NTRCI

NATIONAL TRANSPORTATION
RESEARCH CENTER, INCORPORATED
UNIVERSITY TRANPORTATION CENTER

License Plate Recognition

This project was funded by the NTRCI University Transportation Center under a grant from the U.S. Department of Transportation Research and Innovative Technology Administration (#DTRT06G-0043)

Lee D. Han, Myong-KeeJeong, and Francisco Moraes Oliveira-Neto

TABLE OF CONTENTS

List of Figures

List of Tables

CHAPTER 1. INTRODUCTION

The purpose of this study is to assess and develop necessary technologies for tracking large trucks, via license plate recognition, in real time. The main application of such technologies, from the perspective of local government, is speed monitoring and enforcement. Yet other aspects such as homeland security, safety inspection, weight compliance, and vehicle profiling

CHAPTER 2. A BRIEF REVIEW OF LPR TECHNOLOGY

License Plate Recognition, or LPR, technology had its early enforcement application as early as 1979 when British Police Scientific Development Branch prototyped a system for trial deployment in Wokingham, UK. During the past decade, LPR technology saw wider real-time deployment as computer, communication, and video technologies matured.

LPR Technology primarily consists of six algorithms:

1. Plate locating;
2. Plate orientation and sizing;
3. Normalization;
4. Character segmentation;
5. Optical character recognition (OCR); and
6. Syntactical/geometrical analysis.

According to a study conducted by Han et al, plate locating is in itself a challenge to start with. Because an average of 4% of all trucks on I-40 have no license plates for recognition, the best one could hope for is an average of 96% recognition rate. However, as shown in figure 1, most LPR cameras are aimed at the bumper area and this effectively reduces the rate further to about 92%.

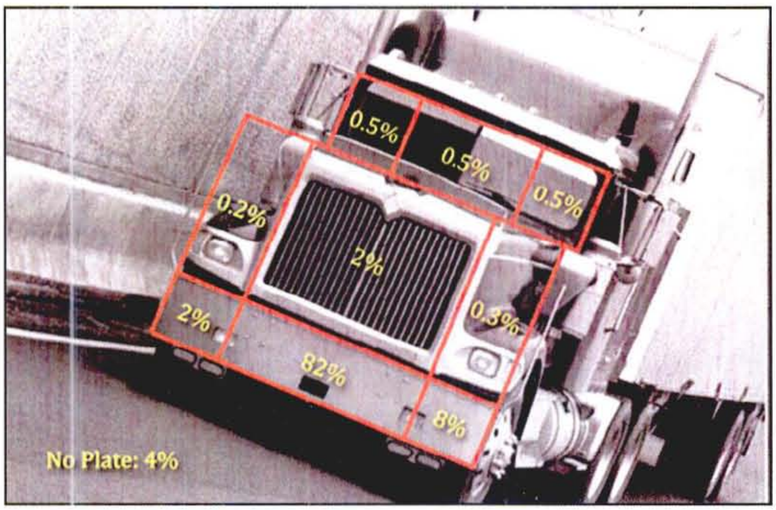

Figure 1. Illustration. Mounting Locations of Truck License Plates.

Compounded with other challenges such as the following, LPR may sometimes see 70% or even lower accuracy.

- Motion blur and camera vibration;
- Poor lighting and visibility;
- Bent, damaged, dirty, and modified plates;
- Lack of unified fonts and colors;
- Reflectivity of plates and paints; and
- Plate designs "unfriendly" to LPR

Examples of some "less friendly" designs are shown in figures 2-7. Some of the Florida plates have a dark vertical bar in the middle, which often makes it difficult for LPR. Several states use vertically stacked characters, e.g. "PWR" on Kansas plates. This practice is troubling when OCR algorithm is applied. Minnesota plates can have stacked and small characters at the front as well as the back of a plate, which further challenges LPR. Many states use plates with good contrast. However, there are states that use light green (e.g. North Carolina) or red (e.g. Massachusetts) letterings, which are difficult because the frequencies of light sensitive to these colors are quite different from one another. Michigan's dark blue background is not helpful for LPR application either.

Figure 2. Photograph. Example of Florida License Plate.

Figure 3. Photograph. Example of Kansas License Plate.

Figure 5. Photograph. Example of Minnesota License Plate.

Figure 4. Photograph. Example of North Carolina License Plate.

Figure 6. Photograph. Example of Massachusetts License Plate.

Figure 7. Photograph. Example of Michigan License Plate.

As shown in figure 8, the rates of LPR for correctly recognizing plates from various states reflects the fact that states with "less friendly" designs generally suffer lower accuracy.

