MID-AMERICA
TRANSPORTATION CENTER

Report # MATC-MS&T: 296

Final Report
WBS:25-1121-0003-296

University of Nebraska Lincoln

K-STATE
Kansas State University

KU
THE UNIVERSITY OF
KANSAS

MISSOURI
S&T
University of
Science & Technology

LINCOLN
University

University of Missouri

IOWA STATE
UNIVERSITY

THE UNIVERSITY OF IOWA

Freeway Travel Time Estimation using Existing Fixed Traffic Sensors – A Computer-Vision-Based Vehicle Matching Approach

Zhaozheng Yin, Ph.D.
Assistant Professor
Department of Computer Science
Missouri University of Science and Technology

Wenchao Jiang
Ph.D. Student
Missouri University of Science and Technology

Haohan Li
M.S. Student
Missouri University of Science and Technology

MISSOURI
S&T
University of
Science & Technology

2015

A Cooperative Research Project sponsored by
U.S. Department of Transportation-Research
and Innovative Technology Administration

The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. This document is disseminated under the sponsorship of the Department of Transportation University Transportation Centers Program, in the interest of information exchange. The U.S. Government assumes no liability for the contents or use thereof.

Freeway Travel Time Estimation using Existing Fixed Traffic Sensors – A Computer-Vision-Based Vehicle Matching Approach

Zhaozheng Yin, Ph.D
Assistant Professor
Department of Computer Science
Missouri University of Science and Technology

Wenchao Jiang
Ph.D. student
Department of Computer Science
Missouri University of Science and Technology

Haohan Li
M.S. student
Department of Computer Science
Missouri University of Science and Technology

A Report on Research Sponsored by

Mid-America Transportation Center

University of Nebraska-Lincoln

March 2015

Technical Report Documentation Page

1. Report No. 25-1121-0003-296	2. Government Accession No.	3. Recipient's Catalog No.	
4. Title and Subtitle Freeway Travel Time Estimation using Existing Fixed Traffic Sensors – A Computer-Vision-Based Vehicle Matching Approach		5. Report Date March 2015	
		6. Performing Organization Code	
7. Author(s) Zhaozheng Yin, Wenchao Jiang and Haohan Li		8. Performing Organization Report No. 25-1121-0003-296	
9. Performing Organization Name and Address Mid-America Transportation Center 2200 Vine St. PO Box 830851 Lincoln, NE 68583-0851		10. Work Unit No. (TRAIS)	
		11. Contract or Grant No.	
12. Sponsoring Agency Name and Address Research and Innovative Technology Administration 1200 New Jersey Ave., SE Washington, D.C. 20590		13. Type of Report and Period Covered August 2013 – December 2014	
		14. Sponsoring Agency Code MATC TRB RiP No. 34786	
15. Supplementary Notes			
16. Abstract Vehicle re-identification is investigated as a method to analyze traffic systems, such as the estimation of travel time distribution in a freeway network. In this paper, a vision-based algorithm is proposed to match vehicles between upstream and downstream videos captured by low-resolution (360*240) surveillance cameras and then estimate the travel time distributions. The algorithm consists of three stages: (1) vehicles are detected by Motion History Image (MHI) and Viola-Jones vehicle detector, and then image segmentation and warping are conducted to the detected vehicle images; (2) features (e.g., size, color, texture) are extracted from vehicle images to uniquely describe each vehicle in low-resolution images; and (3) vehicles from two cameras are matched by solving two problems: a Support Vector Machine (SVM) classifies whether a pair of vehicles are identical or not, and linear programming globally matches groups of vehicles between upstream and downstream cameras with context constraints. The proposed algorithm was validated on two sections of freeway in St. Louis, Missouri, United States, which outperforms the state-of-the-art methods and accurate travel time estimation is achieved based on the re-identification results.			
17. Key Words Vehicle re-identification, Vehicle matching, Low-resolution cameras, Camera network.		18. Distribution Statement	
19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 29	22. Price

Table of Contents

List of Figures

List of Tables

List of Abbreviations

Support Vector Machine (SVM)
Motion History Image (MHI)
Histogram of Oriented Gradient (HOG)
Local Binary Pattern (LBP)
Scale Invariant Feature Transform (SIFT)
High Definition (HD)
Missouri Department of Transportation (MoDOT)
Mid-America Transportation Center (MATC)
Vehicle Re-identification (VRI)
Gaussian Mixture Model (GMM)
Standard Deviation Signature (SDS)

Ackowledgements

This study was leveraged by the research project entitled “Freeway Travel Time Estimation using Existing Fixed Traffic Sensors – A Computer-Vision-Based Vehicle Matching Approach,” which was supported in part by the Missouri Department of Transportation (MoDOT) and Mid-America Transportation Center (MATC).

