr. Stephen Richards, Dr. Assad Khattak, Dr. Shashi Nambisan, and my amazing lab mates: Ryan Overton, Qiang Yang, Wei Lu, Jianjiang Yang, Hongtai Yang, Shuguang Ji, Casey Langford, Taekwan Yoon, Sam Moss, Zane Pannell, Bryan Bartnik, Jun Liu, Amber Woodburn, Yang Zhang, Bumjoon Bae, Ziwen Ling, and so on.
ABSTRACT

Probe vehicle and floating traveler data can provide more detailed information about highway use across a roadway network than traditional transportation data sources. However, there are numerous concerns about accuracy, e.g., road user coverage, locational accuracy, and aggregation methods. To address these concerns, evaluations must be completed using a highly accurate data collection method to capture ideal ground truth. For the purpose of this dissertation, license plate recognition (LPR) technology is considered to be the suitable collection method for, and in lieu of, the all ground truth. The data can be obtained using a pair of mobile LPR units to automatically acquire and record license plates at sequential locations along a study route. LPR acquired license plates are then matched automatically by means of a self-learning text-mining algorithm. The algorithm relies on the weighted edit distances of each license plate character to drastically increase the number of correctly matched license plates (97% matching rate with 1% false-positives). To ensure that LPR technology is the best option for the evaluation of real-time data, the license plate matching algorithm requires enhancements to improve matching accuracy and learning speed.

To address the required enhancements, this dissertation evaluates the initial matching process of the algorithm to help increase the speed of learning and matching of license plates. This was completed by updating the starting association matrix- the probability matrix which supplies the similarity measure for the edit distance calculation to determine the likelihood of a match between two associated LPR stations. To further enhance the matching algorithm, the research sought to improve on the procedure for estimating association matrices for problematic LPR stations by deriving an association matrix for a pair of LPR stations. Lastly, the LPR technology and the matching algorithm are employed to capture ground truth and employed to determine the key considerations when evaluating real-time travel times. The overall results are a drastic reduction in learning time, increase in matching accuracy at problematic LPR stations, and strong understating of the key considerations when using LPR as ground truth.
# TABLE OF CONTENTS

CHAPTER 1. INTRODUCTION.................................................................................................................. 1

CHAPTER 2. ENHANCEMENTS TO SELF LEARNING LICENSE PLATE MATCHING ALGORITHM: THE STARTING ASSOCIATION MATRIX ................................................................. 4

Abstract .................................................................................................................................................... 5

2.1 Introduction ........................................................................................................................................ 6

2.2 Background ......................................................................................................................................... 8

2.2.1 Self-Learning License Plate Matching Algorithm ................................................................. 9

2.3 Motivation for Enhancing the Matching Algorithm ................................................................. 11

2.4 Where do we start? ......................................................................................................................... 12

2.4.1 Pre-assigned Values .............................................................................................................. 13

2.4.2 Zero Values .......................................................................................................................... 15

2.4.3 Proposed Starting Association Matrices ............................................................................. 17

2.5 Evaluation Procedure ................................................................................................................... 19

2.5.1 Data Source .......................................................................................................................... 19

2.5.2 Estimation of the Starting Association Matrix .................................................................. 20

2.5.3 Performance Measure .......................................................................................................... 21

2.6 Results .......................................................................................................................................... 22
CHAPTER 3. LICENSE PLATE MATCHING USING DERIVED ASSOCIATION MATRICES

Abstract ............................................................................................................................................. 35

3.1 Introduction ..................................................................................................................................... 35

3.2 Review of Association Matrix ........................................................................................................... 38
  3.2.1 Definition ..................................................................................................................................... 38
  3.2.2 Example of an Ideal Association Matrix ......................................................................................... 40
  3.2.3 Estimation from Truth Matrices ..................................................................................................... 42
  3.2.4 Estimation from Matching Algorithm ............................................................................................. 43

3.3 Deriving an Association Matrix .......................................................................................................... 44
  3.3.1 Proof ........................................................................................................................................... 45
  3.3.2 Theoretical Examination of Deriving Association Matrix ................................................................. 46

3.4 Simulated Experiments ....................................................................................................................... 48
  3.4.1 When should we use the derived association matrix? ................................................................. 49
  3.4.2 Closeness of Association Matrices ................................................................................................. 52
  3.4.3 What is the performance of the derived association matrix? ....................................................... 52
  3.4.4 Performance of LPR matching ..................................................................................................... 54
3.5 Results ............................................................................................................................ 55

3.5.1 When should we use the derived association matrix? ............................................. 55

3.5.2 What is the performance of the derived association matrix? ............................... 57

3.5.3 Summary .................................................................................................................... 58

3.6 Conclusion ..................................................................................................................... 60

References ............................................................................................................................ 61

CHAPTER 4. THE USE OF LICENSE PLATE RECOGNITION TECHNOLOGY FOR GROUND TRUTH DATA COLLECTION ................................................................................. 62

Abstract .................................................................................................................................. 63

4.1 Introduction ....................................................................................................................... 64

4.2 Ground Truth and Data Sources .................................................................................... 65

4.2.1 Ground Truth ............................................................................................................... 66

4.2.2 Traffic Data Sources for Comparison .................................................................... 67

4.2.3 Study Route and Data Collection ............................................................................. 68

4.3 Transforming Travel Datasets ....................................................................................... 71

4.3.1 Determining Comparison Procedure for Data Sources .......................................... 71

4.3.2 Moving Average of Travel Speeds ......................................................................... 73

4.3.3 Conversion of Data from Travel Speed to Travel Time ......................................... 74

4.3.4 Outlier Elimination ................................................................................................. 75
4.5 Data Quality Evaluation Measurements ................................................................. 77
  4.5.1 Visual Investigation of Travel Data ................................................................. 77
  4.5.2 Root Mean Square Error .................................................................................... 78
4.6 Results ..................................................................................................................... 79
  4.6.1 Investigation of Ground Truth Data ................................................................. 79
  4.6.3 Visual Inspection of Each Data Source and the Ground Truth ..................... 81
  4.6.4 Root Mean Square Error Results ..................................................................... 83
4.7 Conclusion ............................................................................................................... 86
References ...................................................................................................................... 88
CHAPTER 5. CONCLUSION ......................................................................................... 92
VITA ............................................................................................................................... 94
LIST OF TABLES

Table 2-1 Proposed Starting Association Matrices................................................................. 18
Table 2-2 Summary of data sample. ......................................................................................... 20
Table 4-1 Summary of selected traffic data technologies and providers. ......................... 69
Table 4-2 Summary of sample size and percentage of RTMS for roadside technology
dataset. ................................................................................................................................. 79
Table 4-3. RMSE of travel speed and travel time for all data sources.............................. 85
LIST OF FIGURES

Figure 2-1 Impact of single match ................................................................. 14

Figure 2-2 Conceptual relationship of pre-assigned value with (a) bias and impact of single match and (b) learning speed for starting association matrix .......................... 16

Figure 2-3 Pre-assigned value performance .................................................. 23

Figure 2-4 False matching rate .................................................................. 25

Figure 2-5 Existing association matrix performance .................................. 27

Figure 2-6 Existing association matrix performance for (a) April 6th, (b) April 7th and (c) May 25th .......................................................... 28

Figure 3-1. Example of unitary association matrix ........................................ 39

Figure 3-2 Frequency tree for a two character example ............................. 41

Figure 3-3. Theoretical Closeness of derived association matrix ............... 48

Figure 3-4 Differences between association matrices with shared read rate of 40% (a), 50% (b), 60% (c) and 70% (d) .......................................................... 56

Figure 3-5 Allowable number of sample size and OCR character accuracy to use derived association matrix ................................................................. 57

Figure 3-6 mrate performance curve for OCR character accuracy ............. 59

Figure 4-1 Study route ............................................................................... 70

Figure 4-2 Two methods for comparing speed data ................................... 72
Figure 4-3 Comparison of travel time and travel speed for segment A to C. .................. 76

Figure 4-4 Travel speed of ground truth data on segment 'A to C'............................... 80

Figure 4-5 Comparison of data sources vs. ground truth data (travel speed).............. 82

Figure 4-6 Distribution of 'travel time of data source vs travel time of ground truth'. ..... 84

Figure 4-7 Inverse of RMSE value. .............................................................................. 86
CHAPTER 1. INTRODUCTION

Travel data are important to the understanding and improvement of the transportation systems. Over the past years the amount of available traffic data has drastically increased. Transportation agencies are using new sources of travel data, i.e., GPS and cellular phones, to understand and improve transportation system performance. This increase in data is fundamentally changing the methods of travel data collections. Traditional methods, e.g., fixed point sensors, must be in place on roadways to capture travel data. New methods rely on floating traveler data sources to provide real-time aggregated data by locating travelers using their cellular phone, Bluetooth or GPS signals.

Floating traveler data may provide more detailed information about road use across a roadway network. However, there are concerns about accuracy, e.g., locational accuracy, road user coverage, and aggregation methods. To address these concerns, evaluations must be completed using a highly accurate data collection method to provide ground truth- absolute data measured in the field. To select the proper ground truth collection method, many things should be considered, such as measurement error and sample size. For the purpose of this dissertation, license plate recognition (LPR) technology is considered the suitable collection method for, and in lieu of, the ground truth. LPR data are obtained using a pair of mobile LPR units to automatically acquire and record license plates at sequential locations along a study route. The acquired license plates are then matched automatically by means of a self-learning text-mining algorithm. The algorithm relies on a travel time window and weighted edit distances for each license plate character to drastically increase the number of correctly matched license plates from a roughly 35% matching rate to 97% matching rate with a 1% false-positives.

The self-learning license plate matching algorithm has been tested and proven satisfactory at matching license plates between two LPR stations for traditional
applications, but requires enhancements to be proficient enough for short-term data collection (equivalently, small sample size), which needs faster learning speed. The proposed improvements are to the initial process of the matching algorithm; more specifically, the starting association matrix. Until the algorithm has learned enough, the starting association matrix lays down all groundwork for determining the edit distances that are a crucial part for establishing whether two license plates are a match. In order to guarantee the fastest and most effective attainment of accurate edit distances, a better starting association matrix must be chosen. The enhancements are aimed at improving the true matching rate, false matching rate and learning speed.

To further enhance the algorithm, the problem of reduced performance when the matching algorithm is used on LPR stations over a great distance (100 plus miles) is addressed. This reduction in performance is due to vehicles no longer travelling within an average travel time window and/or a low sample of vehicles travelling between the two LPR stations. The research proposes using a third LPR station to generate additional information to derive a better association matrix for an existing pair of LPR station, thusly replacing the existing learned association matrix. To evaluate this derived association matrix, two simulations are employed to 1) determine when the matrix should be used and 2) evaluate the overall performance of license plate matching.

By repurposing or multi-purposing technologies such as Bluetooth, GPS, and ubiquitous cellular devices, innovative efforts have seen mixed success in aggregating data from a multitude of devices to derive travel time condition at different geometric resolutions. The challenge, though, is how the performance of these emerging technologies could be measured against a bona fide ground truth (of travel time) when the ground truth can be exceedingly costly and difficult to obtain. To this end, this dissertation uses the license plate recognition and matching algorithm to establish high-accuracy travel time ground truth. The objective was to provide several key considerations for real-time traffic data evaluation for general cases, rather than a definite and specific conclusion.
The key items includes: obtaining reliable ground truth data, transforming and comparing incompatible datasets, and data quality evaluation measurements.

The dissertation is organized in journal article format since each chapter is to be submitted to an academic journal. Following this chapter, the second chapter contains the enhancements to the self-learning license plate matching algorithm that address the initial process of matching and the starting association matrix. The third chapter proposes the use of a derived association matrix as an alternative to the learned association matrix in the matching algorithm for problematic LPR set-ups. The fourth chapter discusses the use of LPR technology and the matching algorithm as ground truth and the key consideration for travel time examination. Conclusions are drawn and future works are recommended in the fifth chapter.
CHAPTER 2. ENHANCEMENTS TO SELF LEARNING LICENSE
PLATE MATCHING ALGORITHM: THE STARTING ASSOCIATION MATRIX
This chapter presents a modified version of a research paper by Stephanie R Hargrove, Hyeonsup Lim, and Lee D. Han.

Abstract

The self-learning license plate matching algorithm, which was developed by Oliveira-Neto et al. (2013), has been tested and proven satisfactory at matching license plates between two LPR stations for traditional applications, but requires enhancements to be proficient enough for short-term data collection (equivalently, small sample size), which needs faster learning speed. The proposed improvements are to the initial process of the matching algorithm; more specifically, the starting association matrix. Until the algorithm has learned enough, the starting association matrix lays down all groundwork for determining the edit distances that are a crucial part for establishing whether two license plates are a match. In order to guarantee the fastest and most effective attainment of accurate edit distances, the better starting association matrix must be chosen. The enhancements are aimed at improving the true matching rate, false matching rate and learning speed.

Twelve potential starting association matrices were evaluated. The results reveal that several starting association matrices helped the algorithm perform much better compared to an identity matrix and the selection of a starting association matrix is dependent on the application of the collected travel information. For example, applications requiring minimal false matches may require a matrix with element value of 100 in the main diagonal and 1 in the off-diagonals. While applications that are short on time and need a high number of matches with small error may require an existing association matrix created from a large volume of matches. The top starting association matrices, after one learning iteration, achieved matching rates of 97% with 1.3% false
matching for high learning speed (25 license plates) and 0.8% false matching for a slightly slower learning speed (150 plates).

2.1 Introduction

License Plate Recognition (LPR) technology is a popular data collection method for applications relying on vehicle matching, such as automated speed enforcement, vehicle tracking, and access control. In a perfect world, LPR technology would capture 100% percent of all vehicles within the field of view and read all license plates perfectly, but the technology, though mature, is still flawed. Depending on the model of camera, set-up (mobile or stationary), location (side of road or overpass), on-site calibration, diversity of captured license plates, and numerous other factors, the typical read rate of a LPR camera rarely surpasses 80% and is commonly around 60% or less [1-3]. The portion of license plates that can be correctly matched can fall as low as 35% or below. Studies have tried improving LPR technology accuracy by looking at hardware or OCR engines [4-7], but such improvements cannot address uncontrollable conditions that affect the overall technology performance.