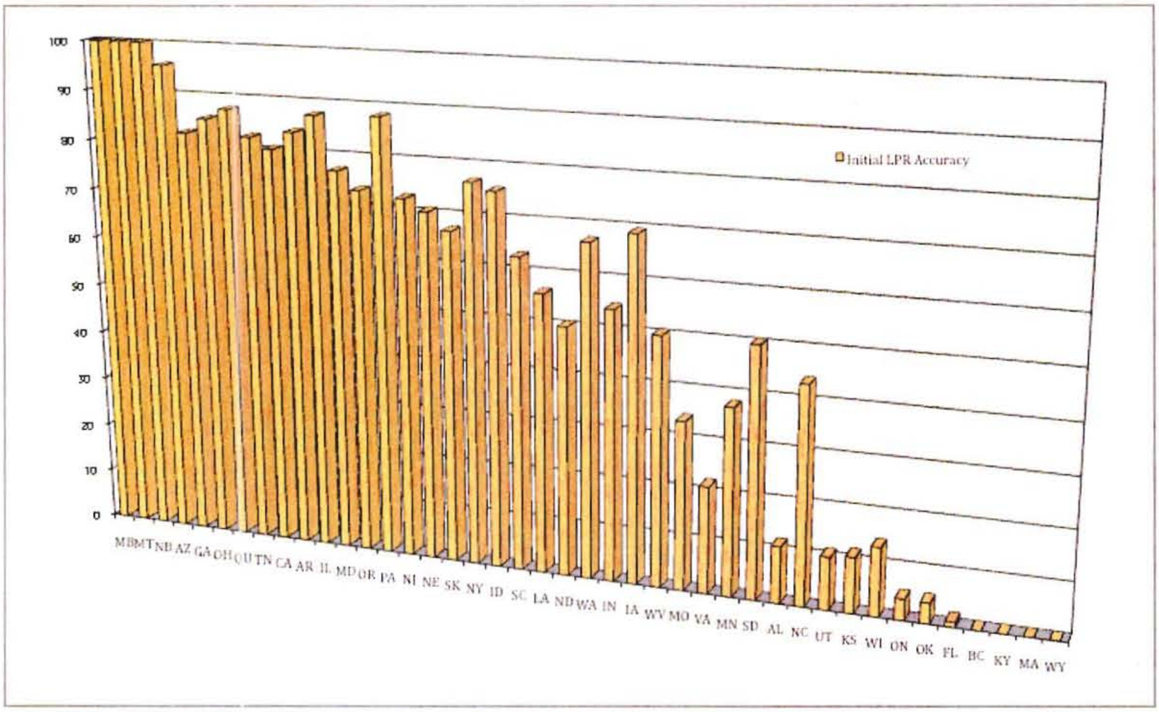

Figure 8. Bar Chart. License Plate Recognition Accuracy Based on Each State's License Plate Design.

CHAPTER 3. THE CHALLENGES OF PLATE MATCHING

To measure truck speed with LPR technology, it is essential that the same plate, e.g. truck, is identified and matched at two, or more, locations. Given the aforementioned not always perfect performance of LPR technology, it is questionable how many trucks would have their plates read correctly both times for a correct match.

If two LPR units were deployed along a certain Interstate corridor, say point 1 and point 3 in figure 9, and all westbound trucks going through 1 are going though 3 also, with a sub-par correct recognition rate around 60% each, one can do some quick and simple estimations to have a feel for the potential matching rates.

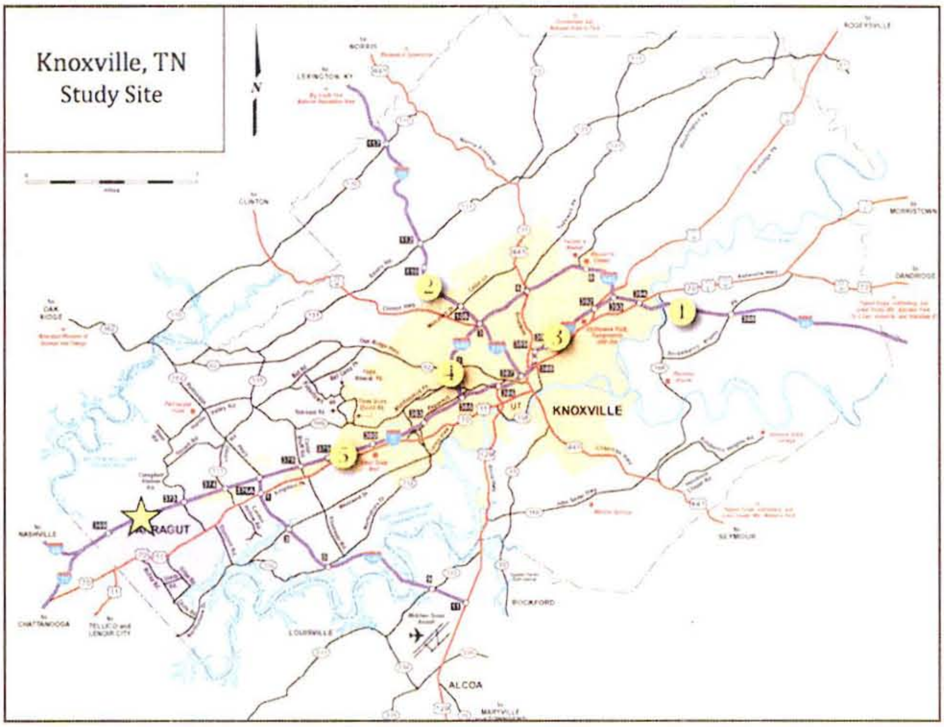

Figure 9. Map. Knoxville, Tennessee Study Site.

For the case that the correct reading rates are totally random and independent at the two locations, one would expect a correct matching rate of a meager 36%, or 60% × 60%, which is not exactly encouraging. On the other hand, if the two units read the identical 60% of plates correctly, a matching rate of "at least" 60% is attainable. We emphasize "at least" because while 60% are correctly read and matched, those incorrectly read by LPR units may still have a chance of getting matched correctly. For example, the plate in figure 10 may or may not be read correctly at either LPR station.

Figure 10. Photograph. Example of Tennessee License Plate.

Obviously when the plate is read correctly at both stations, we have a match; when it is read correctly at one and incorrectly at the other, we do not have a match. Then there is a slight chance when the plate is read incorrectly at both sites, we might still have a match if the two LPR units erred in the same fashion.

Table 1. Scenarios of Matched and Mismatched Plates from Two LPR Stations.

LPR Station 1	LPR Station 2	
	Correctly Read	Incorrectly Read
Correctly Read	85217HY≡85217HY	85217HY≠B5217HY
		85217HY≠B5217HY
Incorrectly Read	85217HV≠85217HY	85217HY≠BS217HV
	BSZITNV≠85217HY	B5217HY≡B5217HY

The range of matching rate for this simplistic scenario is roughly between 36 and 60%. To improve on this less-than-desirable rate, our study employs text-mining techniques, which is presented in Chapter 6.

CHAPTER 4. FIELD WORK AND INITIAL ANALYSES

In the summer of 2007, a group of graduate and undergraduate students at the University of Tennessee were recruited and trained on the University of Tennessee campus for collecting LPR data in the field (figures 11-13).

After several practice and test data collection sessions at a single site, the students went to the test sites by I-40 in west Knoxville and collected multiple sets of data (figure 14). Among them, five sets of data are used for later data analysis and plate matching purposes. Video of the license plates was also collected for the purpose of ground-truthing and for verifying the accuracy and reliability of LPR units.