Disclaimer

The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. This document is disseminated under the sponsorship of the U.S. Department of Transportation's University Transportation Centers Program, in the interest of information exchange. The U.S. Government assumes no liability for the contents or use thereof.

Abstract

Vehicle re-identification is investigated as a method to analyze traffic systems, such as the estimation of travel time distribution in a freeway network. In this paper, a vision-based algorithm is proposed to match vehicles between upstream and downstream videos captured by low-resolution (360*240) surveillance cameras and then estimate the travel time distributions. The algorithm consists of three stages: (1) vehicles are detected by Motion History Image (MHI) and Viola-Jones vehicle detector, and then image segmentation and warping are conducted to the detected vehicle images; (2) features (e.g., size, color, texture) are extracted from vehicle images to uniquely describe each vehicle in low-resolution images; and (3) vehicles from two cameras are matched by solving two problems: a Support Vector Machine (SVM) classifies whether a pair of vehicles are identical or not, and linear programming globally matches groups of vehicles between upstream and downstream cameras with context constraints. The proposed algorithm was validated on two sections of freeway in St. Louis, Missouri, United States, which outperforms the state-of-the-art methods and accurate travel time estimation is achieved based on the re-identification results.

Chapter 1 Introduction

Vehicle Re-identification (VRI) is critical to track vehicles in a transportation network with distributed sensors. By tracking vehicles in a network, important traffic parameters such as travel time can be obtained, which are of great value for traffic engineers for detecting traffic jams, controlling traffic variability, and designing future transportation networks.

A variety of technologies have been investigated for VRI. Detailed algorithms vary in accordance with the sensors based on which VRI is implemented. These sensors include induction loop sensors [1, 2], bluetooth [3], wireless magnetic sensors [4], and video cameras and so on. A typical vehicle re-identification procedure consists of three stages: vehicle detection, feature extraction, and vehicle matching. The accuracy of vehicle detection, the availability of features, and the selection of a matching algorithm all have important effects on the robustness of a VRI system.

1.1 Related Work

This paper focuses on the vision-based VRI algorithm, which is one of the most straightforward and intuitive techniques that can be used to re-identify the same vehicle as it moves between two sensors. This type of technique has been extensively researched due to the prevalence of surveillance cameras installed above traffic roads [5, 6, 7, 8, 9, 10]. Vehicles are easily re-identfied by the plate number in Ozbay et al. [5], although Wang et al. [6] took a different approach by extracting a color histogram, Histogram of Oriented Gradient (HOG), and aspect ratio as vehicle features in their study. Later, Jiang et al. [7] added Local Binary Pattern (LBP) to improve the accuracy. To deal with the constantly changing viewpoint, Hou et al. [8] calibrated vehicles' poses by using 3D models of the vehicles.

Although these methods have achieved relatively good performances, they all rely on the availability of high-resolution cameras. When dealing with low-resolution cameras, Sumalee et al.

[9] was able to achieve only 54.75% re-identification precision in videos with a resolution of 764*563 pixels. Sun et al. [10] attempted to mitigate these camera limitations by combining vision-based and induction loop sensor-based vehicle features to re-identify vehicles.

A variety of matching algorithms have been developed; their main differences lie in the way they define the probability of one vehicle being identical/different to another. Wang et al. [6] directly incorporated the weighted sum of feature distances as the probability of a pair of vehicles being identical. Kamijo et al. [11] took a different approach, performing dynamic programming on two sequences of vehicles passing between upstream and downstream cameras to identify individual vehicles. However, this method required that the order of vehicles remains relatively unchanged. Tawfik et al. [1] defined a threshold for each feature distance and used a dicision tree cascade framework to determine whether two vehicles were identical, while Sumalee et al. [9] and Cetin et al. [12] both used a Bayes-based probabilistic technique to fuse vehicle features for the re-identification decision.

There are two main drawbacks to all previous vehicle matching algorithms: (1) the threshold and weight for each feature are usually manually determined, and (2) the vehicle pairs may not be linearly separable in the feature space. This is important because most of the previous work has depended on linear decision models.