There are three facets that can affect the overall performance of matching LPR-captured license plates. The first two facets are the capturing and reading of a license plate; together they account for the total performance of LPR technology. The capture rate (rate of successful plate recognitions in the field of view) can be affected by both environmental conditions and parameters relating to the camera’s hardware, installation, or on-site calibration. The read rate (rate of correctly interpreting an entire license plate) is based upon the performance of the OCR engine. The accuracy of these facets is commonly uncontrollable by the data output user and dependent on the performance of the LPR technology.

The third facet is the process of matching a license plate that has been captured at two LPR stations. The matching of a license plate text is different from other OCR text
matching problems. License plates have no available libraries or references of truth values to match the target text. Thus, it is impossible to know the true character of a recognized license plate without visual identification.

For each recognized license plate captured at one LPR station, there is a set of potential license plate matches at another LPR station. Many transportation agencies address these shortcomings by relying on manpower to improve the accuracy of the matching process by modifying incorrectly captured license plates one by one; thus, making the license plate capturing process more costly, time consuming, and undesirable to potential users[8-10]. Some users have added data processing functions, such as travel time window constraints to the matching process to aid in the reduction of outliers and false matches [11, 12].

Obviously, when a plate is read correctly at both stations, a match is declared. Though, when just one character is incorrectly recognized a match cannot be declared. Even though LPR has failed to read a plate correctly, the system still has returned mostly correct individual character recognitions. Comparing an imperfectly read plate against another plate, one could still make a judgment on whether it is a match based on the similarity. The self-learning license plate matching algorithm proposed by Oliveira-Neto et al. (2012, 2013) applies a text-mining technique called edit distance. This measures how close two plate strings (sequence of characters) are from each other based on weight functions to compare each individual pair of characters.[2, 3, 13].

The current self-learning license plate matching algorithm has been tested and proven satisfactory at plate matching for traditional transportation applications. But improvements are required to ensure the algorithm is proficient enough to improve learning speed of the algorithm. To ensure a proficient matching algorithm, matching must be performed at high accuracy levels under all conditions 24/7.

The proposed enhancements to the algorithm will focus on the initial matching process; more specifically, the starting association matrix. The starting association matrix is
made based on a prior believed distribution, and is updated as it learns. Thus, the starting association matrix lays down all groundwork for determining the edit distances that are a crucial part for establishing the similarity between two license plates. In order to guarantee the quickest and most effective attainment of accurate edit distances, the best starting association matrix must be identified. The enhancements are aimed at improving the true matching rate, false matching rate and learning speed.

2.2 Background

In 2009, Oliveira-Neto et al. made a license plate matching algorithm that employed a traditional Levenshtein edit distance (ED) technique [14] to improve the matching efficiency of imperfectly read plates [13]. Combining the ED technique with a travel time window to help reduce outliers, the proposed algorithm was able to accurately match about 80% of license plates with 3% false matches. The algorithm then received additional updating to include a generalized edit distance (GED) technique with a weight function, an association matrix containing the probabilities of matches, and a self-learning procedure [2, 3]. This algorithm achieved matching rates of 95-96% with false matching around 1%, but requiring the learning time of multiple days [3]. The following section discusses the procedure of matching two license plate strings using the edit distance technique.

Edit distance is a text mining technique used to estimate the similarity between two strings. For two license plate strings \( X \) and \( Y \), the edit distance determines the minimum editing cost to convert one license plate string to another by combining a sequence of individual character comparisons to determine the degree of similarity. Suppose there are two sequential LPR stations \((g, h)\) that read the same plate as \( X = "ABC123" \) and \( Y = "A8C1Z3" \). Applying the Levenshtein ED technique from the original algorithm to the potential match, two edit operations (the substitutions “B” to “8” and “2” to “Z”) would be required to convert \( X \) to \( Y \). If the edit distance cost of two falls below a certain assigned threshold, then \( X \) and \( Y \) will be recognized as a match.
The OCR engine of LPR technology commonly has trouble separating ambiguous characters (1/I, 0/O, 2/Z, 8/B, and so on). With this knowledge, one would be able to make a more educated decision as to whether $X$ and $Y$ from the previous example would be a match or not. The generalized edit distance does just that by assigning different weights to the edit operations as a function of the character.

These weighted edit distances are calculated directly using a set of matched pairs from two associated LPR stations. By incorporating the individual character matches from the matched set, one can derive an association matrix that represents the likelihood that character $a$ reported by LPR station $g$ is reported as character $b$ by LPR station $h$.

### 2.2.1 Self-Learning License Plate Matching Algorithm

The association matrix between two LPR stations is a $N^*$ by $N^*$ square matrix whose elements are the conditional probability $p(b|a)$ with a character reading $b$, $b \in N^*$, in station $h$ or $a$, $a \in N^*$, in station $g$. The set $N$ is the set of possible alpha-numeric characters{$0, 1, 2, ..., 9, A, B, ..., Z, \lambda$}, where $\lambda$ is the null character. The null character can be more than just an unknown symbol, but also represent deletions and insertions in the plate string text. When observing an association matrix, diagonal elements represent the same character being recognized at both LPR stations, while all other matches are present in the off-diagonal elements. Each row refers to a given character’s recognition at station $g$, with columns referring to the recognition at station $h$. The elements of an association matrix are represented in both unitary (counts of matches) and probabilistic (conditional probability) values. To estimate the $p(b|a)$ of matrix A, the following equation is used to convert the unitary values into the conditional probability values:

$$p(b|a) = \frac{\rho_{ab}}{\rho_a}, \quad (2-1)$$

where $\rho_{ab}$ represents the number of times character $b$, $b \in N^*$ is associated to character $a$, $a \in N^*$ in the unitary association matrix and $\rho_a$ is the number of times character $a$ has
been matched at LPR station $g$. As Oliveira-Neto et al. (2013) stated the conditional probability elements of matrix $A$ are then used in the calculation of a weight function in the following equation:

$$\gamma(a \rightarrow b) = \log\left(\frac{1}{p(b|a)}\right)$$

(2-2)

where $\gamma$ is the weight function relating the corresponding pair of characters $(a, b)$ such that $\gamma(a \rightarrow b)$ is a non-negative real number assigned to each character's edit operation. These edit operations are then summed to determine their respective plate string's edit distance.

The process of estimating the first learned association matrix is the key to achieve the higher matching performance and faster learning speed. The first learned matrix $A$ is estimated by a set of starting matches $M_s$ that has been determined using a time-window constraint, a starting association matrix $S$, and the ED technique with editing weights calculated using matrix $S$. The set of starting matches $M_s$ is a set of license plates matched by the matching algorithm and the starting association matrix for a given period of time of operation or a fixed number of license plates. When a pair of characters is matched in $M_s$, they are respectively assigned a unitary value of ‘1’ or ‘0’ for a match or non-match. These assignments are then added to matrix $S$ to create the first learned association matrix.

In order to obtain a set $M_s$, a starting association matrix $S$ must be given. Matrix $S$ is a $N^*$ by $N^*$ square matrix whose elements are unitary values. The previous matching algorithms all use an identity matrix $I$ (a $N^*$ by $N^*$ square matrix with ones along the diagonal elements and zeros for all the off-diagonal elements) for matrix $S$. Matrix $I$, in terms of performance, would mean that the OCR engine performs perfectly and every match is a true match. For heuristic learning purposes, matrix $I$ has performed reasonably when the matching algorithm is learned enough, but there may be better options to obtain higher matching performance and faster learning speed.
Once the first learned association matrix is created, the self-learning process continues by repeatedly applying the matching process to progressively find better association matrices. In other words, each iteration $k$ is expected to estimate a better matrix $A_k$ directly from a matching set $M_k$, which is obtained from the previous matrix $A_{k-1}$. The matching algorithm will then continue to learn until the difference between two successive estimations falls below a pre-assigned threshold.

The matching algorithm takes a heuristic approach in determining the elements within an association matrix. Accordingly, the estimated association matrix will never be the true (ideal) association matrix and the purpose is to find a matrix that is sufficient for the immediate goals. An ideal-like association matrix is obtainable but would require an environment with little variability for the LPR stations and a large set of matched license plates, or a matching algorithm with supreme learning speed and accuracy that can constantly adapt to changing conditions.

### 2.3 Motivation for Enhancing the Matching Algorithm

The surrounding conditions of a LPR station are ever-changing. Various characteristics are rarely alike for any instance when a plate is captured: speed, viewing angle, lateral location in a lane, occlusion, plate colors, fonts, plate orientation, and plate design vary widely between vehicles and between LPR stations. The environment is also constantly changing, be it weather (rain, snow, fog, clear) or lighting condition (day, night, shadow, glare from sun). These all are uncontrollable and ever changing outside factors affecting the LPR technology accuracy. In order to address these factors, the association matrix needs to be continually updated. Once an association matrix is created it is used for all matching until it is updated. The current matching algorithm has a learning period of one day; meaning the association matrix is updated every evening. An association matrix must be constantly updated in order to reflect the changing conditions of real-time data. A starting association matrix with the smallest possible set of starting matches is expected to best reflect these changing conditions as the new set of matching will be
accounted more. However, the performance of the matching algorithm will be relying overly on first matches. In other words, the performance may get much lower when the first matches are false. Accordingly, the starting association matrix $S$ is also related to the learning speed, which here is defined as the number of matches required to acquire a certain true matching rate. For instance, a poor starting association matrix may require a match set of 500 license plates to achieve 90% true matching rate while a well-designed starting association matrix requires only 25 or less. By matching with an ‘optimal’ or ‘near-optimal’ starting association matrix, the required set of starting matches could be drastically reduced; thus increasing the learning speed.

### 2.4 Where do we start?

For an extreme starting point, assume matrix $S$ is a zero matrix with one learned character match, and then the single matched character will be the only information available to determine the next character match. Consequently, if a small amount of matches have a large impact on the probability of character association, then incorrect matches could result in an association matrix with a biased association. The bias of an association matrix might not be an irreparable problem, but will require additional learning time and a larger set of starting matches to correct.

To examine the impact of single match on determining the next match, the following equation is used:

$$\Delta P = \frac{(\rho_{ab} + 1)}{(\rho_a + 1)} - \frac{\rho_{ab}}{\rho_a},$$

(2-3)

where $\Delta P$ is the change in conditional probability $p(b|a)$ when a single match is added to an association matrix, with $\rho_{ab}$ and $\rho_a$ as defined before.

Figure 2-1 displays the impact of a single match on the conditional probability $p(b|a)$ for multiple scenarios. For the scenario of $\rho_{ab} = 0$ and $\rho_a = 1$, character $a$ has been
matched once at LPR station $g$ and never with character $b$ from LPR station $h$; so, $p(b|a)$ is equal to zero while the one match for character $a$ is equal to one. When the next match for character $a$ occurs, the probability $p(b|a)$ will go from zero to 0.5, as seen in Figure 2-1. If the learning process was stopped at this point, the result would be two pairs of matches with very small (weighted) edit distances, while the other 35 potential matches gain the infinite (weighted) edit distance saying they are impossible cases. Now for the scenario of $\rho_{ab} = 0$ and $\rho_a = 10$, $p(b|a)$ drops down to 0.09, meaning the likelihood has reduced by 0.41 with the addition of only 9 characters.

**2.4.1 Pre-assigned Values**

It is conceived that to alleviate the impact of a single match, larger numbers (larger $\rho_{ab}$ and $\rho_a$, but with the same likelihood) should be assigned to the elements of the starting association matrix. However, too large pre-assigned value can still produce additional issues of biased results. For an extreme case, the new match set will not have any impact at all if the pre-assigned value is infinite, which means the matching algorithm purely relies on the starting association matrix without any consideration of learning. Therefore, the pre-assigned value was included as a main control parameter to generate a starting association matrix.

The two methods of allocating the pre-assigned values to the elements of a matrix considered are 1) the multiplication of a value $\alpha$, where $0 \leq \alpha$, and 2) the addition of a uniform matrix $U$ (a $N^+ \times N^+$ square matrix with elements containing uniform values) to every cell. In Figure 2-2, the conceptual relationship of pre-assigned values and the impact of single match, bias, and learning speed are demonstrated. The two performance lines/curves represent two different types of potential starting association matrices with pre-assigned values.
The first type (dotted line) represents the multiplication of an identity matrix $I$ by value of $\alpha$. The resulting matrix $S$ would contain a main diagonal cell value of $\alpha$ and an off-diagonal value of 0. Unless this matrix $S$ is very close to the ideal association matrix, which is unlikely, it is assumed that every match always has the same character in both stations, meaning the cameras read all characters perfectly. If the assumption is wrong, which is realistic, the large $\alpha$ will make the starting association matrix more biased, requiring a larger number of matches to get the ideal association matrix, as seen in Figure 2-2(a). For example, suppose that 25% of the character pairs read at two stations are true matches. If $\alpha = 750$, it would take at least 250 additional matches to return to a realistic distribution of matches (this is only when all of them are a pair of different characters). Actually, it will require much more than 250 because the starting
association matrix is likely to capture the same characters as a match set. The learning
time required to obtain 250 additional matches for an individual character is large when
considering a starting set size of 100 license plates contains on average 600
characters. Therefore, the relationship between learning speed and pre-assigned values
becomes important in determining the starting association matrix. As seen in Figure 2-2
(b), if the pre-assigned value α is too low or too high it will negatively affect the learning
speed.

For the second type (solid line), a uniform matrix U is added to an identity matrix
multiplied by value α. By having the uniform matrix add a value of one, the off-diagonals
now contain a unitary value representing one match; therefore, decreasing the impact of
one additional match for the staring association matrix. Furthermore, the second type
allows that different characters can also be true match sets, which is comparable with
the first type. This difference could also improve the calculation of matches’ edit
distance values; this is discussed in further detail in Section 2.4.2.

It is hard to say what the optimal value of α will be along the matrix’s performance
line/curve within Figure 2.2(a) and (b). In order to determine the pre-assigned value’s
true relationship with learning speed, impact of single match and bias; multiple pre-
assigned values should be evaluated.