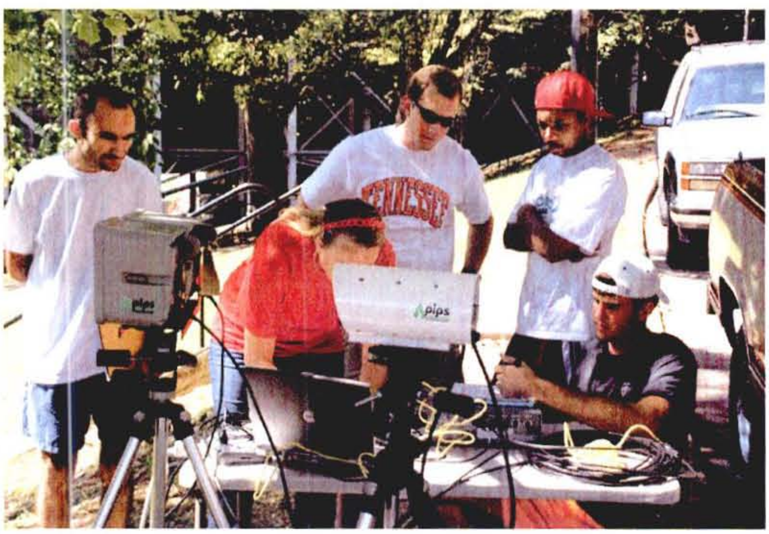

Figure 11. Photograph. Students Receiving LPR Training.

Figure 12. Photograph. Students Receiving LPR Training.

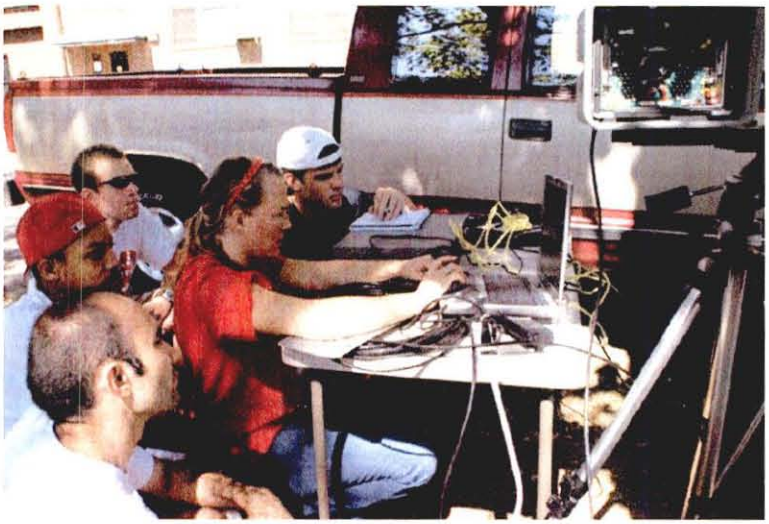

Figure 13. Photograph. Students Receiving LPR Training.

Figure 14. Photograph. On-Site LPR Data Collection.

The two locations chosen for data collection are in West Knoxville at I-40 Campbell Station Road Exit and at the Weigh Station Exit (figures 15-16). With a distance just slightly over one mile, the sites are close enough to ensure a large number of trucks going through both sites and to allow more efficient project management.

The downstream LPR site at the Weigh Station is near the weigh-in-motion (WIM) sensor. This is to simulate a potential future deployment strategy when license plate information could be used, in addition to weight information, for directing trucks to either the static scale for further inspection or to bypass.The five sets of "coupled" LPR data are summarized in table 2. The reported license plate numbers, as read by the LPR units, were compared with the ground truth as extracted by the students after the field session. The results are also tabulated.

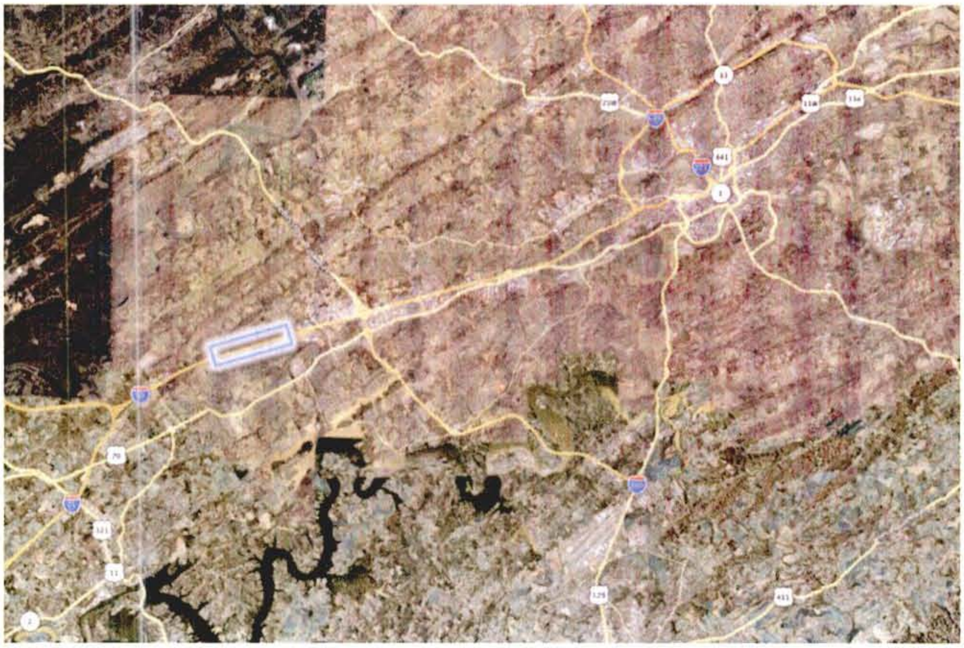

Figure 15. Map. LPR Data Collection Zone.

Figure 16. Photograph. LPR Data Collection Zone.

Table 2. Summary of 5 Sets of "Coupled" LPR Data.