1.2 Challenges

The challenges related to vision-based vehicle re-identification can be summarized as follows:

1. In low-resolution camera images, a vehicle may be represented by a relatively small number of pixels. General visual features such as Histogram of Oriented Gradients (HOG) [13], Local Binary Pattern (LBP) [14], and Scale Invariant Feature Transform (SIFT) [15] will not work well since these local-statistics-based features tend to be

inaccurate when there are insufficient pixels.

2. The lighting conditions under which the cameras operate may change considerably over time, which may cause the color of a pair of identical vehicles to appear different when viewed by upstream and downstream cameras.

3. The viewpoints inevitably vary between the upstream and downstream cameras, resulting in marked variations in the vehicle's texture.

The above challenges mean that the identification of reliable visual features for low-resolution vehicle images is vital. These features should be invariant to both illumination and viewpoint. Meanwhile, because of the limited information provided by low-resolution vehicle images, a more effective matching strategy is required to clearly classify identical/different vehicle identities.

Note, although some of the challenges can be mitigated by choosing High Definition (HD) cameras, it increases the hardware cost and bandwidth cost to transmit the videos. For example, there are 300+ traffic surveillance cameras existing in the St. Louis area, and the video streams from TransSuite to us have a resolution of 360*240 pixels. The project collaborating with the Missouri Department of Transportation (MoDOT) and Mid-America Transportation Center (MATC) aims to attack the vehicle re-identification problem from the software side using the existing hardware.

1.3 Proposal and Contributions

The objective of this paper is to introduce a vison-based vehicle re-identidication algorithm from videos captured by two low-resolution and non-overlapping cameras. Additionally, challenges such as illumination and viewpoint changes are considered in this paper. Finally, travel time distribution between two camera locations is estimated based on the re-identification results. The contributions of this paper are three-fold:

1. In this paper, vehicles are detected by Motion History Image (MHI) and Viola-Jones vehicle detectors. The influence of illumination change is mitigated by MHI. After warping the upstream and downstream videos with homograph matrices, their viewpoints are calibrated. Features including the size, color, and texture information are extracted from warped vehicle images. This specially-designed procedure works well for vehicle detection and feature extraction in low-resolution videos.
2. Rather than fusing features by linear weighted summation, a clearer gap is found to separate identical and different vehicles by the Support Vector Machine [24], which is a strong classifier mapping the original training data to a hyperplane, thus resulting in a non-linear and more robust decision model.
3. A global optimization problem is formulized that extends the assignment framework illustrated by Cetin et al. [12] to a more general model in which the miss detection of vehicles and vehicles entering or exiting the section between two cameras is considered.

The rest of the paper is organized as follows. The next section presents the problem formulization and system overview. Detailed descriptions of the VRI system, including vehicle detection, feature extraction, and matching strategy, are described in sections 3-5, respectively. The test results are discussed in section 6. The paper ends with conclusions and future work.

Chapter 2 Problem Statement and Overview of the System

Vehicle Re-identification (VRI) essentially resolves the following mathematical problem:

$$\text{Upstream Vehicle Set: } U = \{U_1, \dots, U_i, \dots, U_I, \mathrm{U}_\varphi\}$$

$$\text{Downstream Vehicle Set: } D = \{D_1, \dots, D_j, \dots, D_J, \mathrm{D}_\varphi\}$$

$$\text{Reidentification: } \{U_1, \dots, U_i, \dots, U_I, U_\varphi\} \leftrightarrow \{D_1, \dots, D_j, \dots, D_J, D_\varphi\}$$

where U_i and D_j are the i_{th} and j_{th} vehicle captured by upstream and downstream cameras within the same time interval, respectively. U_φ and D_φ are special void objects, different from U_i or D_j, that can be used to map unmatched vehicles. Two different aspects of this re-identification problem are important.

Firstly, it is a **classification** problem. A category must be assigned to each vehicle pair (U_i, D_j), namely either "U_i and D_j are identical" or "U_i and D_j are different." From another point of view, this is a **mapping** problem. For each upstream vehicle in the set U, we need to find the most identical and unique vehicle in D, the downstream set, and vice versa. Because some vehicles may be misdetected in the downstream or exit before reaching the downstream camera, vehicles in U may not correspond to any mapped vehicle in D, so these vehicles are mapped to a void object D_φ. Similarly, vehicles in D with no matched vehicles in U are mapped to the void object U_φ. All mappings involving D_φ and U_φ can be many-to-one mapping; the remaining elements in sets U and D are all one-to-one mappings.