### 2.4.2 Zero Values

The first examination of starting association matrices ran parallel to a performance
analysis of the ED threshold. The Equation 2-2 shows the weighted edit distance
calculation by using the logarithm function, which yields much larger ED values when
the likelihood p(b|a) is small. However, the ED becomes infinite when p(b|a)
approaches zero. Therefore, the algorithm never determines the matched pairs with
zero cells in the association matrix as a true match set because their ED is infinite. All
the more, if the infinite ED does not allow the match of the zero cells when performing
the learning process, a zero cell may never become anything other than zero within the association matrix.

Figure 2-2 Conceptual relationship of pre-assigned value with (a) bias and impact of single match and (b) learning speed for starting association matrix.
A zero value within an association matrix does not only mean “no match,” but also a true missing value. There are then two reasons for a true missing value; either the value is missing at random or the infinite ED value is preventing the learning process from changing the value from zero[15]. For a small starting matched set, the likelihood of a value being missing at random is high.

Based on prior knowledge of possible match errors from LPR technology, an assumption was made that all zero values could be values missing at random, i.e., a possible match. If the assumption is true, then replacing the zero cells would not lead to bias, and a correct edit distance would be calculated; resulting in higher matching rates and overall learning. To test this assumption, a portion of the proposed starting association matrices was pre-assigned the uniform matrix U with a value of 1; consequently, removing all zero cells.

2.4.3 Proposed Starting Association Matrices

The proposed starting association matrices are broken into two categories: ‘no prior learning’ and ‘prior learning.’ The matrices evaluated with no prior learning are the identity matrix and other identity matrices that have been altered using pre-assigned values and uniform matrices. The four matrices with prior learning include three matrices corresponding to the sample data and one from a separate sample. Additional starting association matrices were proposed to evaluate the pre-assigned value and uniform matrix theory discussed in Section 2.4.1. The twelve evaluated starting association matrices types are listed in Table 2-1.

2.4.3.1 Matrices with No Prior Learning

Rudimental matrices are evaluated to gain a better understanding of overall matching performances. The identity matrix (ID) is the starting association matrix commonly used for all past matching algorithms. This matrix has provided satisfactory results, but we assume that there is still room for improvement. By applying the theory of pre-assigned
values and uniform matrices discussed in Section 2.4.1, four additional versions of the identity matrix are created. Matrix type $I\alpha$ represents the identity matrix multiplied by the pre-assigned values of $\alpha = 10, 100 & 1,000$. Matrix type $I\alpha U$ represents the addition of the uniform matrix $U$ with the value of one to matrix type $I\alpha$.

Table 2-1 Proposed Starting Association Matrices.

<table>
<thead>
<tr>
<th>Notation</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>Identity Matrix- on diagonal cells are equal to 1 and all off diagonals cells are zero.</td>
</tr>
<tr>
<td>$I\alpha$</td>
<td>Identity Matrix $\cdot \alpha$ - all cells within the identity matrix are multiplied by $\alpha = 10, 100, \text{ and } 1000$.</td>
</tr>
<tr>
<td>$I\alpha U$</td>
<td>Identity Matrix $\cdot \alpha + U$ - all cells within the identity matrix are multiplied by $\alpha = 10, 100 \text{ and } 1000$ then matrix $U$ is added.</td>
</tr>
<tr>
<td>A1</td>
<td>10-18-13 Association Matrix</td>
</tr>
<tr>
<td>U1</td>
<td>10-18-13 Association Matrix + $U$</td>
</tr>
<tr>
<td>U2</td>
<td>04-06-2010 Association Matrix + $U$</td>
</tr>
<tr>
<td>U3</td>
<td>04-07-2010 Association Matrix + $U$</td>
</tr>
<tr>
<td>U4</td>
<td>05-25-2010 Association Matrix + $U$</td>
</tr>
</tbody>
</table>

2.4.3.2 Matrices with Prior Learning

Existing association matrices can be very valuable to the matching algorithm. Instead of beginning with a ‘clean’ starting association, it may be more constructive to just update an existing association matrix. If all OCR engines have similar patterns of errors when capturing characters, then using an existing association matrix could increase the learning speed by using previously determined (weighted) edit distances. However, this could be inappropriate where the surrounding conditions under matching license plates
were significantly different (or changed). Consequently, leading to an increase in false matches, reduction in matching rate and large upsurge in the learning time is needed to re-associate the LPR stations.

Matrix $A1$ comes from a temporary set-up on October 18th, 2013 for a one mile stretch of the Interstate system in Nashville, TN. During the 12 hour set-up, LPR technology was able to match 1,672 license plates while the current matching algorithm matched 2,606. A very strong association matrix was estimated due to the large number of true matches. The purpose of choosing matrix $A1$ is to determine if a strong association matrix, even from a separate set-up with a different version of LPR cameras, could possibly be a universal starting association matrix.

This is based on the assumption that all OCR engines will experience the same error; therefore, resulting in a common (weighted) edit distance for all character matches. Matrix $U1$, the addition of a uniform matrix to matrix $A1$, was also evaluated to determine if the zero values (discussed in Section 2.4.2) played a role in the performance of the matching rate for an existing matrix.

The existing matrices $U2$, $U3$ and $U4$ are corresponding association matrices for the three days (April 6th, April 7th, and May 25th) of sample data used in the evaluation. These association matrices were all created using the matching algorithm proposed in [3] with the addition of a uniform matrix $U$ with a value of one. The purpose of evaluating these existing matrices is to determine whether each existing association matrix will outperform all other existing starting association matrices during their respective day.

### 2.5 Evaluation Procedure

#### 2.5.1 Data Source

Two LPR cameras were mounted 3 miles apart on variable message boards on a stretch of the Interstate system to monitor the front license plates of trucks 24/7 for the
year 2010. A survey period of 3 days (April 6\textsuperscript{th} and 7\textsuperscript{th} and May 25\textsuperscript{th}) was selected to assess the performance of the proposed starting association matrices. These days were selected based on the availability of a ground truth set to validate the performance of each starting association matrix and for comparison of performance between enhanced algorithm and previous algorithm. Table 2-2 shows a summary of the captured samples containing the number of captured license plates, the license plate reading rate, the number of true matches, the number of potential matches, and the exact matching rate for the respective days and LPR stations.

### Table 2-2 Summary of data sample.

<table>
<thead>
<tr>
<th>Date: Station</th>
<th>April 6th</th>
<th>April 7th</th>
<th>May 25th</th>
<th>Overall</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>g</td>
<td>h</td>
<td>g</td>
<td>h</td>
</tr>
<tr>
<td>Captured License Plates</td>
<td>2415</td>
<td>4265</td>
<td>2508</td>
<td>4416</td>
</tr>
<tr>
<td>License Plate Reading Rate</td>
<td>35%</td>
<td>56%</td>
<td>36%</td>
<td>54%</td>
</tr>
<tr>
<td>True Matches</td>
<td>219</td>
<td>245</td>
<td>221</td>
<td>685</td>
</tr>
<tr>
<td>Potential Matches</td>
<td>598</td>
<td>841</td>
<td>698</td>
<td>2137</td>
</tr>
<tr>
<td>Exact Matching Rate</td>
<td>37%</td>
<td>29%</td>
<td>32%</td>
<td>32%</td>
</tr>
</tbody>
</table>

#### 2.5.2 Estimation of the Starting Association Matrix

The goal of the evaluation process is to only examine the performance of the starting association matrix and not the overall performance of the whole matching algorithm. Therefore, only the first iteration of the learning process is performed. The evaluation process varies slightly from the original matching algorithm; in that, the set of starting matches is determined by a number of matched license plates rather than a period of time, to see the needed size of starting matches to achieve a satisfactory association matrix. Also, all iterations of learning are completed using edit distance with the weight
function. These changes are used in determining the process of estimating the initial association matrix and matching performance using the proposed starting association matrices, and can be seen as follows:

1. Apply the calculated travel time window and matching procedure to find a set $M_s$ from a sample of LPR data with the initial settings:
   a. Define the starting association matrix $S$;
   b. Calculate ED $\delta(X,Y)$ using matrix $S$ to calculate all costs;
   c. Use an assignment of 0 or 1 for non-matched and matched characters, respectively;
   d. Set a threshold $\tau$ such that if $\delta(X,Y) \leq \tau$ the string outcomes $X$ and $Y$ are classified as a match;
   e. All matches are assigned to set $M_s$.
2. From the set of matched plates obtained in step 1, tabulate the matched character occurrences and add to the starting association matrix to create an updated unitary association matrix $A$.
3. Compute the probability elements for matrix $A$ using Equation (2-1); therefore, $A = p(b|a)$.
4. Using matrix $A$, perform the matching procedure for the entire sample of LPR data. The result is a full matched set $M$ of the LPR data sample.

2.5.3 Performance Measure

To measure the performance of each starting association matrix, the learning speed and matching accuracy will be evaluated. To measure the speed of learning, the set of starting matches is used as the benchmark of time and speed. Therefore, the smaller the starting matches, the faster the learning speed. The starting sample size is examined in increments of 25 matched license plates (roughly 150 characters) in a range of 25 to 550. By evaluating in terms of license plates rather than time, a better
understanding of the time it takes to estimate the association matrix is achieved, by using the known volume on the examined roadway.

After the starting matched set has been matched and the association matrix has been updated in step 1 through 3, the matching is performed for the entire dataset in step 4. The performance measures are defined in terms of positive matching rate (pmr), false matching rate (fmr) and matching rate (mrate) calculated based on the matching results, as seen in the following equations:

\[
\text{pmr} = (|M| - |F|)/|M'|, \\
\text{fmr} = |F|/|M|, \\
\text{mrate} = \frac{2}{\frac{1}{1 - \text{fmr}} + \frac{1}{\text{pmr}}}
\]

where \( F = \{(X_m, X_n) | X_m, X_n \in M, X_m' \neq X_n'\} \) is the set of false matches from \( M \), \( M' = \{(X_m, X_n); X_m' = X_n'\} \) is the total set of possible matches, with \( M \) as defined before. The mrate is the harmonic mean of the fraction of matches that are the true matches \((1 - \text{fmr})\) and the fraction of positive matches that are identified \((\text{pmr})\). This mean calculation is commonly used for the comparison of rates.

## 2.6 Results

The starting association matrices in Table 2-1 are evaluated using the starting procedure in Section 2.6.2 to determine the performance at incremental starting sets. All figures, unless otherwise noted, contain results for April 7th, 2010. This day was chosen based on the largest number of possible matches. Figure 2-3 shows the curves of the matching rate by starting sample size for the pre-assigned values. The traditionally used identity matrix’s (ID) performance curve shows a low matching rate (0.735) in the
beginning, though as the starting set size increases so does the matching at a consistent rate. The performance curves of the ID matrix clearly demonstrates that as the starting matches sample increases so does the performances of the matrix. It is likely that even though these starting matrices perform slower than the others, with time and a larger starting set, a better association matrix ($mrate = 0.95$) could be achieved by the identity matrix.

![Figure 2-3 Pre-assigned value performance.](image)

Interestingly, Figure 2-3 demonstrates that the learning speed will get slower if the pre-assigned value $\alpha$ is too low or too high. As expected, the learning speed performance
for the values 1 (identity matrix), 10, 100 and 1000 go respectively from moderate to better to moderate and then low; thus, resembling the theory presented in Figure 2-2 (b). This happens because as the pre-assigned value is initially increased, the impact of a single match is decreased. As the value continues to increase, so does the bias towards all true matches, resulting in decreased learning. It is not possible to determine the optimal pre-assigned value from the results. Though, the value range for this LPR camera can be assumed as 1 to 100.

More interestingly, Figure 2-3 shows that the learning speed of an identity matrix multiplied by 1000 was the worst performing matrix in terms of the pre-assigned matrices. When the uniform matrix was added to the matrix, the initial matching performance improved greatly (from 0.697 to 0.907). Similar conditions were also observed for the pre-assigned value of 10 and 100. The addition of the uniform matrix and 36 matches to frequency $\rho_a$ decreased the bias towards the new match by encouraging the formulation of a more realistic edit distance value.

Figure 2-4 shows the curves for the false matching rate by starting sample size for all proposed association matrices. According to Figure 2-4 for matrix types $\alpha$, the pre-assigned value of 1000 has the best false rates, quickly followed by $\alpha U$ with the same pre-assigned value. As expected, the bias nature of a large pre-assigned value not only reduced the learning speed (seen in Figure 2-2), but decreased the impact of a single matches (seen by the $fmr$ values of zero and 0.2%). This means that a large pre-assigned value aids the learning process in the reduction of false matches.

An unexpected result was the closeness of performance of the identity matrix ($ID$) and the matrix $\alpha$ with $\alpha = 10$ and 100. The largest variance seen in the positive matching rate of the three matrices is 0.008. Though, as seen in Figure 2-5, the false matching rate fluctuates as the main diagonal increases for the matrices, with $\alpha = 100$ achieving the lowest $fmr$. Similar performance is also seen in matrix $\alpha U$ with $\alpha = 10$ and 100.
Some common trends in the pre-assigned values are 1) as the value $\alpha$ increases, the overall matching rate and false matching rate decreases, and 2) the addition of the uniform matrix results in better matching performance and learning speed, but an increase in false matches.

![Figure 2-4 False matching rate.](image)

Overall, the best performing matrix with no prior learning was matrix $I\alpha U$ where $\alpha = 10$, resulting in a peak $mrate$ of 0.97, $pmr$ of 0.953 and $fmr$ of 1.1%. A close second was matrix $I\alpha U$ with $\alpha = 100$. In terms of minimum false matching, while still maintaining a
high initial learning speed, matrix $kU$ where $\alpha = 1000$ performed the best with $fmr$ equal to 0.2% and an overall $mrate$ of 0.957.