Dataset	Location	Image Captured	Plates Captured	Plates Read Correctly	LPR Accuracy
1	Campbell Station	417	343	218	63.6%
	Weigh Station	251	233	138	59.2%
2	Campbell Station	420	381	239	62.7%
	Weigh Station	401	388	245	63.1%
3	Campbell Station	787	695	439	63.2%
	Weigh Station	365	358	223	62.3%
4	Campbell Station	858	731	418	57.2%
	Weigh Station	290	281	176	62.6%
5	Campbell Station	662	521	311	59.7%
	Weigh Station	276	270	174	64.4%
Overall	Campbell Station	3144	2671	1625	60.8%
	Weigh Station	1583	1530	956	62.5%

Several things are immediately notable:

1. As mentioned earlier, LPR technology is not always perfect. Furthermore, our study used "mobile" setup, e.g. LPR units on tripods. These are professional-grade heavy tripods, but tripods nonetheless. Comparing to "fixed" or "permanent" installations common to the usual LPR implementations, a mobile setup tends to vibrate more and may "drift" from the proper alignment as a result of the shockwaves from all the high-speed truck traffic.
2. For different conditions, locations, and data collection periods, the LPR accuracy stayed relatively stable with an average accuracy of just under 62% and a standard deviation of about 2%.
3. The numbers of "image captured" and "plates captured" are not quite the same. In fact, the number of image captured should always be more than the number of plates captured as a result of how LPR works. The LPR camera snaps about 30 video frames per second, which usually include more than one frame of each license plate. The LPR algorithm then decides which of these frames yield the best "read" and discards the other ones. From time to time more than one of these frames may be used and are considered captured due to a number of reasons, which we'll not get into here. But the actual number of plates captured is less than the number of images.

CHAPTER 5. PLATE MATCHING

With LPR plate information from both locations, Campbell Station and Weigh Station, already digitized by the LPR machines, plate matching can potentially be conducted in real-time if the information were transmitted and processed at a central location. For this study, however, the plate matching effort was performed after the fact and in the laboratory for this study.

Of the 2,671 plates captured at Campbell Station site and the 1,530 at the Weigh Station site, 858 of them were found, manually, to be from identical trucks. In other words, if the plates of all these 858 trucks were correctly recognized by the respective LPR units at both sites, we would have a matching rate of 100%. However, as we reported earlier, we only have a 62%, or so, correct recognition rate, so the matching rate is not likely to be that high. In fact, out of the 858 plates, only 507 were correctly matched, which puts the matching rate at about 59%.

At this point, the research team's goal is to improve the matching rate. The challenge here is to match plates that were not read correctly by the LPR units.

The plates that were not read correctly by LPR units at first were deemed incorrectly read even if they had only one misread character. In other words, there might be enough information, five or six correctly recognized characters, for making a good match. It should be noted though that in a real-time operational situation, we would not know which plates are recognized correctly by the LPR units. Therefore, some probabilistic issues need to be taken into consideration.

Based on tens of thousands of LPR reading results, a matrix representing the probability of one character recognized as another was constructed. This matrix, in table 3, shows that any character as reported by LPR has a probability of being correct or incorrect. For example, if LPR says a character "0," (zero), was recognized, 96.7% of the time the LPR is correct. Although, there exist possibilities that the character was actually an "8," or "9," or "O," (Oh).This is the case for most characters, as the value of the cells along the diagonal of the matrix is usually the largest in each row and somewhere near 100 (percent).

However, there are several characters that are more challenging to read correctly than others. One example is "B." When a "B" is reported by LPR, only about 60% of the time it is correct. Near 30% of the time the "B" should have been an "8;" some other times the "B" should have been a "9" or an "H."

The worst case is letter "O," (Oh). When LPR reports letter "O," according to the matrix, only 12.5% of the time it is correct. More than 75% of the time it should have been "0," (zero). With these insights, the team set about to improve the matching rate.