Based on this mathematical formulation, the VRI algorithm introduced in this paper consists of three phases:

1. **Vehicle Detection**. Videos are recorded from upstream and downstream cameras simultaneously and vehicles are detected by Motion History Image (MHI) and Viola-Jones detectors. The detected vehicle images are preprocessed using segmentation and warping techniques.
2. **Feature Extraction**. Features including vehicle size, color, and texture information are extracted from the vehicle images. A feature distance vector describing the similarity of each pair of vehicles is then obtained.
3. **Vehicle Matching**. The classification problem illustrated above is solved by a Support Vector Machine (SVM), while the mapping problem is regarded as a global optimization problem with some constraints and can be solved by linear programming.

Chapter 3 Vehicle Detection

Detecting vehicles in each frame is the initial critical step of a Vehicle Re-identification (VRI) system. This step is especially challenging when the lighting conditions and viewpoints vary among cameras. For each frame of a video, such as the one shown in figure 3.1a, the technique of Motion History Image (MHI) [16] is first adopted to detect image regions with moving pixels. The results of this process are shown in figure 3.1b. The Viola-Jones vehicle detector [17] is then applied to find the precise positions of the vehicles (fig. 3.1c) within the regions of moving objects. To further locate the boundaries of the vehicles, segmentation is applied to separate the foreground (vehicles) from the surrounding background (fig. 3.1f). Finally, the vehicle image is warped to the viewpoint directly facing the lanes (fig. 3.1g), thus mitigating the problem of viewpoint differences between the upstream and downstream cameras. The same process is applied to both the upstream and downstream videos.

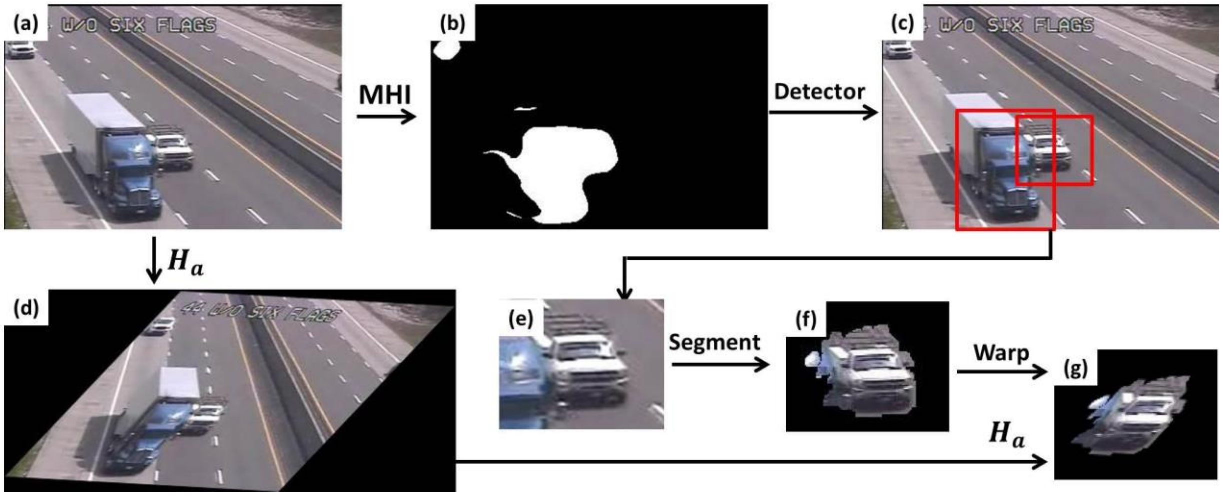

Figure 3.1 Flowchart for vehicle detection: (a) a frame in a video, (b) moving object detection result for MHI, (c) positions of vehicles detected by Viola-Jones detector, (d) warped image of (a), (e) one cropped vehicle image, (f) vehicle image after eliminating background, (g) warped vehicle image