Figure 2-5 shows the curves of the matching rate by sample size for the existing association matrices $A1$ and $U1$. By adding a uniform matrix to the existing matrix $A1$, initial matching performance improved from 0.958 to 0.990. Thus, the edit distance formation is improved, allowing more matches at higher rate. The downside is the increase of false matches, caused by the starting association matrix assuming that all possible matches have been observed. The one existing matrix without the addition of the uniform matrix ($A1$), has the lowest false matching rate ($1.0 - 1.2\%$) performance of all the proposed matrices with prior learning. As expected, both matrices $A1$ and $U1$ prove that an association matrix from a separate day, set-up and type of LPR camera can be used potentially as a ‘universal’ starting association matrix.

Figure 2-6 shows the performance curves of the matching rate by sample size for existing association matrices based on the study sample days. In Figure 2-5, the following matrix refers to the following study day: (a) matrix $U2$ from April 6th, (b) matrix $U3$ from April 7th and (c) matrix $U4$ for May 25th. Based on visual inspection, most frequently matrix $U1$ outperformed the other existing matrices. This is important, because matrix $U1$ is the one matrix without prior learning of the study sample. Thus, the results show that better performance comes from an existing association matrix with a large amount of matches and learning, than from the respective LPR station with a small number of potential matches.
Figure 2-5 Existing association matrix performance.

Figure 2-6 demonstrates the study sample’s association matrices did not exhibit any pattern of outperforming other existing matrices during their respective day. Matrices $U_2, U_3$ and $U_4$’s performance curves did remain close in performance; though, experienced variation and a minimal increase in learning. This may be due to limited changes in the conditions of the LPR station.

Oliveira-Neto et al. examined the performance of the self-learning matching algorithm, where matrix $S = I$, against our same data source (we only used 3 of the 5 days they used) in [3]. It is important to note that, all of their results have gone through multiple self-learning iterations while ours have only been through one with varying starting sample sizes. Three of the existing association matrices ($U_2, U_3, U_4$) should be examined with extra criticism, since they are the estimated association matrices created by this study. Also, their starting sample is not based on the number of plates, but a period of time. Some of their notable results are as follows.
Figure 2-6 Existing association matrix performance for (a) April 6\textsuperscript{th}, (b) April 7\textsuperscript{th} and (c) May 25\textsuperscript{th}.
For one day of learning, their pmr reaches about 90%. Our identity matrix (ID) also obtained a 90% positive matching rate, but required 525 starting matches to achieve that rate for April 7th. This starting size may be comparable to their one day of learning, when considering the maximum starting set size achieved for April 7th was 543. For a week of learning, their matching rate reached over 94%. In just one learning iteration, seven of our proposed starting association matches (kα with α = 10, kαU with α = 100, A1, U1, U2, U3, U4) reached pmr values over 95% with varying starting set sizes (4 with only 25 license plate matches). Eventually, the previous matching algorithm’s pmr achieved over 95%. Matrices U1, U2, U3 and U4, all existing association matrix with the added uniform matrix, achieved a pmr 0.95 or higher with a starting set size of only 25 plates.

Clearly, our enhanced algorithm outperformed the current algorithm. By changing the starting association matrix, the positive matching results remain close, but the learning time is greatly reduced from over a week to just a single learning iteration with a starting set of 25 license plates.

In summary, the enhanced algorithm is fast and adaptive. The current algorithm simply takes too long (a week) to achieve a high matching rate. The enhanced algorithm is far superior in terms of learning speed. The versatility of the starting association matrices leaves the user with the option of choosing a matrix that fits their application needs. For example, applications requiring minimal false matches may use a matrix with element value of 1000 in the main diagonal and 1 in the off-diagonals. While applications that are short on time and need a high number of matches with small error, may use an existing association matrix created from a large volume of matches.

2.7 Conclusion

Our proposed enhancements improve the license plate matching rate and learning algorithm significantly without any modification to the LPR mechanism itself. The
enhanced algorithm can be applied to any LPR system to capture conditional association matrices based on the time of day, characteristics of traffic, and weather. With the small required starting set sample of 25 license plates the time to update the association matrix would only be dependent on the volume of traffic. As shown in this paper, the top starting association matrices, after one learning iteration, achieved matching rates of 97% with 1.3% false matching for high learning speed (25 license plates) and 0.8% false matching for a slightly slower learning speed (125 plates). This is an astounding improvement for LPR stations that had overall reading rates of 34% and 55% that could only achieve an overall matching rate of 32%.

In terms of applications, the enhanced algorithm can be applied to a number of real-time tracking applications to improve matching and reduce the cost of manually identifying correct matches, including traveler information systems, active traffic management, vehicle tolling, and automated speed monitoring and enforcement, etc. Temporary set-ups for data collection studies, e.g. evaluation of real-time travel speed or origin-destination, can also benefit from the enhanced algorithm’s reduced learning time. The versatility of the starting association matrices allows LPR users to choose the best matrix that fits their application needs.

As for further research, we know that the enhanced algorithm only needs 25 matched license plates to update the association matrix, but when it should be updated must be examined. This will require additional examination of the OCR engine’s read performance rate. By understanding what conditions have the highest impact on the OCR’s read rate, the association matrix can address each one. In contrast, the best procedure may be to just update the association matrix after every 25 matches. This additional information would ensure that the best association matrix will be used for each arising condition.

The current matching algorithm highly relies on the calculation of a character’s edit distance. Further research must be done to address the infinite error that occurs when
the edit function experiences the $p(b|a)$ equals to zero. The addition of the uniform matrix to the starting association matrix should only be a temporary fix until a new weighted edit formulation is determined. Though the addition may prove useful in improving the matching rate, by removing all zeros from the matrix, the learning process now believes that every match is plausible - which is not true. Exploration into other text-mining fields may help the formulation process of a weight function and other edit distance methods.
References


8. Shawn Turner, James R., Mike Fontaine, and Brian Smith, Guidelines for Evaluating the Accuracy of Travel Times and Speed Data, 2011.


CHAPTER 3. LICENSE PLATE MATCHING USING DERIVED ASSOCIATION MATRICES
This chapter presents a modified version of a research paper by Stephanie R Hargrove, Hyeonsup Lim, and Lee D. Han.

**Abstract**

To perform the post-processing matching of license plates between two license plate recognition (LPR) stations a self-learning matching algorithm is employed. The key component of this algorithm is an association matrix that represents the license plate matches between two LPR stations that is estimated directly from a set of matched character pairs. The matching algorithm’s performance decreases as the distance between two LPR stations increases. This is due to vehicles no longer travelling within an average travel time window and/or a low sample of vehicles travelling between the two LPR stations. This paper proposes using a third LPR station to generate additional information to derive a better association matrix for an existing pair of LPR station, thusly replacing the existing learned association matrix. To evaluate this derived association matrix, we employ two simulations to 1) determine when the matrix should be used and 2) evaluate the overall performance of license plate matching.

**3.1 Introduction**

License plate recognition (LPR) technology has been widely applied in numerous transportation applications including automated speed and law enforcement, vehicle tracking, and vehicle tolling. All of these applications require LPR technology to match a license plate at two locations. In order to do so without additional post-processing, each license plate string (sequence of characters) must be identified correctly to declare a match. If just one character is misread then a match cannot be declared without any additional post processing.
License plate recognition technology uses optical character recognition (OCR) engines to identify the text strings of license plates. The matching of OCR recognized license plates is far more complicated than the matching of traditional OCR text, such as text from books. The matching of traditional OCR text has the benefit of readily available dictionaries containing a finite number of vocabulary or strings, context to help determine the likely meaning of the word, and a standard syntax for all characters. On the other hand, license plates strings almost never have meanings and multiple potential syntaxes; one cannot even be certain a plate string was recognized correctly by the LPR algorithm without manual verification. Though, not all is lost, because the process of matching two license plate strings can go beyond looking at the string as a whole and instead use the sequence of comparisons of individual characters.

Consider, for example, license plate strings $X$ and $Y$ are read at two LPR stations, with the result $X$="ABC123" and $Y$="A8C1Z3". By comparing these two strings, one can make a guess as to whether or not $X$ and $Y$ are a match. To perform the matching of these strings, Oliveira-Neto et al. [1] proposed using the Levenshtein edit distance (ED) technique[2]. By applying the Levenshtein ED technique to the example, two fundamental operations (the substitutions "B" to "8" and "2" to "Z") are required to convert $X$ to $Y$; hence, the total edit distance is two. If the edit distance between the two strings falls below an assigned threshold value, a “match” is declared with some level of certainty.

By examining the predictability of OCR character recognition patterns (1/I, 0/O, 2/Z, 8/B, and 5/S), one could make a more educated guess as to whether $X$ and $Y$ are a match. Oliveira-Neto et al., consequently, proposed a self-learning license plate matching algorithm that included a generalized ED technique with a weight function [3, 4]. The generalized ED technique assigns different weights to the edit operations as a function of the character; allowing a measure of similarity between the two plate strings. This technique is reliant on an association matrix to provide the measure of similarity of whether a character, e.g., “B,” recognized by an upstream LPR station is recognized as
“B” or “8” or any other characters at the downstream station. These measures of similarity are based on the performance of an associated pair of LPR stations.

LPR technology can achieve different levels of accuracy depending on the hardware and software of the camera, set-up (mobile or stationary), location (side of road or overpass), and on-site calibration. Outside factors also play a role in the level of accuracy; e.g., traffic and weather conditions. The two facets of LPR technology that have the largest effect on the overall performance are the capturing and reading of license plates. The capture rate is the rate of successful plate recognitions in the field of view. This is commonly affected by uncontrollable outside factors and parameters pertaining specifically to the camera’s hardware, installation, or on-site calibration. The read rate is the rate of correctly interpreting an entire license plate. This rate is based solely on the performance of the OCR engine. The accuracy of these facets is commonly uncontrollable by the data output user and dependent on the performance of the LPR technology.

From experience, the typical read rate of LPR cameras rarely surpass 80%, and more commonly perform at 60% or lower [1, 3-5]. Consequently, the portion of license plates strings that are correctly recognized and matched between a pair of LPR stations drop down to 35 % or less. The current license plate matching algorithm has been tested and proven above satisfactory at plate matching for sequential LPR stations with reasonable distance between stations (a few miles). However, there are challenging cases where LPR stations are non-sequential, spaced far apart (over 100 miles), experience minimal matches between them, and/or have poor OCR read rates. For these cases, it can be difficult to establish strong association matrices that would yield satisfactory matching results.

In order to establish an association matrix that could yield satisfactory matching results for the aforementioned challenging cases, more information is needed. We propose that by employing a third, and perhaps temporary, LPR station, additional information can be
gathered to *derive* a replacement association matrix for the initial pair of LPR stations. To evaluate this derived association matrix, we employ two simulations to 1) determine when the matrix should be used and 2) evaluate the overall performance of license plate matching.

### 3.2 Review of Association Matrix

The association matrix is the foundation of the license plate matching algorithm. If a poor association matrix is used, the result is low matching rates and/or increased false matches. The following section defines the association matrix and all current estimation methods.

#### 3.2.1 Definition

The association matrix $A$ between two LPR stations $(g, h)$ is a $N$ by $N$ square matrix whose elements are the conditional probabilities $p(b|a)$ of observing a character reading $b$, $b \in N$, in station $h$ for a given character reading $b$, $a \in N$, in station $g$ [3]. The set $N$ is the set of possible alpha-numeric characters $\{0,1,2, ..., 9, A, B, ..., Z, \lambda\}$, where $\lambda$ is the null character. The null character can be more than just an unknown symbol, but also represent deletions and insertions in the plate string text. When observing the association matrix, the probability of correctly matching a character is seen in the main diagonal elements and the misreading of a character in the off-diagonal elements. Each row of $A$ refers to a given character recognition at station $g$, with the column referring to the associated reading at station $h$.

An association matrix’s elements can be represented in two forms: unitary (counts of matches) and statistical (conditional probability) values. Figure 3-1 displays an example of a unitary association matrix- where each value represents the number of matches between each character. To estimate the $p(b|a)$ of matrix $A$, the following equation shows the relationship of the unitary and conditional probability values:
where \( \rho_{ab} \) is the frequency that character \( b \), \( b \in N \) is associated to character \( a \), \( a \in N \) in the unitary association matrix and \( \rho_a \) is the number of times character \( a \) has been matched at LPR station \( g \).

### Figure 3-1. Example of unitary association matrix.

There are three types of association matrices discussed in this paper; the definition of each is as follows:

\[
p(b|a) = \frac{\rho_{ab}}{\rho_a} \tag{3-1}
\]
- Ideal Association Matrix $A_1$ – is an association matrix, which associates perfectly the conditional probability of matches between two LPR stations. This is the ideal matrix that is hoped to be achieved during the learning process of the matching algorithm.

- Learned Association Matrix $A$ – is an association matrix, which is learned directly from comparing output strings from a pair of LPR stations used for matching. This is learned using the matching algorithm.

- Derived Association Matrix $A^*$ – is an association matrix created by multiplying two association matrices connected with a shared LPR station. This is created using two learned association matrices.

### 3.2.2 Example of an Ideal Association Matrix

Suppose there are three LPR stations (1, 2, 3) and the character set used for the license plates has only two elements, either “A” or “B”; thus, the association matrix is a 2 by 2 matrix. The two characters have the same probability of being captured (50% for each); also, representing the distribution of the characters. Assume that the character read rate of each LPR camera’s OCR engine is 80%, for all stations. Then the distribution matrix $D$ and the truth matrix $T$ for the stations will be as follows:

$$
D = \begin{bmatrix} 0.5 & 0.5 \end{bmatrix} \quad T = \begin{bmatrix} 0.8 & 0.2 \\ 0.2 & 0.8 \end{bmatrix}, \quad \text{for all stations}
$$

Knowing the distribution and truth matrix, the frequency of each match can be calculated. As seen in Figure 3-2, each character has the probability of being captured 50 percent of the time (as seen in the first column). By applying the truth matrix, the character will be matched correctly 80 percent of the time (as seen in the middle two columns). The result represents the frequency of each match.
Figure 3-2 Frequency tree for a two character example.