Tabie 3. Matrix Representing the Probability of One Character Being Recognized as another Character.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	
0	96.7	0.0	0.0	0.0	0.0	0.0	0.3	0.0	0.9	0.4	0.0	0.0	0.1	0.8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.2	0.0	0.5	0.0	0.0	0.0	0.1	0.0	0.1	0.0	0.0	0.0	100
1	0.5	89.5	0.2	0.9	0.6	0.1	0.1	1.7	0.2	0.2	0.2	0.1	0.1	0.0	0.0	0.0	0.0	0.2	1.7	1.3	0.0	0.1	0.0	0.1	0.0	0.2	0.0	0.0	0.0	0.4	0.0	0.0	0.2	0.4	1.1	0.0	100
2	0.1	0.1	93.4	0.1	0.1	0.1	0.0	0.4	0.1	0.1	0.0	0.0	0.0	0.0	0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.1	0.0	0.0	0.0	0.0	0.0	0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	5.3	100
3	0.1	0.3	0.4	97.1	0.3	0.1	0.1	0.2	0.2	0.9	0.0	0.0	0.0	0.1	0.0	0.0	0.0	0.3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100
4	0.8	0.2	0.2	0.2	94.2	0.9	1.0	0.0	0.1	0.1	0.5	0.0	0.1	0.1	0.1	0.0	0.0	0.2	0.0	0.0	0.3	0.2	0.1	0.0	0.0	0.0	0.0	0.1	0.0	0.0	0.1	0.1	0.0	0.3	0.0	0.3	100
5	0.0	0.0	0.1	0.5	0.0	97.9	0.4	0.0	0.2	0.5	0.0	0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.1	0.0	0.0	0.0	0.2	0.0	0.0	0.0	0.1	0.0	0.0	0.0	100
6	0.4	0.2	0.1	0.4	0.5	0.8	91.2	0.0	2.6	0.5	0.1	0.1	0.2	0.0	0.2	0.7	0.9	0.0	0.0	0.0	0.3	0.0	0.1	0.0	0.0	0.1	0.0	0.0	0.0	0.0	0.0	0.1	0.3	0.1	0.0	0.1	100
7	0.0	0.1	0.8	0.7	0.2	0.1	0.1	95.2	0.0	0.3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.7	0.0	0.0	0.0	0.0	0.0	0.1	0.0	0.0	0.3	1.5	100
8	1.1	0.1	0.1	0.0	0.0	0.2	0.1	0.0	96.0	0.5	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.6	0.0	0.0	0.0	0.0	0.0	0.2	0.0	0.0	0.0	0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100
9	0.2	0.1	0.0	0.2	0.2	0.1	0.0	0.2	0.5	98.0	0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.1	0.0	100
A	0.4	1.8	0.7	0.0	3.3	0.4	1.8	0.7	3.6	0.0	76.0	0.7	0.0	0.0	0.0	0.0	0.0	0.0	0.4	0.4	2.5	1.8	0.4	1.8	0.4	0.4	0.0	0.7	0.0	0.0	0.4	0.0	0.4	0.7	0.0	0.4	100
B	0.6	1.8	0.0	0.0	0.0	0.0	0.6	1.2	29.2	2.3	0.0	60.2	0.6	0.0	0.6	0.0	0.0	2.3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100
C	1.0	0.0	0.0	0.5	0.0	0.0	0.0	0.0	0.5	0.0	0.0	0.0	93.6	2.0	0.5	0.0	1.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.5	0.0	0.0	0.0	0.0	0.0	100
D	16.6	0.0	0.0	0.0	0.0	0.6	0.0	0.0	0.0	0.0	0.0	0.6	0.0	80.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.6	0.0	0.6	0.6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100
E	0.8	0.0	1.6	0.8	0.0	0.0	0.0	0.8	0.8	0.8	0.0	0.0	0.8	0.8	82.0	0.0	0.0	0.0	0.0	0.0	2.5	0.0	0.0	0.0	0.0	6.6	0.0	0.0	0.0	0.0	0.8	0.0	0.0	0.0	0.8	0.0	100
F	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.8	92.5	0.0	0.8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	4.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.8	0.8	100
G	0.9	0.0	0.0	0.0	0.0	0.0	24.1	0.0	4.5	0.0	0.0	0.9	0.9	0.0	0.0	0.0	62.5	0.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	2.7	0.9	0.0	0.9	0.0	0.0	0.9	0.0	0.0	0.0	0.0	100
H	0.0	0.0	0.2	0.0	0.0	0.0	0.2	0.0	0.0	0.0	0.0	0.0	0.0	0.2	0.0	0.0	0.0	98.4	0.0	0.0	0.0	0.2	0.6	0.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.2	0.0	0.0	0.0	100
I	1.2	34.4	0.0	2.0	0.4	0.0	0.0	2.8	0.4	0.8	0.0	0.0	0.0	0.4	0.8	0.8	0.0	0.4	39.9	1.2	0.0	1.6	0.8	0.4	0.0	1.2	0.0	0.8	0.0	2.4	1.6	1.2	0.0	0.4	4.0	0.4	100
J	2.0	10.8	0.7	4.7	0.0	0.7	0.0	0.0	0.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.7	76.4	0.0	0.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	2.0	0.0	0.0	0.0	0.0	0.7	100
K	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.8	0.0	0.0	0.0	0.0	0.0	92.1	0.0	0.8	2.4	0.0	0.0	0.0	0.8	0.0	0.0	0.0	0.0	1.6	1.6	0.0	0.0	100
L	1.2	8.3	0.0	1.2	1.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.2	0.0	0.0	0.0	0.0	0.0	84.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.2	0.0	0.0	0.0	1.2	0.0	0.0	100
M	1.9	0.0	0.0	1.9	0.0	0.0	0.0	0.0	1.9	0.0	0.0	0.0	0.0	1.9	0.0	0.0	0.0	3.8	0.0	0.0	3.8	0.0	69.8	11.3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.8	0.0	0.0	0.0	100
N	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	5.5	0.0	0.0	0.0	0.0	0.0	93.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100
O	76.9	1.0	0.5	0.0	0.0	0.0	0.0	0.0	0.5	0.0	0.0	0.0	0.0	3.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	12.5	0.0	4.8	0.0	0.0	0.0	0.5	0.0	0.0	0.0	0.0	0.0	100
P	0.0	0.2	0.2	0.0	0.0	0.0	0.0	0.2	0.2	0.4	0.0	0.0	0.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	98.2	0.0	0.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100
Q	26.7	0.0	0.0	0.0	0.0	0.0	13.3	0.0	0.0	6.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	6.7	0.0	46.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100
R	0.0	0.0	0.0	0.0	0.0	0.0	1.3	1.3	5.3	1.3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	2.6	0.0	0.0	0.0	0.0	0.0	0.0	86.8	0.0	0.0	0.0	0.0	0.0	1.3	0.0	0.0	100
S	0.5	0.0	1.6	3.8	0.0	26.3	0.5	1.1	0.5	4.8	0.0	0.0	0.5	0.0	0.0	0.0	0.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	59.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100
T	0.0	2.1	0.0	3.1	0.0	0.0	0.0	8.3	0.0	1.0	0.0	0.0	0.0	0.0	0.0	2.1	1.0	2.1	1.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	69.8	0.0	0.0	0.0	0.0	8.3	0.0	100
U	5.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.5	1.5	0.0	1.5	0.0	0.8	0.0	0.8	0.0	0.0	0.0	0.0	0.8	0.8	0.0	0.0	0.0	0.0	86.9	0.0	0.0	0.0	0.0	0.0	100
V	0.8	0.0	0.0	0.8	0.0	0.8	0.0	0.0	0.0	2.3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	2.3	88.5	0.0	0.0	4.6	0.0	100
W	0.0	0.0	0.0	1.1	0.0	0.0	0.0	0.0	0.0	0.0	1.1	0.0	0.0	1.1	0.0	0.0	0.0	4.3	0.0	0.0	2.2	0.0	7.6	9.8	0.0	0.0	0.0	1.1	0.0	0.0	0.0	1.1	70.7	0.0	0.0	0.0	100
X	0.0	0.0	0.0	1.2	0.0	0.0	1.2	1.2	1.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	94.1	1.2	0.0	100
Y	0.0	0.4	0.0	0.0	0.0	0.0	0.0	0.9	0.0	0.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.4	0.0	0.9	0.4	0.4	95.6	0.0	100
Z	0.0	0.0	3.4	0.0	0.0	0.0	0.0	1.5	0.0	0.0	0.0	0.0	0.3	0.0	0.0	0.0	0.0	0.0	0.3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.3	0.0	0.0	0.0	0.0	0.0	94.2	100

In general, the team used the probability matrix and several other text-mining algorithms to improve the matching rate. The most notable among all approaches used is the Levenshtein Distance, or Edit Distance method, as well as several enhancement algorithms developed by this research team for it.