Motion History Images (MHI) and the Viola-Jones detector perform complementary functions in the proposed algorithm to detect vehicles. Although MHI is an efficient way to find moving objects, as figure 3.1b shows, it is difficult for MHI to determine whether the moving regions consist of one or two vehicles when two vehicles are adjacent to each other or one vehicle is moving with its own shadow. Thus, the Viola-Jones detector is applied to moving regions to determine the accurate positions of vehicles within the candidate regions without the need to search in other impossible regions. In addition, MHI is more resistant to illumination changes than ordinary background subtractions such as the Gaussian Mixture Model (GMM) (18). Figure 3.2a shows a frame when the illumination is changing due to clouds moving across the sun. Figure 3.2b shows the moving object detection result obtained using GMM, which is unsatisfactory because GMM updates the background at constant time intervals, but the illumination level changes relatively rapidly and in an unpredictable way due to the intermittent cloud cover. MHI solves this problem by applying a forward and backward decaying background subtraction [16]. Figure 3.2c shows the results of the MHI moving object detection. The influence of the changing background illumination is removed and the moving objects are clearly detected.

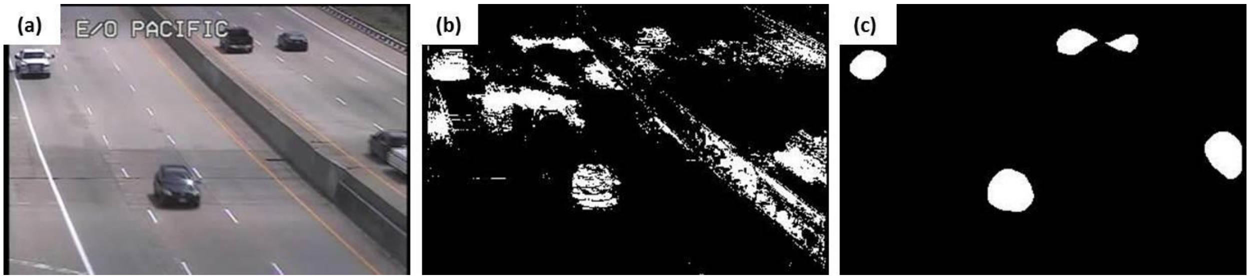

Figure 3.2 Advantage gained by using MHI: (a) original image, (b) GMM detection results, (c) MHI detection results

The detailed information that can be extracted from vehicle images is limited by the low camera resolution, and the background surrounding the vehicles can worsen this problem because it adds noise to the feature extraction. Thus, removing the background from vehicle images is of great significance for extracting valuable features. Here, we consider pixels outside the boundary of a vehicle to be background while pixels within the boundary are part of the foreground image and adopt the Graph-Based Image Segmentation algorithm [19] to rule out the background (fig. 3.1f).

To mitigate the problem of different viewpoints between upstream and downstream cameras, as shown in figures 3.1a and 3.1d, the original image is warped to the viewpoint directly facing the lanes by a homography matrix, $\mathrm{H_a}$, using the warping method proposed in Kanhere et al. [20]. Once H_a is determined it remains fixed because the camera is stationary. $\mathrm{H_a}$ is then applied to every detected vehicle image, as shown in figure 3.1g, thus ensuring that every vehicle is viewed from the same viewpoint.

Chapter 4 Feature Extraction

The vehicle image (template) is the raw vehicle feature. Other features, such as size, color, and texture can be extracted for vehicles. Let V_{U_i} and V_{D_j} denote the warped and background-eliminated vehicle images for vehicles U_i and D_j, respectively. For each vehicle in U, a feature set F_{U_i} can be formed to describe U_i. The corresponding feature set for downstream vehicles is denoted as F_{D_j}. Without loss of generality, only F_{U_i} is described in this section.

4.1 Size Feature

Because the viewpoints and resolutions of the upstream and downstream cameras are warped to be the same, this preprocessing implicitly normalizes vehicles between different cameras and lanes, which simplifies their size comparison. The number of foreground pixels is a good way to estimate the size of U_i, which is denoted as S_{U_i}.

4.2 Color Feature

To fully extract the color information of a vehicle, two color models are adopted: original Hue-Saturation-Illumination (HSI) histograms ($C^H_{U_i}$,$C^S_{U_i}$, $C^I_{U_i}$), and normalized hue histograms ($C^{NH}_{U_i}$).

For original HSI histograms, the three image channels are treated separately. Each channel is divided into 20 bins, thus U_i has three 20-bin histograms: $C^H_{U_i}$,$C^S_{U_i}$, $C^I_{U_i}$, corresponding to the H, S, and I color channels, respectively. However, the original HSI histograms may vary in response to illumination changes. This problem can be mitigated if the illumination and saturation information is normalized. To achieve this, a 20-bin normalized hue histogram, denoted as $C^{NH}_{U_i}$, is added to extract the pure hue information of a vehicle image [21].