Using Equation 3-1, the probability of each match can be calculated using the above results. This is shown below:

\[
\begin{align*}
\frac{p("A" | "A")}{\rho("A")} &= \frac{\rho("A" \& "A")}{\rho("A")} \\
\frac{p("B" | "A")}{\rho("A")} &= \frac{\rho("A" \& "B")}{\rho("A")} \\
\frac{p("A" | "B")}{\rho("B")} &= \frac{\rho("B" \& "A")}{\rho("B")} \\
\frac{p("B" | "B")}{\rho("B")} &= \frac{\rho("B" \& "B")}{\rho("B")}
\end{align*}
\]

\[
\begin{bmatrix}
0.32 + 0.02 \\
0.32 + 0.08 + 0.02 + 0.08 \\
0.08 + 0.08 \\
0.08 + 0.02 + 0.08 + 0.32
\end{bmatrix} = 0.68 \\
\begin{bmatrix}
0.08 + 0.08 \\
0.32 + 0.08 + 0.02 + 0.08 \\
0.02 + 0.32 \\
0.08 + 0.02 + 0.08 + 0.32
\end{bmatrix} = 0.32
\]

Thus, the ideal association matrix for any station pair (1 – 2, 2 – 3, 1 – 3) will be,

\[
A_i = \begin{bmatrix}
0.68 & 0.32 \\
0.32 & 0.68
\end{bmatrix}, \quad \text{for every pair of two stations}
\]
3.2.3 Estimation from Truth Matrices

The truth matrix $T$, or confusion matrix, is a $N$ by $N$ square matrix whose elements are the probabilities $p(t|a)$ of each character's interpretation; where the matrix's rows represent the character reading $a$, $a \in N$ and the columns representing the true character reading $t$, $t \in N$. Oliveira-Neto et al.[3, 6] proposed estimating the association matrix $A$ using the following Bayesian expression:

$$p(b|a) = \sum_t p(b|t) \cdot p(t|a), \quad t \in N$$  \hspace{1cm} (3-2)

In terms of $A$ and $T$, Equation 3-2 can be written as:

$$A = T_g \cdot T_h^{tr}$$  \hspace{1cm} (3-3)

where $T_g$ and $T_h$ are the truth matrices for stations $g$ and $h$, respectively. To make the matrix multiplication possible in Equation 3-3, $T_h$ must be transposed; therefore, the rows of $T_g$ are refer to the character readings at station $g$ and the columns of $T_h^{tr}$ represent the character readings at station $h$. This process of estimating association matrices from truth matrices, though greatly improving the performance of license plate matching, still has many downsides:

- To acquire the needed ground truth of each character, it requires extensive man hours to extract and visually inspect each license plate; resulting in a very costly and time consuming process.
- In order to estimate a good association matrix, large sample sizes are required. Because of the cost and time to manually identify license plates, this would be impossible for large systems of LPR station (100+).
- The truth matrixes contain separated sets of readings; thus, observing unique plate patterns for each station. The ideal association matrix only contains
matches between the associated stations, not the full set of reading for each station. These additional readings could result in the estimated association matrix being skewed from the ideal association matrix.

3.2.4 Estimation from Matching Algorithm

Oliveira-Neto et al. proposed a more efficient way of estimating association matrices by means of a heuristic text-mining algorithm to match plates between two LPR stations [3, 4]. The matching process begins with an initial association matrix, i.e., an identity matrix, to generate a set of learned matches $M$ using the weighted edit distance technique, as described in [4]. By incorporating the individual character matches from the matched set, an estimated association matrix can be determined.

As stated in [4], the weight function is calculated using conditional probability elements of matrix $A$, as seen in the following equation:

$$\gamma(a \rightarrow b) = \log \left( \frac{1}{p(b|a)} \right), \quad (3-4)$$

where $\gamma$ is the weight function relating the corresponding pair of characters $(a, b)$ such that $\gamma(a \rightarrow b)$ is a non-negative real number assigned to each character’s edit operation. These edit operations are then summed to determine their respective plate string’s edit distance.

Oliveira-Neto et al. shows that the estimated association matrix approaches the ideal association matrix as the iterations of the self-learning process progresses [4]. The learning process continues by repeatedly applying the matching process to progressively find more matches and better association matrices. Each iteration $k$ of the learning process estimates a better matrix $A_k$ directly from a matching set $M_k$, that is obtained from the previous matrix $A_{k-1}$. The matching algorithm will stop learning when the difference between two successive estimations falls below a pre-assigned threshold.
The process of estimating the association matrix is important to attaining the highest level of matching performance and quickest learning speed. The matching algorithm takes a heuristic approach in determining the elements within a learned association matrix. Meaning, the estimated learned association matrix will never be the ideal association matrix and the purpose of the algorithm is to find a matrix that is sufficient for the immediate goals.

3.3 Deriving an Association Matrix

When the read rates of LPR cameras are poor and the sample of vehicles is minimal, the self-learning matching algorithm simply doesn’t have enough data to learn from to produce a high-performing association matrix. This can regularly be seen in LPR stations that are spaced a great distance apart, e.g., 100 miles, with minimal matching vehicles traversing between the two stations within a reasonable amount of time. In order to improve the association matrix, more license plate matches are needed to observe the performance of the associated LPR stations. The proposed solution is the addition of a third LPR station, either existing or temporarily deployed, to capture additional license plates. Theoretically, the third LPR station could be located in any synchronization with the other two LPR stations, just as long as there are in common captured matches between all stations. The additional license plate matches captured at the third LPR station will then be used to create two learned association matrices that are used to derive a more accurate association matrix for the initial pair of LPR stations.

The newly derived association matrix will contain the same elements as mentioned in Equation 3-1, the only difference is how it is estimated. Similar to the estimation process using truth matrices described in Section 3.2.3, the derived association matrix is estimated using matrix multiplication of the association matrix between two pairs of LPR stations sharing a common LPR station. For example, if two existing LPR stations 1 and 2 where supplemented with a third LPR station 3, then two additional association matrices $A_{13}$ and $A_{32}$ are estimated from the creation of two additional station pairs.
When $A_{13}$ and $A_{32}$ are multiplied the result is the derived association matrix $A_{12}^*$ between LPR stations 1 and 2. In terms of the matrices the equation is represented as follows:

$$A_{12}^* = A_{13} \cdot A_{32} \quad (3-5)$$

### 3.3.1 Proof

Suppose three 3 stations are capturing license plates and the truth matrices $(T_1, T_2, T_3)$ for all stations are known. With the distribution matrix $D$ of true characters, the distribution of read characters at each station can be calculated by multiplying the true matrices of each station as follows:

$$D \times T_1 = R_1, \; D \times T_2 = R_2, \; \text{and} \; D \times T_3 = R_3$$

where $R$ is a $37 \times 1$ matrix with elements containing the number of reads for each character in set $N$. In the set of matches from station $i$ to $j$, the association matrix $A_{ij}$ contains the conditional probability to get each character in station $j$ given the character in station $i$. Thus, the set of read characters in station $j$, $R_{ij}^*$ can be simply obtained by multiplying the set of read characters in station $i$, $R_{ij}^*$ and the association matrix $A_{ij}$.

$$R_{12}^* \times A_{12} = R_{2}^{12}, \; \text{where} \; R_{12}^* \subset R_1 \text{ and } R_{2}^{12} \subset R_2$$

$$R_{23}^* \times A_{23} = R_{3}^{23}, \; \text{where} \; R_{23}^* \subset R_2 \text{ and } R_{3}^{23} \subset R_3$$

$$R_{13}^* \times A_{13} = R_{3}^{13}, \; \text{where} \; R_{13}^* \subset R_1 \text{ and } R_{3}^{13} \subset R_3$$

To get the derived association matrix, we need to have the same set of matches for the different pairs of stations (the associated true characters are the same), which are the sub-set of all read characters in each station.
If \( R_{12}^{12} = R_{13}^{13}, R_{22}^{12} = R_{23}^{23}, R_{32}^{23} = R_{31}^{13} \) by assuming they are the same set of matches, then:

\[
R_{23}^{23} \times A_{23} = R_{12}^{12} \times A_{12} \times A_{23} = R_{13}^{13} \times A_{13} = R_{31}^{13}
\]

If we assume that we have a perfect matching algorithm, the set of read license plates for all stations will be associated perfectly to the distribution of true license plates passing the stations with respect to the true matrices. Then, the calculation can be even more simplified:

\[
R_{1} \times A_{12} \times A_{23} = R_{2} \times A_{23} = R_{1} \times A_{13} = R_{3}
\]

Therefore,

\[
A_{12} \times A_{23} = A_{13}^*
\]

Likewise,

\[
A_{13} \times A_{32} = A_{12}^*
\]

\[
A_{21} \times A_{13} = A_{23}^*
\]

### 3.3.2 Theoretical Examination of Deriving Association Matrix

From the example of the ideal association matrix (Section 3.2.2), we can obtain a probability that a single match is added to each cell of the association matrix, which we define here as a frequency matrix. For instance, there are two cases ‘A-A-A’ and ‘B-A-A’ of the cell ‘A’ to ‘A’ in the example. Therefore, the frequency of the cell ‘A’ to ‘A’ is 0.32 + 0.02 = 0.34. By adding 1 to the cell each time to a \( N \times N \) zero matrix, we can obtain a new association matrix with a frequency of the event and calculate the closeness between the ideal association matrix and the new association matrix. Closeness is the measure of how close two matrices are to each other. Then, the expected closeness for
the sample size \( n \) (the number of matched characters), will be the sum of the closeness for all possible sets and their frequencies.

If we consider adding a single character match to each cell of an association matrix as an event, then there are \( N \times N \) number of events at each time a single character match is added. The number of all possible events for building the learned association matrix is \((N \times N)_1 \times (N \times N)_2 \times \ldots \times (N \times N)_n = n^{(N^2)}\). Since the derived association matrix is created by multiplying two association matrices, the number of all possible events for building the derived AM is \( n^{(N^4)}\). Due to the exponential growth of the computational work load, we examined the expected closeness to the ideal association matrix for both the learned and derived association matrices, up to the sample size of 5. (For \( n = 5 \) and \( N = 2 \), the number of all possible events for building the derived AM is 152,587,890,625.)

With this sample exercise, the expected closeness for each sample size \((n)\) was calculated. Figure 3-3 shows how close the learned and derived association matrices are to the ideal association matrix, for the sample size of 1 to 5 \((1 \leq n \leq 5)\) and the LPR camera accuracies of 70% to 90%. The calculation was assumed using a 2 by 2 matrix \((N = 2)\), however, the results could change with a different size of matrix.

The result implies that the derived association matrix is closer to the ideal association matrix for low LPR camera accuracies and small sample sizes (number of matched characters). Note that the learned association matrix is closer to the ideal association matrix when accuracy is 90%; however, this does not guarantee higher performance of LPR matching algorithm with the learned association matrix since there are many factors affecting the performance of LPR matching. The evaluation of performance will be discussed in Section 3.4.3.
When there are many sources of variance affecting the performance of a LPR camera, it is necessary to have a large sample size to achieve a desired error precision. More specifically, it is important to know the needed amount of data to obtain an accurate estimate of the learned association matrix. Oliveira’s initial self-learning algorithm required approximately 60,000 characters to estimate a learned association matrix [6]. This approximation may no longer hold true, since much advancement has been made to the self-learning algorithm resulting in a drastic reduction in sample size. But by simulating a license plate OCR dataset, the ability to examine a limitless number of characters is gained. Monte Carlo simulation, therefore, becomes a strong tool for examining the relationship between LPR camera accuracy and required sample size.

3.4 Simulated Experiments

![Figure 3-3. Theoretical Closeness of derived association matrix.](image)
From this relationship we can 1) determine when the derived matrix should be used and 2) evaluate the overall performance of license plate matching with the derived matrix.

### 3.4.1 When should we use the derived association matrix?

The derived association matrix is theorized to be the solution to the shortcoming of the learned association matrix caused by poor accuracy and small sample sizes. Based on the proof and theoretical estimation, a procedure was designed to determine when the derived association matrix should be chosen over the learned. This simulation is based on an idealistic situation where all characters are matched and the need for a travel time window is removed. The following discusses the steps of the simulation, along with an example and the experimental conditions. The set of characters, matches and association matrices are simulated using the following:

1. Generate a distribution matrix \( D \) to represent the distribution of all true characters. The matrix’s elements contain the cumulative conditional probability for each character within \( N \). Matrix \( D \) will be the same for all stations (there is no missing/inserted characters between the stations).

2. Generate a truth matrix \( T \) for each LPR station, which represents the character matching accuracy of the LPR cameras. The elements of these matrices contain the conditional probability across all columns.

3. Generate a random number \( x \), where \( 0 \leq x \leq 1 \), to determine the \( i \)-th true character value for all stations with respect to the cumulative conditional probability in matrix \( D \). This is only done once for each character.

4. For the \( i \)-th true character which was generated in step 3, assign the character read by the OCR at each station by matching the random numbers to the respective cumulative conditional probability of matrix \( T \) for each LPR station. This is repeated for each true character captured at all LPR stations.

5. Based on the predetermined sample size, the simulation creates a sample of read characters from step 4. Then, every time when a character is created, a
learned association matrix $A_{12}$ is updated by adding the pair of characters read at station 1 and 2, and so forth for other pairs of stations.

6. Once the learned association matrices are calculated each time in the step 5, a derived association matrix $A_{13}^*$ is calculated by multiplying $A_{12}$ and $A_{23}$.

7. Then, the closeness to the ideal association matrix from both the learned association matrix $A_{13}^*$ and $A_{13}^*$ are calculated by equation 3-6.

8. To capture the impact of accuracy of the LPR cameras, the process is repeated from step 2 and step 7 with different accuracy of the LPR cameras.