The Edit Distance, or ED, between any two plates is the minimum number of edit operations required to transform one string, $\boldsymbol{x}$, into the other, $\boldsymbol{y}$. These fundamental edit operations include:

- **Substitution** – A character in string $\boldsymbol{x}$ is replaced by another character at the same location in string $\boldsymbol{y}$. For instance there is a substitution operation between "85217HY" and "B5217HY".
- **Insertion** – A character in string $\boldsymbol{y}$ is inserted into string $\boldsymbol{x}$ and thereby increases the length of string $\boldsymbol{x}$ by one character. For example, one needs to perform an insertion to match "8527HY" and "85217HY"
- **Deletion** – A character in string $\boldsymbol{x}$ is deleted and thereby reduces the length of string $\boldsymbol{x}$ by one. To match "85217HY" and "85217H," such an operation is needed.

Let's say a plate number "4455HZ" was captured at two LPR stations. Let string x = "4455IIZ" and string y = "4455HZ", and our task is to compute the minimum number of edit operations required to transform x into y. One can obtain $d(x,y)$, or the minimum number of operations, as 2 for this case. The two operations correspond to the substitution of the first "I" in string x by "H" and the deletion of the second "I" in string x. Note that we could have assigned x and y reversely and the edit distance would remain the same. Furthermore, in real-time implementation the ground truth would be unknown; that is, we would not know if either string were read correctly. Nevertheless, we can still employ the edit distance algorithm to try to find a match.

To understand why two is the minimum number of operations to transform string x into y in our example, imagine the two strings disposed in a two dimensional grid, as shown in figure 18, with string x horizontally displayed on top and string y vertically displayed on the left. To match string x to string y, there are numerous possible combinations and permutations of edit operations that can be taken. Two of these possibilities are represented as paths from the upper left corner to the lower left corner. Our goal here is to find the least or minimum number of operations to accomplish the matching.

The tick marks for the two axes are formed by the corresponding sequences of characters, with string x beside j axis and string y beside i axis. We define a move in the grid to be represented by a link that ends up in a point associated with two characters (x_{i_k}, y_{i_k}). A diagonal downward move represents a substitution, a horizontal move to the right represents a deletion, and a vertical downward move represents an insertion. Also, each node of the grid is associated with a function $\gamma(i_k, j_k)$, which measures the general cost, or distance, of each move along the grid. For the original idea of ED, this cost is set to 1, for insertions and deletions, and in the case of substitutions, 0, if the corresponding characters are identical ($x_{i_k} = y_{i_k}$), or 1, otherwise. If we walk from the origin point (0,0) to the end point (i_m, j_m) on the grid, each potential path is associated with an overall cost d defined as (figure 17):

$$d(i_m, j_m) = \sum_{k=0}^{n} \gamma(i_k, j_k)$$

Figure 17. Equation. $d(i_m, j_m)$.

Where, n is the number of nodes along a path between $(i_0, j_0) = (0,0)$ and $(i_m, j_m) = (|x|, |y|)$; $|x|$ and $|y|$ are the lengths, or the numbers of characters of x and y, respectively.

As an example, consider two paths (drawn by dashed and solid lines) reaching the point (i_m, j_m) as shown in figure 18. Computing the number of editing operations represented by the two paths, we have $d_{solid}(i_m, j_m) = 2$ and $d_{dashed}(i_m, j_m) = 6$.

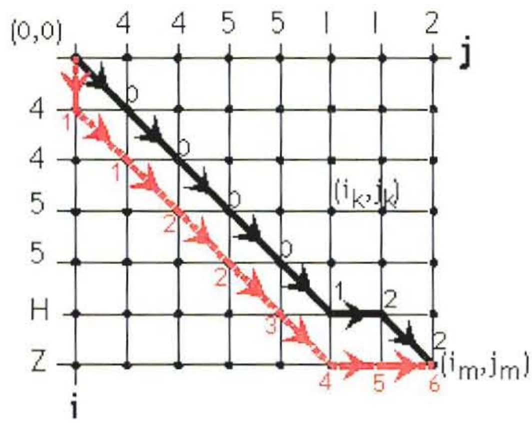

Figure 18. Graph. An Illustration of Alternative Matching Paths with Different Levenshtein Distances.

To obtain the shortest, and hence best, path, one might consider exhausting all possible paths. This, however, is computationally costly. We therefore employed Dynamic Programming techniques per Wagner and Fisher to solve for $d(x, y) = \min\{d(i_m, j_m)\}$. An in depth discussion on this subject can be found in Duda's book on Pattern Classification.

In many applications, string y is provided by a list of words, which has the maximum likelihood of containing the "true" value of a given string x. This pre-specified list of words is called lexicon or reference for matching purposes. Using this list of words, it is possible to detect errors, generate candidate corrections and rank these candidates. In this study, on the other hand, strings x and y represent results from field license plate recognition units with sub-100% correct reading rates. As such, neither string x nor string y can be used as a reference for matching purposes, since, for real-time operations, there is no truly reliable knowledge about the "ground truth."

In the long run as FMCSA is developing a comprehensive inventory of all large trucks, a lexicon of truck license plates, as well as a wealth of additional information, will become available. As a result, increased matching speed and matching rate can be expected.

Obviously, when two plates have an ED of zero, or $d(x, y) = 0$, we have a match. To discount the occasional errors from LPR units, we might define a match to be a situation when ED is less or equal to 1 (or 2, or even larger numbers), we would certainly correctly match some plates that were missed previously. The issue here now is we would also introduce some false positive cases into the system. So the idea is to devise an algorithm, or algorithms, for maximizing the matching rate without significantly increasing the false positive rate. A few actual false positive examples for the case of ED ≤ 2 are shown in table 4.

Table 4. Comparison of Weigh Station LPR Unit and Campbell Station LPR Unit.