Original HSI histograms are satisfactory when illumination conditions are stable. Figure 4.1 shows an example of what happens when the illumination changes. Vehicles **a** and **b** are identical while **c** is different. The original hue histograms of **a** and **b** are quite different, but after normalization, their hue histograms show that their color is similar. However, color information is not sufficient to distinguish between different vehicles of similar colors (e.g., **c**’s color histograms are similar to **a**’s and **b**’s), so texture information is also required.

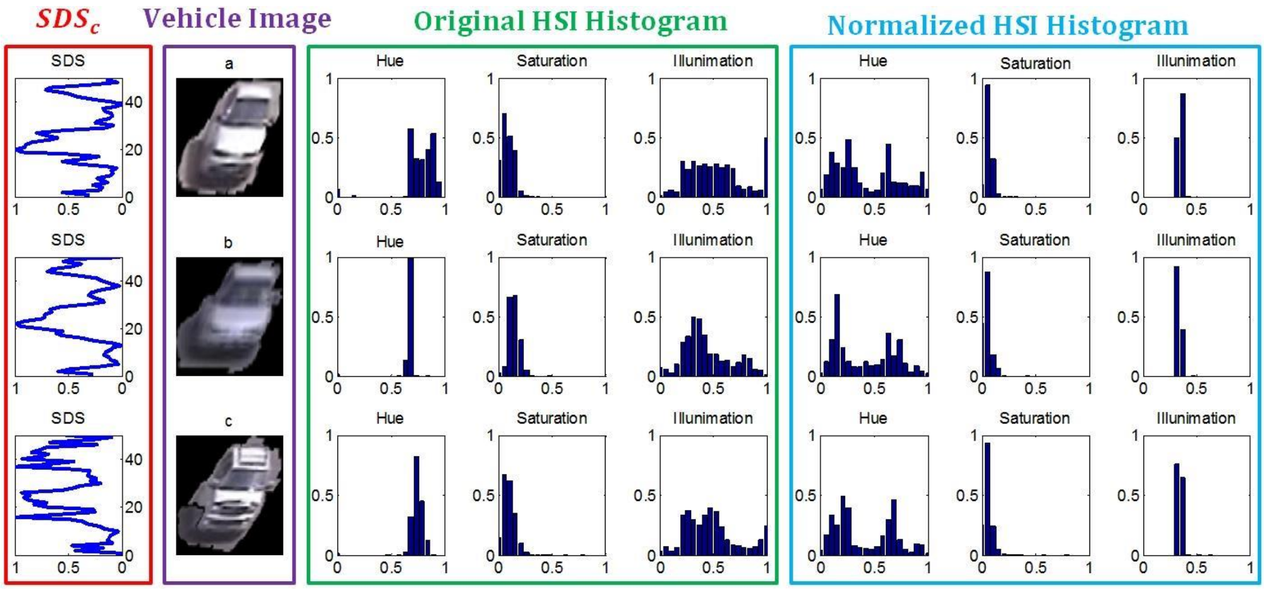

Figure 4.1 Vehicle images and their standard deviation signature (SDS), original HSI histograms, and normalized HSI histograms: a and b are identical while c is different

4.3 Texture Feature

Classical texture descriptors (e.g., HOG, LBP) work poorly in low resolution vehicle images. We therefore propose a standard deviation-based texture descriptor. As shown in figure 4.1, vehicle **a** has no sunroof, but vehicle **c** does. This difference can be described by the standard deviation in the roof regions of the vehicles. Mathematically, the standard deviation signature (SDS) is a one-dimensional vector with the m_{th} dimension equaling the standard deviation of the foreground pixels in the m_{th} row or column of V_{U_i}. Row SDS of U_i is denoted as $\mathrm{T}_{U_i}^{SDS_r}$, while column SDS is denoted as $T_{U_i}^{SDS_c}$. Thus, the m_{th} dimension of $\mathrm{T}_{U_i}^{SDS_r}$ is defined as

$$\mathrm{T}_{U_i}^{SDS_r} = \left(\frac{1}{cols-1}\sum_{n=1}^{cols}\left(V_{U_i}(m,n) - \frac{1}{cols}\sum_{n=1}^{cols} V_{U_i}(m,n)\right)^2\right)^{\frac{1}{2}} \tag{1}$$

where $V_{U_i}(m,n)$ is the grayscale pixel value in row m and column n of V_{U_i}. $cols$ is the column number of foreground area. $T_{U_i}^{SDS_c}$ can be calculated in a similar way, but in the column direction.