For example, there are two stations that only capture characters “A” and “B,” meaning the association matrix is a 2 by 2 matrix. Characters “A” and “B” have a license plates distribution of 40 % and 60%, respectively. This distribution is show in matrix $D$ below:

$$
D = \begin{bmatrix}
A & B \\
\cdot4 & \cdot6
\end{bmatrix}, \quad D^c = \begin{bmatrix}
\cdot4 & 1
\end{bmatrix}
$$

where $D^c$ is the cumulative distribution. LPR stations 1 and 2 both have read rates very close to 68%. The cumulative conditional probability of truth matrices $T_1^c$ and $T_2^c$ are calculated based on the read rates and are as follows:

$$
T_1^c = \begin{bmatrix}
.7 & 1 \\
.35 & 1
\end{bmatrix} \quad \text{and} \quad T_2^c = \begin{bmatrix}
.66 & 1 \\
.3 & 1
\end{bmatrix}.
$$

For a sample size of 4 characters, 4 random numbers (.88, .73, .51, .21) are generated and compared to matrix $D^c$ to determine the true character values (B, B, B, A). To determine the captured character at LPR station 1, another set of random numbers (.37, .61, .18, .33) is generated and compared to $T_1^c$ with the results (B, B, A, A). The last step is repeated for the second LPR station using $T_2^c$ with the result of (B, B, B, A). For this sample, LPR station 1 misread the third character while station 2 read all characters correctly. From these results a learned association matrix $A_{12}$ is calculated. The following is matrix $A_{12}$ shown in count and probability form:
The following experimental conditions were used to evaluate the closeness of the association matrices:

- The use of travel time is exempted from the matching process. Though travel time is an important procedure in the matching algorithm, it remains important to measure the closeness of the self-learning matching without the parameter of a travel time window.
- When determining read rate of LPR stations, the simulation assumes that all LPR cameras perform at the same level of accuracy and generates a truth matrix based on that accuracy. Identical accuracy is not common in the real world, but setting a constant accuracy for each simulation aides in a better understanding of the relationship between LPR read rates and sample sizes.
- The simulation captures one character at a time, versus capturing a whole license plate string. Meaning the simulation does not consider how many characters are in each plate, since an association matrix is updated based on character-basis. In doing this the reliance on the position or sequence of a character during recognition is removed.
- When the characters are captured at each station, the simulation assumes that they are all matched perfectly, although they may or may not be read correctly. This allows us to see the pure impact of accuracy of the LPR cameras on the closeness to the ideal association matrix from the learned and derived association matrix, even though the accuracy affects the performance of the matching algorithm.
- The accuracy of the LPR camera is a probability of reading a character as its original form (e.g., ‘a-a’, ‘b-b’)

\[
A_{12} = \begin{bmatrix} 1 & 1 \\ 0 & 2 \end{bmatrix} = \begin{bmatrix} .5 & .5 \\ 0 & 1 \end{bmatrix}
\]
• The number of sample size is a number of characters matched by the algorithm (not necessarily saying the number of possible matches unless the capture rate of the algorithm is 100%)

### 3.4.2 Closeness of Association Matrices

To measure the closeness of the derived and learned association matrices, they are compared to the ideal association matrix used from the simulation. The following equation calculates the closeness (difference) in the two association matrices:

\[
\text{closeness} = \frac{\sum_{i=1}^{N} \sum_{j=1}^{N} |a_{ij} - b_{ij}|}{\sum_{i=1}^{N} \sum_{j=1}^{N} |b_{ij}|},
\]

where

\[b_{ij}\] is the elements with row \(i\) and column \(j\) of the ideal AM,

\[a_{ij}\] is the elements with row \(i\) and column \(j\) of the compared AM.

### 3.4.3 What is the performance of the derived association matrix?

Although the first experiment examines the closeness between the derived and learned to the ideal association matrix, the closeness is not enough to see the performance of matching, such as matching rate. The first experiment assumed that the algorithm could match every possible pair perfectly to purely see the impact of accuracy of OCR to determining whether the derived or learned association matrix should be used. However, the performance of matching algorithm decreases as the accuracy of OCR decreases.

As a result of a low performing matching algorithm, the association matrices of multiple pairs of stations will likely have different sets of matches. Since the derived association
matrix works well with the same set of matches (as discussed in the proof), the low accuracy of OCR will result in the derived association matrix worsening.

To evaluate the performance of the derived and learned association matrix for varying LPR accuracy, the matching algorithm and travel times were added to the simulation to perform a real-world assessment. The sets of LPR stations, matches and association matrices are simulated using the following:

1. Generate a distribution matrix $D$ to represent the distribution of all true characters. The matrix’s elements contain the cumulative conditional probability for each character within $N$. Matrix $D$ will be the same for all stations (there is no missing/inserted characters between the stations).

2. Generate a truth matrix $T$ for each LPR station, which represents the character matching accuracy of the LPR cameras. The elements of these matrices contain the conditional probability across all columns.

3. Generate $n$ vehicles with the travel times at each station with the normal distribution with a mean of 60 and a standard deviation of 10. The time window was set according to the normal distribution with the z value of 1.96 to capture approximately 95% of true matches from the population. Based on the matrix $D$ in step 1, generate 6 true characters for each vehicle.

4. Based on the true matrix $T$ in step 2, generate read license plates for every vehicles at all stations. Some of the vehicles may not be captured depending on the capture rate of license plates.

5. Run the self-learning LPR matching algorithm to get the learned association matrices of each pairs of stations.

6. The derived association is calculated by multiplying the two sets of connected learned association matrices, which were obtained in step 5.

7. Run the matching algorithm without learning to evaluate the performance of the matching using the learned association matrices.
8. Run the matching algorithm without learning to evaluate the performance of the matching using the derived association matrices.

The following experimental conditions were used to evaluate the performance of the association matrices.

- In contrast to the first experiment, the travel time of vehicles were also considered in this simulation to reflect more realistic license plate matching. The narrow travel time window in the LPR matching algorithm makes the algorithm faster, but the matching rate lower unless the travel time window is large enough to capture all true matches.

- Likewise the first experiment, the simulation assumes that all 3 LPR cameras perform at the same level of accuracy and generates a truth matrix based on that accuracy. However, as this simulation considers license plates with 6 characters, the accuracy includes not only the accuracy of OCR, but also the capture rates of license plates and characters in each plate.

3.4.4 Performance of LPR matching

Using the set of matches from step 8 and 9 of the performance simulation, the positive matching rate (\(pmr\)), false matching rate (\(fmr\)) and matching rate (\(mrate\)) are calculated based on the matching results, as seen in the following equations:

\[
pmr = (|M| - |F|)/|M'|, \quad (3-7)
\]

\[
fmr = |F|/|M|, \quad (3-8)
\]

\[
mrate = \frac{2}{\left(\frac{1}{1-fmr}\right) + \left(\frac{1}{pmr}\right)} \quad (3-9)
\]
where $F = \{(X_m, X_n) | (X_m, X_n) \in M, X_m' \neq X_n'\}$ is the set of false matches from $M$, $M' = \{(X_m, X_n); X_m' = X_n'\}$ is the total set of true matches, with $M$ as defined before. The mrate is the harmonic mean, a common measure of rates, of the fraction of matches that are the actual matches ($1 - fmr$) and the fraction of positive matches that are identified ($pmr$).

### 3.5 Results

#### 3.5.1 When should we use the derived association matrix?

Figure 3-4 (a-d) shows the closeness to the ideal association matrix and the number of characters, i.e., sample size, needed for the learned and derived association matrices. Each image represents a different LPR character read rate ranging from 40 to 70%. The solid line represents the closeness of the derived association matrix to the ideal association matrix, whereas the dotted line represents that of the learned association matrix.

As seen in Figure 3-4, the closeness to the ideal association matrix gets smaller as the sample size increases for both the learned and derived association matrices. Also, the derived association matrix is closer to the ideal association matrix up to a certain number of matched characters for the each accuracy of LPR cameras.

The intersecting point, where the derived association matrix no longer is closer to the ideal association matrix, gets smaller (moving to the left in the Figure 3-4) as the accuracy increases. This is summarized in Figure 3-5, to examine when the derived association matrix should be used based on the accuracy of the LPR cameras. Each point of Figure 3-5 represents an intersecting point where the learned association matrix becomes closer to the ideal association matrix.

Note that this simulated experiment assumed that the matching algorithm is perfect; therefore, the characters have 100% positive matching even though the read characters
may be incorrect. This is depending on their accuracy of LPR cameras and the closeness to the ideal association matrix may not reflect the performance of matching algorithm thoroughly.

Figure 3-4 Differences between association matrices with shared read rate of 40% (a), 50% (b), 60% (c) and 70% (d).
3.5.2 **What is the performance of the derived association matrix?**

To measure the performance of matching algorithm depending on whether the learned or derived association matrix is used, an additional simulated experiment was performed. In this simulation, as described in Section 3.4.3, the Oliveira-Neto et al's self-learning LPR algorithm was used with the consideration of travel time[4].

Figure 3-6 shows the matching rate performance curves of by the number of matched plates for the learned and derived association matrices. The results are averaged from three different simulations with the same accuracy of LPR cameras. In the results, the matching algorithm using the derived association matrix performed better most of time.
compared to the learned association matrix. Especially, the gap of performance is large when the number of matched license plates is small.

As shown if Figure 3-6, to achieve a 90% matching rate, the matching algorithm using the derived association matrix only requires about 300 matched license plates while the learned association matrix requires more than 1,200 matches. To achieve the 90% positive matching rate, the derived association matrix requires only about 500 matches whereas the learned association matrix had to learn more than 1,800 matches.

3.5.3 Summary

To evaluate the derived association matrix, we employ two experiments: to 1) determine when the matrix should be used and 2) evaluate the overall performance of license plate matching. The first experiment examined the closeness of the derived and learned association matrix to the ideal association matrix. The second simulated a case study of license plate matching that evaluated the performance of the derived and learned association matrix. In order to examine multiple parameters, the two experiments where compared with different conditions; for example, experiment 1-used characters as the sample, no travel time constraints, 100% matching while experiment 2- used license plates as the sample, had a travel time constraint, and the matching was based on the LPR accuracy; consequently, resulting in a perfect world versus real world analysis.

The first experiment shows that the derived is close to the ideal when the sample size is small and the accuracy is low, but the second one shows that the performance of matching is still higher by using the derived even when the accuracy is high. The results show that even with the learned association matrix having the smaller closeness value, does not guarantee that it will also have the higher performance rate. It could be that the derived association matrix’s performance does not depend on the closeness to the ideal, but the calculation of edit distance or the travel time window in the matching algorithm. Or that having a high accuracy may aid in equal sets of matches for 3
stations (as seen in the proof in Section 3.3.1) so that the derived association matrix outperforms the learned association matrix.

Figure 3-6 mrare performance curve for OCR character accuracy.
3.6 Conclusion

By utilizing a derived association matrix, new life can be brought back to LPR stations that may have been forgotten, due to poor location or performance. By providing a sense of where and when the derived association should be used, LPR users can achieve station set-ups that failed in the past, due to low number of matches. Not only does this make LPR technology more flexible, but the derived association matrix also showed an increase in learning speed. As shown in the paper, the derived association matrix was able to reach a 90% matching rate with 300 matched license plates while the learned association matrix required 1,200 matches.

With tens of thousands of LPR stations deployed in the United States; it is possible that derived may be a better alternative for matching license plates over a large network of LPR stations. The derived association matrix may also become a solution for other unsuccessful OCR text matching applications other than license plate text.

Future studies will be required to examine the other factors related to the performance of the license plate matching, such as the edit distance criteria and travel time window. The simulation of more than 3 LPR stations would also be valuable in determining if derived can be determined from more than two pairs of stations. Most importantly, the formulation of the edit distance needs to be critiqued and possibly even replaced with a new calculation.
References


CHAPTER 4. THE USE OF LICENSE PLATE RECOGNITION TECHNOLOGY FOR GROUND TRUTH DATA COLLECTION
This chapter presents a modified version of a research paper by Stephanie R Hargrove, Hyeonsup Lim, Lee D. Han, and Brad.

Abstract

Advancement in information technology in recent years enables some non-traditional methods for estimating travel time information through the use of large datasets of various accuracy and availability. By repurposing or multi-purposing technologies such as Bluetooth, GPS, and ubiquitous cellular devices, innovative efforts have seen mixed success in aggregating data from a multitude of devices to derive travel time condition at different geometric resolutions. The challenge, though, is how the performance of these emerging technologies could be measured against a bona fide ground truth (of travel time) when the ground truth can be exceedingly costly and difficult to obtain.

To this end, this study took advantage of a high-performing license plate recognition and matching system (97% matching rate with less than 1% false-positives) to establish high-accuracy travel time ground truth at two sites for 12 hours or so each. The sites were chosen where a handful of other technologies were also deployed so that concurrent data could be collected and compared. The objective of this paper is to provide several key considerations for real-time traffic data evaluation for general cases, rather than a definite and specific conclusion. The key items include: obtaining reliable ground truth data, transforming and comparing incompatible datasets, and data quality evaluation measurements.
4.1 Introduction

As real-time travel information programs and state-of-the-art transportation applications become a norm for both the transportation community and public users, traditional data collection systems may no longer capture the newly, expected level of travel data. However, advancements in information technology in recent years have enabled non-traditional methods for estimating travel information through the use of large non-traditional datasets of various accuracy and availability. As traffic data continues to increase in size, the need for a better understanding of the data’s quality becomes more important. For real-time travel information data, a variety of data collection methods and technologies are employed to produce aggregated travel data from multiple sources of data. Evaluation of this aggregated and data collection method is essential not only prior to the use of the data, but also continuously as the data are used for various purposes to ensure ongoing validity. The measurement of data quality is multifaceted and requires guidance to ensure fair and consistent evaluation procedures.

To aid in the delivery of travel data to consumers, the Federal Highway Administration (FHWA) ruled that all States must have an established real-time information program for traffic and travel conditions covering all Interstate system highways. This federal mandate comes from the Safe, Accountable, Flexible, Efficient Transportation Equity Act: A Legacy for Users (SAFETEA-LU) under the heading of Congestion Relief that requires the Secretary of Transportation to implement a Real-Time Management Information Program [1]. The goal is for each State to make available travel data in real-time to local governments and travelling public of all major freeways.

The high cost to install and maintain a large number of roadside traffic sensors covering the entire length of all major freeways is a limiting factor in the ability of state agencies to collect real-time travel data on their own. Therefore, the need for real-time travel speed and time has resulted in the advent of numerous private companies collecting and distributing traffic data independently of government agencies. These private
companies have marketed and sold their real-time traffic data to Departments of Transportation, who in turn report to the general public. The traffic data provided is aggregated from various sources, including Global Positioning Systems (GPS), probe vehicles, cellular devices, historical data, and even social media. The aggregation of all these data sources to generate a single value for a single time increment is not a simple task and requires verification and validation. However, the private sector chooses to conceal many algorithms used in the aggregation process and this has become a concern when determining the quality of real-time travel information.