Weigh Station LPR Unit			Campbell Station LPR Unit		
Time	LPR Report	Ground Truth	Time	LPR Report	Ground Truth
13:30:50	"5587"	"5587"	13:28:46	"15S7"	"15157"
14:46:37	"1297D"	"12990"	14:43:37	"12905"	"12905"
15:19:36	"1234"	"1234"	15:16:57	"9214"	"9214"
15:14:12	"2JZ294"	"2JZ294"	15:13:21	"2JF204"	"2JF204"
14:14:39	"9713"	"9713"	14:13:13	"9214"	"9214"

Tabulated in table 5 are results from several different approaches we investigated in this study. The simple match method has been reported already. ED, or Edit Distance method, is rather straightforward also. Results from ED can be further improved with the use of Travel Time (TT) constraints, which will reduce the overall pool each plate is compared with and, hence, reduce the spurious matches.

Table 5. Results from Several Different Approaches Investigated.

Algorithm	Potential Matches	Correct Matchs	Incorrect Matches	Correct Match Rate
Simple Match	858	507	0	59%
ED ≤ 1	858	690	58	80%
ED ≤ 2	858	785	677	91%
ED ≤ 1 + TT	791	643	4	80%
ED ≤ 2 + TT	791	721	16	91%
GED≤ 0.9	858	749	110	87%
CED≤ 0.9	858	743	59	87%
GED≤ 1.0	858	766	152	89%
CED≤ 1.0	858	762	101	89%
GED≤ 1.0 + TT	791	740	8	94%
CED* ≤ 0.9	858	742	33	86%
CED* ≤ 1.0	858	761	33	89%
WED ≤ ?	Currently under development.			

Two other approaches, General ED (GED) and Constrained ED (CED), were also developed with increased matching accuracy. For GED, instead of counting each insertion, substitution, etc., identically as having an exact Levenshtein distance of 1.0, we "shortened" the distance for some pairs of characters. For instance, while it is quite unlikely for a reported "D" to have been actually a "3," and thus a justified distance of 1.0, there is a very real chance, with empirical statistics as evidence, for it to be actually a "0" (zero) or "O" (oh). As such, an edit distance less than 1.0 should be used to reflect the likelihood of the mistake. This shorter distance, or GED, was developed based on the comprehensive 36 by 36 matrix presented a couple of pages above.

Similarly, we also developed a CED algorithm constraining the number of certain types of editing operations. For instance, when we assign near zero distances between "0" (zero) and "O" (oh), "1" and "I", "2" and "Z", and so forth, we should limit the number of such editing operations to avoid extreme cases. Such limitation minimizes the number of spurious matches and computer search/calculation time. Once again, the use of travel time constraint, TT, also helps reduce the incorrect matches.

However, one may not be able to rely on TT for certain operational conditions, such as when the two LPR units are separated by a longer distance or when the tracking and matching of the two plates is more than just for the purpose of speed estimation. To this end, we developed a CED* algorithm, which uses Constrained ED, but allows only candidates with minimal editing operations.

As shown in the result table, these algorithms have improved the matching rate from 59% to near 90%, while keeping the false matching rate in check. Considering the actual LPR accuracy is only around 62%, the result is not only encouraging, but also enabling for future enforcement applications.

CHAPTER 6. LARGE TRUCK SPEEDS

While a real-time, high performance plate-matching algorithm has many useful applications ranging from law enforcement to traffic management to national security, we would like to go back to the initial intent of this study, which was to provide a tool for automated speed monitoring and enforcement/deterrent purposes. Obviously, with the known distance between the two LPR units and the synchronized system time stamps, the calculation of the travel speed of each matched truck is trivial. But before deploying a real-time system, which is the intent of Phase B of this study, we should take a close look at if and how trucks are slowing down as the result of the reduced speed limit that was enacted in 2006.
Available to this study are several speed data sets for the test site:

- 1997 Tube: a set of truck speed data collected by Han et al using coupled tubes in 1997. The data are for trucks in all lanes. (Only the cumulative speed distribution curve is available for display.)
- 2005 RTMS: a set of general traffic data collected by a TDOT contractor using radar detectors simulating inductive loops (firing at near 90-degree angle).
- 2007 Radar: a set of speed data collected by Han et al with a radar gun for trucks in the slow lane.
- 2007 LPR: a set of truck speed data derived from the LPR plate matching effort detailed in this report. The speed data are for trucks that were in the slow lane at the Campbell Station Road site.
- 2008 Radar Slow: a set of speed data collected by Han et al with a radar gun for trucks in the slow lane.
- 2008 Radar Mid: a set of speed data collected by Han et al with a radar gun for trucks in the middle lane.

Of these speed data sets, all are for trucks except the RTMS data, which used an assumed average vehicle length to estimate average vehicle speed. In addition to the fact that RTMS speeds are not for trucks only, the TDOT contactor that used these RTMS units also suggested that the speed data from these RTMS units exhibit an unstable and unreliable quality. As such, this study will not include RTMS. The rest of the data sets are plotted and presented in figures 19-20.

Using Logistic regressions, we were able to also approximate the cumulative speed distribution curves for all datasets (figure 20). It is evident that even though the average truck speed has declined since the reduction of posted speed in 2006, the effect is not as pronounced as anticipated. In fact, if one looked at the speed distributions closely, the percentage of speeding trucks has only increased. Without going into detailed statistical discussion, we will go over several observations:

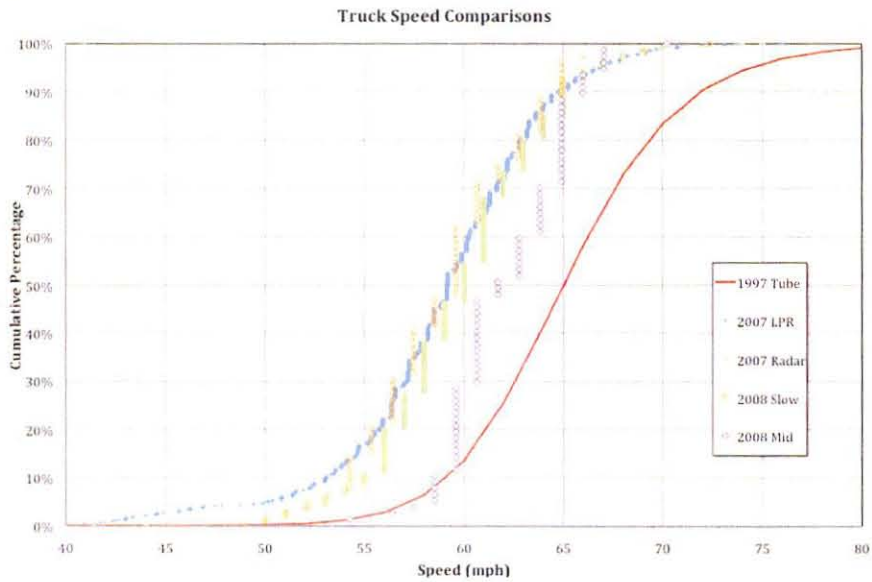

Figure 19. Graph. Truck Speed Comparisons.