The red box in figure 4.1 shows $T_{U_i}^{SDS_c}$ for each of the three vehicles. It is clear that the signatures of **a** and **b** are similar, while the signature of **c** is much rougher because **c** involves more textural differences.

4.4 Feature Distance

The feature sets F_{U_i} and F_{D_j} are the sets of all features extracted from U_i and D_j, respectively:

$$F_{U_i} = \left\{S_{U_i}, C_{U_i}^H, C_{U_i}^S, C_{U_i}^I, C_{U_i}^{NH}, T_{U_i}^{SDS_r}, T_{U_i}^{SDS_c}, V_{U_i}\right\} \tag{2}$$

$$F_{D_j} = \left\{S_{D_j}, C_{D_j}^H, C_{D_j}^S, C_{D_j}^I, C_{D_j}^{NH}, T_{D_j}^{SDS_r}, T_{D_j}^{SDS_c}, V_{D_j}\right\} \tag{3}$$

where the raw vehicle images V_{U_i} and V_{D_j} are used to calculate the template distance. Thus, the feature distance describing the similarity of vehicle pair $\left(U_i, D_j\right)$ is denoted as

$$\mathrm{DIS}\left(U_i, D_j\right) = \left[DIS_S, DIS_{C^H}, DIS_{C^S}, DIS_{C^I}, DIS_{C^{NH}}, DIS_{T^{SDS_r}}, DIS_{T^{SDS_c}}, \mathrm{DIS}_{Temp}\right] \tag{4}$$

Each dimension of $\mathrm{DIS}\left(U_i, D_j\right)$ is the distance of the corresponding feature of $F_{U_i}$and F_{D_j}. Based on the properties of size, color, and texture features, different distance metrics are used in the feature distance vector.

The size distance DIS_S of a vehicle pair $\left(U_i, D_j\right)$ is defined as

$$DIS_S = \frac{S_{U_i}}{S_{D_j}}. \tag{5}$$

The color distance DIS_C of a vehicle pair $\left(U_i, D_j\right)$ is defined as

$$\mathrm{DIS}_C = 1 - \left(1 - \frac{\sum_{\mathrm{k}=0}^{20}\left(\mathrm{C}_{U_i}(\mathrm{k}) \times C_{D_j}(\mathrm{k})\right)^{\frac{1}{2}}}{20\left(\mu(\mathrm{C}_{U_i}) \times \mu\left(\mathrm{C}_{\mathrm{D}_j}\right)\right)^{\frac{1}{2}}}\right)^{\frac{1}{2}} \tag{6}$$

where k is the k_{th} dimension of the histograms and $\mu(*)$ is the arithmetic mean function. Equation 6 is applied to DIS_{C^H}, DIS_{C^S}, DIS_{C^I}, and $DIS_{C^{NH}}$.

The length of the standard deviation signature of a vehicle pair may not be the same, but they can be normalized by linear interpolation. The texture distance DIS_T of $\left(U_i, D_j\right)$ is defined as their covariance,

$$DIS_T = \frac{\mu\left(T_{U_i}T_{D_j}\right) - \mu(T_{U_i})\mu(T_{D_j})}{v\left(T_{U_i}\right)^{\frac{1}{2}} v\left(T_{D_j}\right)^{\frac{1}{2}}} \tag{7}$$

where $v(*)$ is the variance function. Equation 7 is applied to both $DIS_{T_{SDS_r}}$ and $DIS_{T_{SDS_c}}$.

Besides the feature distances described above, a template distance is also adopted based on the grayscale pixel subtraction of V_{U_i} and V_{D_j}. The binary images L_{U_i} and $L_{D_j}$are obtained with foreground pixels equaling 1 and background pixels equaling 0. Then the maximal value $B\left(U_i, D_j\right)$ of two-dimensional cross correlation between L_{U_i} and L_{D_j} is calculated as

$$\mathrm{B}\left(U_i, D_j\right) = \max_{(\mathrm{k},\mathrm{l})} \sum_{\mathrm{m}=0}^{\mathrm{M}-1} \sum_{\mathrm{n}=0}^{\mathrm{N}-1} \mathrm{L}_{U_i}(\mathrm{m},\mathrm{n}) L_{D_j}(\mathrm{m}-\mathrm{k},\mathrm{n}-\mathrm{l}) \tag{8}$$