The new methods of travel data collection, i.e., floating traveler data, provides more detailed information about road use across an entire roadway network. However, there are numerous concerns about accuracy, e.g., locational accuracy, road user coverage, and aggregation methods. To address these concerns, evaluations must be completed using a highly accurate data collection method to provide ground truth. Ground truth is absolute data actually measured in the field. To select the proper ground truth collection method, many things should be considered, such as measurement error and sample size. For the purpose of this paper, license plate recognition (LPR) technology is considered to be the optimal collection method for all ground truth. LPR data is obtained using a pair of mobile LPR units to automatically acquire and record license plates at sequential locations along a study route.

The objective of this paper is to provide several key considerations for real-time traffic data evaluation for general cases, rather than a definite and specific conclusion. The key items include: obtaining reliable ground truth data, transforming and comparing incompatible datasets, and data quality evaluation measurements.

### 4.2 Ground Truth and Data Sources

With the increasing advancements of travel time technology and the vast amounts of data becoming accessible, the many needs for evaluation also increase. The challenge
is, however, there is not one absolute solution until perfect ground truth is obtained by capturing the whole roadway population. Though the ideal ground truth may be near impossible to obtain, an acceptable error can be defined. Thus, within the limitation of data accessibility, the evaluation is performed using the best available resources and analysis metrics.

### 4.2.1 Ground Truth

To select the proper source that is close to ground truth, many things should be considered, such as measurement error and sample size. Also, actual travel time and travel speed information must be independently collected for the data collection location to validate all data sources for more accurate comparisons. These ground truth data collection methodologies [2-5] have included probe vehicle [6-8], Bluetooth [9-15], LPR technology [16-21] and radio frequency identification (RFID) [22, 23]. For the purpose of this study, LPR technology is employed to collect all ground truth. License plate recognition is chosen over other roadside technologies, because the technology’s small detection window results in a relatively small travel time error, especially at higher speeds [24]. LPR technology also permits a microscopic examination of the traffic flow by tracking vehicles within the road lanes, allowing a better understanding of the what travel speed is being reported— the ‘fast’ or the ‘slow’ lane.

The ground truth is collected using a pair of mobile LPR units to automatically acquire and record license plates at sequential locations along the study route. The acquired license plates are then matched automatically by means of a self-learning text-mining algorithm developed by the University of Tennessee [25, 26]. The algorithm utilizes a travel time window and weighted edit distances for each license plate character to drastically increase the number of correctly matched license plates (98% matching rate with less than 1% false-positives). The travel data of a vehicle is then calculated by finding the difference between the timestamps of a matched license plate. Du et al.
have completed a comprehensive review of the existing LPR technologies and applications [17].

4.2.2 Traffic Data Sources for Comparison

Even though multiple studies have completed similar examinations of private providers [9, 16, 27-29], it is crucial to re-examine them across different locations and time periods. The individual traffic data collected by each private provider is highly unique to both the provider and the location. For example, some states make available real-time traffic data from the state-installed roadside sensors, thus increasing the accuracy of the private provider’s reported traffic data. The State of Tennessee, however, only makes their roadside sensor data available in five minute time increments, while other states have much smaller time increments available. Thus, a five minute time period becomes more difficult to utilize than a 30 second time period when reporting in one minute increments.

To ensure a thorough examination of traffic data sources, two private providers and two roadside technologies were selected. The private providers are INRIX and HERE (formerly known as NAVTEQ) and are henceforward referred to as Data Provider 1 (DP1) and Data Provider 2 (DP2). These data providers are both private vendors that supply real-time travel speeds and times on a subscription basis. Both providers report aggregated traffic data using the Traffic Message Channel’s (TMC) coded locations at one-minute increments. Many concerns arise about private providers because they lack the ability to share the data aggregation process and the raw data points used to determine the reported traffic data. By comparing the reported aggregate data to collected ground truth, the accuracy of each private provider will be examined in this paper.

If the State of Tennessee decides to independently collect real-time traffic data, or perform additional accuracy analysis of private providers, it will be important to establish the reliability of readily available technology to capture the needed data. Thus,
Bluetooth and RTMS sensors are also evaluated. Bluetooth sensors have become a common tool for collecting traffic data because they are cost-effective, easy to set-up, and require minimal knowledge of the technology to operate [9, 11-14]. A Bluetooth sensor is capable of monitoring and measuring vehicular and pedestrian movement by identifying and comparing unique Media Access Control (MAC) addresses associated with Bluetooth-enabled electronic devices. These systems can be used to timestamp individual vehicles by sampling Bluetooth enabled devices and vehicles from the traffic stream of a predetermined route. By matching MAC addresses captured at two different locations using post-processing software, road speeds and travel times are derived. Each post-processing software application can yield unique results; Traffax’s BluSTATS software was employed for this study to guarantee similar results as other transportation agencies collecting Bluetooth data.

Many States already have RTMS sensors deployed on interstate highways in four major cities across the state. If RTMS sensors prove to be a reliable data source, then States can confidently use the readily available data for real-time traffic data and quality control. RTMS sensors detect volume, occupancy, speed and vehicle classification for a cross-section of a roadway every 30 seconds. The sensors are installed road side and are aimed to capture all lanes of traffic. Over time, an RTMS sensor may become less accurate at capturing traffic data due to moving of sensors, damage from natural elements, and/or inconsistent maintenance. To measure the continued accuracy of RTMS sensors, the technology was added to the evaluation. Table 4-1 contains a summary of all evaluated traffic data technologies and providers.

4.2.3 Study Route and Data Collection

The location selection was based on two requirements. The first requirement being the availability of all data sources safely within a TMC roadway segment. Second, the traffic must experience high variance in speeds and travel times. This ensures a location that is the most difficult for monitoring.
### Table 4-1 Summary of selected traffic data technologies and providers.

<table>
<thead>
<tr>
<th></th>
<th>Bluetooth</th>
<th>DP1</th>
<th>DP2</th>
<th>RTMS</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Data Type</strong></td>
<td>Capture Time &amp; Device Sensor Strength</td>
<td>Speed &amp; Travel Time</td>
<td>Speed &amp; Travel Time</td>
<td>Volume, Occupancy, Speed &amp; Vehicle Classification</td>
</tr>
<tr>
<td><strong>Aggregation Level</strong></td>
<td>Individual Data</td>
<td>Aggregated Data</td>
<td>Aggregated Data</td>
<td>Aggregated Data</td>
</tr>
<tr>
<td><strong>Data Source</strong></td>
<td>Cellular and in-vehicle Bluetooth devices</td>
<td>State installed sensors, probe vehicles, GPS, cellphone</td>
<td>State installed sensors, probe vehicles, GPS.</td>
<td>Roadside vehicle detectors</td>
</tr>
<tr>
<td><strong>Time Resolution</strong></td>
<td>As captured</td>
<td>1 minute</td>
<td>1 minute</td>
<td>30 seconds</td>
</tr>
<tr>
<td><strong>Accuracy Checks Performed</strong></td>
<td>Post collection processing with filters.</td>
<td>Independently verified in large-scale testing.</td>
<td>Data checks prior to map matching.</td>
<td>Post collection processing with filters.</td>
</tr>
</tbody>
</table>

The site location encompassed a 1,800 ft. stretch of roadway on Interstate 40 and 65 in downtown Nashville, Tennessee, as shown in Figure 4-1. The LPR stations refer to three roadway overpasses above the interstate. The interstate is three lanes until the addition of a fourth lane from the second on-ramp. Just beyond station C is the I-40 and I-65 split, with two lanes going to each. This is a common place for incidents and unpredicted congestion; therefore, a great location to experience high variance.

During the data collection, the segment A to C contained six LPR cameras, three Bluetooth sensors, two RTMS sensors, and two TMC sections containing traffic data for both data providers. Two LPR cameras were set-up on each overpass at stations A, B
and C. The cameras shot down and captured the rear license plates of eastbound travelling vehicles on I-40. The Bluetooth sensors were mounted on light poles at stations A and C of the roadway segment (a third was at station B, but was lost due to a traffic incident). The two RTMS sensors are mounted along I-40, with the first at station A and the second located directly between stations B and C. The first observed TMC location for data provider data starts just prior to station A and stops near station B. The second TMC location continues sequentially after the first TMC and stops after station C.

Figure 4-1 Study route.
4.3 Transforming Travel Datasets

Before the evaluation process was conducted, several data composition and quality control techniques were implemented to verify the validity of the ground truth data collected and the preparedness of all data sources. This step is completed by determining a procedure for comparing the ground truth data to the collected data source, establishing appropriate time periods, and eliminating outliers.

4.3.1 Determining Comparison Procedure for Data Sources

Two methods are assessed to evaluate the travel speeds of collected data sources against ground truth: The first method assigns all data to a specific time period, e.g., one or five minutes, and calculates average speed for each period. The second method keeps all data in its original resolution (without changing the time period or generating average value). Figure 4-2 illustrates the two considered methods. The y-axis represents the speed in miles per hour and the x-axis represents time. All ground truth is represented in black and the data source being compared to is in red. Error (e) of the data source is represented by each red arrowed line for each segment of measurement.

Averaging data in method one establishes a simpler way to calculate deviations between more than two data sources since the paired comparison analysis will be based on single values for each time period of each data source. However, this method may produce incorrect results, which could be significantly different from the original data set. For example, if a chosen time period is larger than the original data set’s period, the accuracy of the calculated average for the time period cannot be guaranteed unless the number of traffic counts for each time period is known in the original data set. Furthermore, if the determined time period starts or ends at different times, the newly generated data may yield significantly inaccurate information, which is not identifiable unless all raw data is accessible.
Figure 4-2 Two methods for comparing speed data.
Because LPR technology can observe individual vehicles, this study compared travel speeds of each vehicle captured with LPR and to all other data sources, as seen in method two of Figure 4-2. For data sources with accessible raw data, such as Bluetooth and RTMS, the five minute moving average was calculated based on the time when each data point was identified.

### 4.3.2 Moving Average of Travel Speeds

Moving average of traffic data captured by the roadside technology was calculated to compare with travel speeds of the ground truth’s individual vehicles. The term *moving* is used because every time a new observation becomes available for the time series, it replaces the oldest observation in the equation, and a new average is computed. As a result, the average changes or moves as new observations become available.

Determine an appropriate time interval is very important; thus, it is crucial to have a realistic moving average. To achieve this, several trials examining different time intervals were implemented. It was determined that a five minute time interval provided the most effective sample sizes while maintaining the characteristics of traffic flow at each time period.

The time period $i$ was selected based on the time stamp of vehicle $k$ identified, including two and a half minute intervals, before and after the identified time stamp. After selecting all vehicles within the five-minute range at time period $i$, the Space Mean Speed (SMS) is calculated. The moving average of time period $i$ is defined as:

$$
\bar{v}_i = \frac{\sum_{k=1}^{n_i} \frac{1}{\bar{v}_{ik}}}{n_i}
$$

(4-1)

Where
\( \bar{v}_i \) is the space mean speed at time period \( i \)

\( n_i \) is the number of vehicles at time period \( i \)

\( v_{ik} \) is the travel speed of \( k \)-th vehicle at time period \( i \)

4.3.3 Conversion of Data from Travel Speed to Travel Time

Travel time is a standard measure of freeway service quality [30]. Travel time information is an indispensable part of travel information systems[31]. When capturing travel data using technology in the field it is important for the sensor spacing to spatially match the spacing of INRIX and HERE’s TMC section. However, in the field this can be impossible, because of roadway infrastructure and safety. Thus, travel speed data is converted to travel time for corresponding TMC sections. This serves as an additional metric to show its potential impact on travel time estimation.

The average section travel time \( tt_i \), which can be considered as a "true" mean travel time of the temporal and spatial section, can be estimated from the unbiased estimate of the space-mean speed as [32]:

\[
    tt_i = \frac{\Delta x}{\bar{v}_i}
\]

(4-2)

Where

- \( tt_i \) is the average section travel time at period \( i \)
- \( \Delta x \) is the distance traveled
- \( \bar{v}_i \) is the space mean speed at period \( i \)

For slow speeds, the smallest of speed change (1 mph), can have a large effect on the calculated travel time. For example, a one mile road section with a speed reduction
from 60 to 59 mph will experience a travel time increase of roughly one second. While the same one mile reduction from 10 to 9 mph will increase the travel time by 40 seconds. Therefore, if there is a constant one mph error across a speed dataset, the calculation of travel times using the dataset could be greatly affected at low speeds. For this study, as travel time is obtained by the inverse of travel speed, variations increase at low speeds for the PM peak and low speeds. Figure 4-3 illustrates the difference between using travel time and travel speed as an evaluation target. During the PM peak, some vehicles at low speed in the LPR data have longer travel times, over twice as long as the travel time recorded by Bluetooth. Because of this variation, travel times calculated for the roadside technologies may require additional scrutiny.

4.3.4 Outlier Elimination

Eliminating outliers, like changing a time period, may yield a different data set. Thus, no outlier was eliminated from the provided data from RTMS, INRIX and HERE in order to have an accurate evaluation of their reported travel speed and time.

For the Bluetooth data, the BlueSTATS software was used to determine all outliers. The basic concept is that all data points that are 3 or more standard deviations from the mean are categorized as an outlier.

In the matching procedure of the LPR observations, there is a method to select certain candidates based on a time window, which is determined by mean and standard deviation of travel time at each time period; thus, eliminating outliers. There is a small chance of incorrect matches, but it is considered not significant because LPR technology has a higher accuracy, as discussed, and renders visual investigation of individual license plates unnecessary.
Figure 4-3 Comparison of travel time and travel speed for segment A to C.
4.5 Data Quality Evaluation Measurements

With an increased demand for measuring travel data quality, multiple studies have been performed to examine the evaluation process [2, 4, 6, 29, 33-36]. One of the well-known comparisons of travel time technology was completed by the I-95 Coalition by performing five minute space mean speed validation across four states [29]. While taking into account the past evaluation methods, the following section will discuss the significance of our chosen measurements; including, visual investigation (using travel speed over time, confidence intervals and histograms) and the calculation of root mean square error for travel speeds and times.