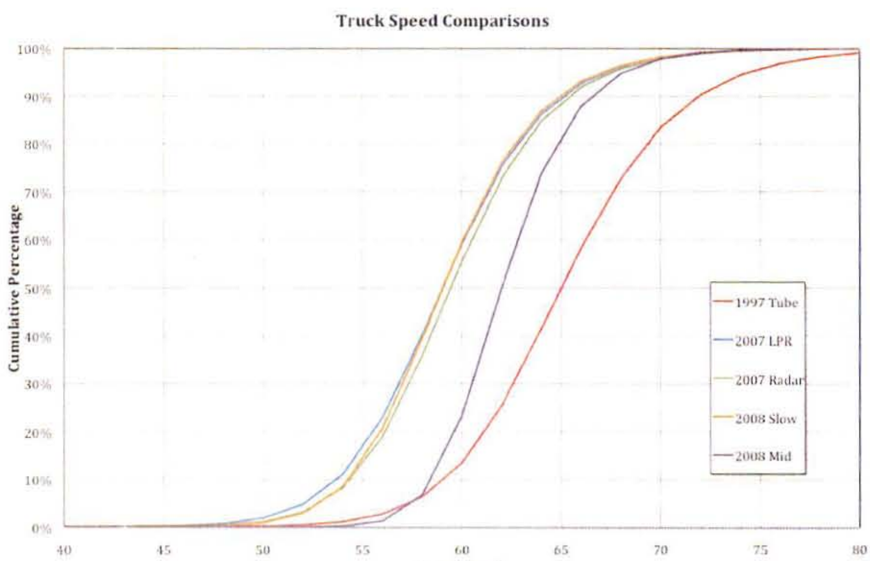

Figure 20. Graph. Truck Speed Comparisons.

Table 6. Comparison of Trucks Speeding vs. Not Speeding.

Percentage		1997 Tube	2007 LPR	2007 Radar	2008 Slow	2008 Mid
Not Speeding		50%	18%	13%	14%	2%
Speeding	all speeding trucks	50%	82%	87%	86%	98%
	at least 5 mph faster	17%	42%	44%	41%	77%
	at least 10 mph faster	4%	11%	12%	10%	19%
	at least 15 mph faster	1%	3%	3%	2%	4%
	at least 20 mph faster	< 1%	< 1%	< 1%	< 1%	< 1%

- The 2007 LPR and 2007 Radar speed data show great similarity. This lends more credibility to the resultant LPR speeds. Bear in mind that while the LPR speeds are averaged over the distance from site 1 to site 2, the radar data was taken as point speed at Campbell Station Road. Therefore, the two distributions are not expected to be identical.
- The 2007 Radar, for the slow lane, and the 2007 Slow data sets are quite similar. Therefore, the speeding habit of the truckers has not changed or been discouraged by enforcement, or the lack of, during this time.
- Even though trucks in the middle lane were expected to travel at a higher speed, the comparison between 2008 Slow and 2008 Mid is still noteworthy. It is shocking to see that 98% of, or almost all, trucks in the middle lane were speeding.
- As mentioned earlier, the percentage of trucks speeding has increased significantly after the speed limit decreased from 65 to 55 mph. The average speed (time-mean speed) of all trucks only decreased from 66.3 mph to 61.8 mph during this time. One might argue that the 4.5 mph reduction with very limited speed enforcement could be considered effective. The other side of the coin, though, is the average truck speed did not decrease 10 mph corresponding to the 10 mph speed limit reduction. As a result, the expected 18% reduction in NOx, was not realized.

CHAPTER 7. CONCLUSIONS

As reported herein, this study has successfully accomplished the proposed tasks and concluded that automated truck speed enforcement on Interstate highways using license plate recognition technology (LPR) is highly feasible, even when LPR performance is, at times, less than desirable. The points below seem to suggest that we should move forward towards deploying this approach.

- Many metropolises reduced the posted speed limit for large trucks for the purposes of air quality and safety;
- To derive benefits from such action, the reduced speed limit has to be enforced;
- A large-scale enforcement of the new speed limit is often extremely challenging due to fiscal and human resource constraints;
- Large trucks are required to enter weigh stations;

- By tracking these trucks at various locations along the Interstate highways, warnings and citations could be issued when they stop at the weigh station;
- Most tracking systems are either too expensive, non-universal, or still in development;
- All trucks are required to have license plates;
- Even though LPR technology is not perfect, the text-mining algorithms developed in this study can match a high percentage of truck plates and, also, determine their speeds;
- It would be desirable to proceed with a real-time LPR large truck tracking and speed monitoring study before the eventual deployment.

CHAPTER 8. REFERENCES

- Duda, R. O., P. E. Hart, and D. G. Stork (2000). Recognition with Strings. Pattern Classification.2nd Ed. Wiley Inter Science. Ch. 8,p. 413-420.
- Han, L.D., F.J. Wegmann, and A. Chatterjee (1997). Using License Plate Recognition Technology for Transportation Study Data Collection, project report submitted to Tennessee Department of Transportation, Nashville, TN.
- Levenshtein V.I. (1966). Binary Codes Capable of Correcting Deletions, Insertions, and Reversals. Soviet Phys. Dokl, 10(8), pp. 707-710.
- Tang, T, M. Roberts, and C. Ho (2003). Sensitivity Analysis of MOBILE6 Motor Vehicle Emission Factor Model, Federal Highway Administration, FHWA Resource Center, Atlanta, GA.
- Wagner, R. A. and M. J. Fischer (1974). The String-To-String Correction Problem. J. Association Computer Machinery, Vol. 21, No. 1, pp. 168-173.