4.5.1 Visual Investigation of Travel Data

Although several numerical metrics are provided for evaluation measurements, it is also important to investigate and interpret the figures visually to determine variations, patterns, and trends within the data. The following describes the reasoning for each visual in:

- **Travel Speed over Time for Ground Truth vs. Each Data Sources** – To have a clear understanding of the flow of traffic during the entire observation period, it is important to plot the travel speed.

- **90% Confidence Interval** - Confidence intervals (CI) are used to test the significance of difference, or sometimes identify outliers, by investigating whether the travel speeds of each data source are within a boundary for time and space. If the distribution of the original data set is known, such as t distribution or normal distribution, the CI can be obtained with mean and standard deviation. However, since the distribution of travel speed is unknown and may vary at each time period, the confidence interval was calculated based on percentiles (5th to 95th).

- **Error Distribution** – In order to visualize error patterns it is helpful look at them within a histogram. Turner et al suggests that the error distribution should be the
frequency distribution of individual errors[4]. Though by just examining the errors without the relationship of travel speed, the ability to examine the percentage of error is lost. The histogram utilized by this study calculates probability densities by dividing the travel time of the examined data source by the travel time of the ground truth. A probability density of one means no error, while greater than one is experiencing a data source larger than the ground truth and vice versa for values below one. This distribution is helpful in visualizing bias, range and magnitude; something that can’t be seen in averaged values.

4.5.2 Root Mean Square Error

The Mean Square Error (MSE) is “useful when we are concerned about large errors whose negative consequences are proportionately much bigger [sic] than equivalent smaller ones (e.g., a large error of 100 vs. two smaller ones of 50 each).”[37] Thus, using MSE rather than Mean Absolute Error (MAE) means that larger errors of travel speed are accounted for more than an equivalent amount of smaller errors.

The Root Mean Square Error (RMSE) is the square root of MSE yields, which has the advantage of having the same units as the quantity being estimated. The RMSE is defined as:

\[
RMSE = \sqrt{\frac{\sum (X_i - G_i)^2}{N}} = \sqrt{\frac{\sum e_i^2}{N}}
\] (4-3)

Where

\(X_i\) is the obtained data from other sources at time period \(i\)

\(G_i\) is the ground truth data at time period \(i\)

\(e_i\) is the difference(error) at time period \(i\)

\(N\) is the number of observations used in computing the \(RMSE\).
4.6 Results

4.6.1 Investigation of Ground Truth Data

Prior to comparing travel speed between data sources and ground truth, the ground truth was investigated. The first item examined was the sample size of the ground truth in comparison to the Bluetooth and RTMS data. Table 4-2 contains the brief overview of the sample sizes taken during October 18, 2013 from approximately 7:00 AM to 6:00 PM. LPR technology was able to capture 15 to 20 percent of the RTMS volume, while Bluetooth captured less than 5 percent. LPR also permits a microscopic examination of the traffic flow by tracking vehicles within the road lanes, allowing a better understanding of what the travel speed is being reported— the ‘fast’ or the ‘slow’ lane.

<table>
<thead>
<tr>
<th>Roadway Section</th>
<th>Data Source Sample Size</th>
<th>Percentage of RTMS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LPR</td>
<td>LPR*</td>
</tr>
<tr>
<td>A to C</td>
<td>2,758</td>
<td>4,811</td>
</tr>
<tr>
<td>A to B</td>
<td>3,431</td>
<td>6,835</td>
</tr>
<tr>
<td>B to C</td>
<td>4,885</td>
<td>8,671</td>
</tr>
</tbody>
</table>

* LPR data after the self-learning matching algorithm was applied.

** The Bluetooth/RTMS was calculated using the total sample size for section A to C.

With two LPR cameras installed in each location, four different flows were captured for one segment. As seen in Figure 4-4, the traffic flow “Fast to Fast” represents the highest speed along the study period. For easier interpretation, the outside lane is referred to as
Figure 4-4 Travel speed of ground truth data on segment ‘A to C’.
“Slow,” while the middle and inside lanes are referred to as “Fast.” About 15 minutes of data, starting at 13:30, are missing due to refueling power generators. The missing time periods were not included in the analysis. During PM peak, especially, there are two clearly separate flows along the same roadway. The first flow, “Fast to Fast” and “Fast to Slow”, is recovering to uncongested flow conditions, while the second flow, “Slow to Slow” and “Slow to Fast,” remains congested.

4.6.3 Visual Inspection of Each Data Source and the Ground Truth

By visually inspection the data error patterns exhibiting bias, range, and magnitude become more discernable than a single average value. This section will compare four data sources to the ground truth: Bluetooth, INRIX, HERE, and RTMS. In Figure 4-5, travel speed over time of each data source was plotted separately against the data of individual vehicles captured with LPR, along with the five minute moving average and 90% C.I. boundary for the segments.

The travel speed of Bluetooth is, overall, slightly above the LPR. During PM peak, the travel speed of Bluetooth data seems to follow a fast flow along the two separate speed flows, except for segment B to C, which seems to show that congested flow recovers to free flow at the same time on the roadway. One possible cause of this tendency is that the embedded module of Bluetooth technology could tend to recognize low speed flows as outliers since high speed flows were counted more than low speed flows. There may be other reasons such as measurement errors of distance, time, biased sampling by strength of Bluetooth signal, and locations where the devices were installed, etc.
Figure 4-5 Comparison of data sources vs. ground truth data (travel speed).
Travel speed of DP1 follows LPR relatively well during uncongested periods. Like Bluetooth, DP1 travel speed tends to be greater than LPR, which means travel speed provided by DP1 is higher than LPR most of the time. Also, during the PM peak, the travel speed of DP1 data seems to follow a fast flow along the two separate speed flows like all other data sources. Furthermore, the time when congested flow appears on DP1 data is later than LPR.

The overall visual investigation of Figure 4-5 reveals the travel speed of DP2 data to be closest to LPR. The travel time of DP2 is barely out of the boundary of 90% C.I. of LPR, except for the times of 7-8 and 9-10. There also seems to be relatively small time lag capturing the congested flow from the travel speed of DP2 compared to the data of LPR.

The travel speed of RTMS data appears to be consistently greater than the speed of LPR. It is very close to the upper boundary of 90% C.I. of LPR. This may be results of RTMS’s accuracy predominately being in traffic counts [38], while the RTMS speed results varied depending on the mounting location [39].

Figure 4-6 shows the distribution of the travel times for the data sources divided by travel time of ground truth. Note that there is a mildly significant portion of vehicles with values less than 0.5, which means their actual travel times were twice or even longer than the travel times recorded, for all data sources. DP2 seems to provide relatively more accurate data.

### 4.6.4 Root Mean Square Error Results

To examine the variation between the ground truth’s individual data and the moving average, the individual data used for comparison with all other data sources were also compared to the ground truth’s moving average (LPR*). Summary results show the overall accuracy of travel speed data using RMSE as the metric. Table 4-3 shows the RMSE values for all data source comparisons for the whole examination period.
Figure 4-6 Distribution of ‘travel time of data source vs travel time of ground truth’.
Table 4-3. RMSE of travel speed and travel time for all data sources.

<table>
<thead>
<tr>
<th></th>
<th>RMSE of Travel Speed (mph)</th>
<th>RMSE of Travel Time (sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>A to C</td>
<td>A to B</td>
</tr>
<tr>
<td>LPR*</td>
<td>8.7</td>
<td>8.9</td>
</tr>
<tr>
<td>Bluetooth</td>
<td>14.0</td>
<td>15.6</td>
</tr>
<tr>
<td>DP1</td>
<td>20.2</td>
<td>21.4</td>
</tr>
<tr>
<td>DP2</td>
<td>11.4</td>
<td>12.0</td>
</tr>
<tr>
<td>RTMS</td>
<td>20.4</td>
<td>21.9</td>
</tr>
</tbody>
</table>

*LPR* : Individual data of LPR were compared to a five minute moving average of LPR data at each vehicle to see how much variation could be identified for individual data of ground truth from their moving average.

In Figure 4-7, the inverse of RMSE was used for the z-axis unit, meaning that the higher (blue) bar represents better accuracy, and the values of RMSE were labeled accordingly. LPR was most accurate when compared to itself, proving high confidence in the ground truth source. It is more helpful visualizing the RMSE results using Figure 4-7 compared to Table 4-3. This is a very key consideration when presenting and understanding results. It is also important to evaluate more than the whole evaluation period. It is also beneficial to separate the dataset into speed bins, by time of day, or by level of congestion.

In summary, the travel speed data of DP2 yields the most accurate results in terms of RMSE compared to LPR data, except segment B to C where Bluetooth has less RMSE. The segment B to C did not have two separate flows along the same roadway as other segments had. RTMS and INRIX data yield greater RMSEs, almost as twice as DP2’s for segments A to C and A to B.
4.7 Conclusion

This study investigated multiple data sources to evaluate data accuracy as compared to LPR (ground truth.) By comparing the root-mean square errors of travel speed and time, the study found the performance ranking are in the order of HERE, Bluetooth, INRIX, and RTMS. The study’s main goal was to establish initial framework for determining the LPR as a viable ground truth option and the key considerations that must be accounted for while evaluating travel time and speed datasets.

Travel data is important in understanding and improving transportation systems. Over the past years the amount of traffic data has drastically increased. This growth is fundamentally changing data collections methods. By using LPR technology to gather ground truth a better capture of the traffic speed and time can be achieved. LPR allows
the user to collect lane by lane and actual identify vehicles travelling over the length of roadway. When LPR ground truth is combined utilized with the key considerations a more thorough and easy to understand evaluation of new data providers and old/current data collection methods is achieved.

In order to precede forward with future evaluations of real-time traffic data, more datum is needed. The increase in data would allow a more substantial analysis of additional metrics that could be used to further describe patterns and trends. Also, more investigation of ground truth, such as a range of measurement error and sample size, should be a key part of future studies. Furthermore, to make decision of overall data accuracy and data quality, multi criteria decision making method could be used such as Analytic Hierarchy Process (AHP) or Simple Multi-Attribute Rating Technique (SMART) [40].

Additional research is required to evaluate RTMS as a viable option for continuous evaluation of data providers like INRIX or HERE. Since most state agencies have already invested in RTMS type of technologies, some degrees of fusion of RTMS with other technologies may be beneficial in decision considerations. The results of the paper show a consistent difference between the ground truth and the RTMS speeds. It is possible that using LPR technology, RTMS could be calibrated to achieve a higher accuracy; therefore, becoming a constant data source for evaluation.
References


38. Middleton, D., D. Gopalakrishna, and M. Raman. *Advances in traffic data collection and management*. In *One of three white papers that support a series of workshops on Data Quality, which were held in March. 2003*.


CHAPTER 5. CONCLUSION

This dissertation compiled studies on enhancing a self-learning license plate matching algorithm and its utilization to capture ground truth travel times. The matching accuracy and learning speed efficiency of the algorithm are the major concern of this research. A series of simulations and field experiments were performed to support this research.

First, the beginning of the matching algorithm is examined to optimize learning speed and accuracy. It was found that the versatility of the starting association matrices leaves the user with the option of choosing a matrix that fits their application needs. For example, applications requiring minimal false matches may use a matrix with element value of 1000 in the main diagonal and 1 in the off-diagonals. While applications that are short on time and need a high number of matches with small error, may use an existing association matrix created from a large volume of matches. A future study is recommended to consider combined matrixes, e.g. using an alternate association matrix until a certain sample size is obtained and then using an unbiased starting association matrix.

Second, the experiment was conducted on derived association matrices to simulate their need at LPR stations. To evaluate the derived association matrix, we employ two experiments: to 1) determine when the matrix should be used and 2) evaluate the overall performance of license plate matching. The first experiment examined the closeness of the derived and direct association matrix to the ideal association matrix. The second simulated a case study of license plate matching that evaluated the performance of the derived and direct association matrix. As a result, derived association matrices are better for low traffic flows (small sample sizes) and/or poor LPR camera read rates.

Finally, a case study on probe vehicle technology was performed using a high-performing license plate recognition and matching system (97% matching rate with less
than 1% false-positives) to establish high-accuracy travel time ground truth at two sites for 12 hours or so each. The sites were chosen where a handful of other technologies were also deployed so that concurrent data could be collected and compared. The objective of this paper is to provide several key considerations for real-time traffic data evaluation for general cases, rather than a definite and specific conclusion. The key items include: obtaining reliable ground truth data, transforming and comparing incompatible datasets, and data quality evaluation measurements. A future study is recommended to further scrutinize the process of calculating ground truth travel times with additional datum and establish a detailed decision making method.

All together, the dissertation provides a drastic reduction in learning time, increase in matching accuracy at problematic LPR stations, and a strong understating of the key considerations when using LPR as ground truth.
VITA

Stephanie R. Hargrove was born and raised in New Smyrna Beach, Florida. In 2005, she graduated from the University of Tennessee, Knoxville with a BS in Civil Engineering. In 2007, Stephanie graduated with her Master's degree and a thesis titled ‘Commercial Vehicle Enforcement using License Plate Recognition Technology.’ From 2007 to 2010, Stephanie worked as a Transportation Engineer at Wilbur Smith and Associates. In 2015, she was granted a doctoral degree in Civil Engineering with concentration in Transportation Engineering at the University of Tennessee, Knoxville. As a Ph.D. student, Stephanie was recipient of the U.S. Dept. of Transportation’s Eisenhower Graduate fellowship twice, a surveying lab instructor and completed research for the National Transportation Research Center at Oak Ridge National Laboratory. During her graduate studies Stephanie was also a recipient of the National Science Foundation EAPSI fellowship for China, W.K McClure scholarship, WTS Helene M. Overly scholarship, and multiple Tennessee section Institute of Transportation Engineering scholarships. Her research interests include computational transportation science, transportation modeling, transportation policy, and operation research.