where M and N are the length and width of each of the binary images, separately. If $(\mathrm{k}_{\mathrm{max}}, \mathrm{l}_{\mathrm{max}})$ can maximize equation 8, then the template distance of $\left(U_i, D_j\right)$ is defined as

$$\mathrm{DIS}_{Temp} = \frac{\sum_{\mathrm{m}=1}^{\mathrm{M}} \sum_{\mathrm{n}=1}^{\mathrm{N}} \left| V_{U_i}(\mathrm{m},\mathrm{n}) - V_{D_j}(\mathrm{m}-\mathrm{k}_{\mathrm{max}}, \mathrm{n}-\mathrm{l}_{\mathrm{max}}) \right|^2}{255 * 255 * \mathrm{B}\left(U_i, D_j\right)} \tag{9}$$

$\mathrm{DIS}\left(U_i, D_j\right)$ describes the similarity relationship between a vehicle pair $\left(U_i, D_j\right)$ and serves as the input for SVM, which will be discussed in the next section.

Chapter 5 Vehicle Matching

As noted in the second section, two problems need to be solved: classification and mapping. This section presents details showing how the two problems are solved by using the Support Vector Machine (SVM) and linear programming techniques.

5.1 SVM-based Classification

SVM has been successfully applied to solve a number of classification problems [22, 23]. The primary objective of SVM is to find a gap between two categories (namely positive and negative), and this gap should be as wide as possible. For a linear inseparable problem, SVM maps the original finite-dimensional space into a much higher-dimensional space, making the separation linear in that space and thus obtaining better classification results.

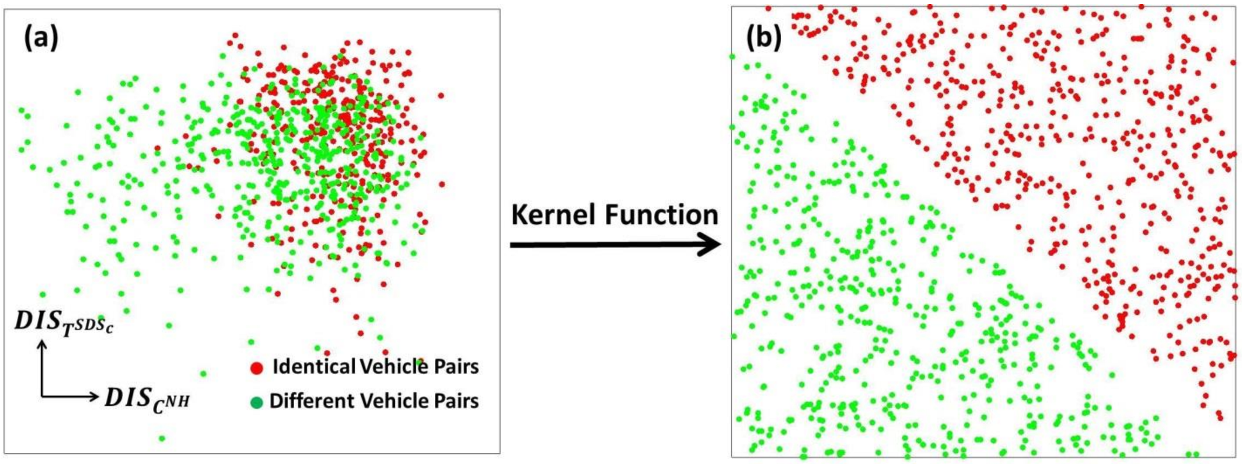

Figure 5.1 SVM for two category linear inseparable classification: (a) original feature space, (b) higher feature space

Interested readers can refer to Scholkopf and Smola’s work [24] for a full mathematical analysis of SVM. Here, we will instead consider an example to illustrate how SVM works in practice. As shown in figure 5.1, the X axis is $DIS_{C_{NH}}$ for a vehicle pair and the Y axis is $DIS_{T_{SDS_c}}$

for the same vehicle pair. We begin by randomly choosing 500 identical vehicle pairs (red points) and 500 different vehicle pairs (green points). In figure 5.1, these are not linearly separable. However, after the original feature space is mapped into a higher dimensional feature space with a kernel function, the two categories can be easily separated (fig. 5.1).

For a standard SVM framework, there are two stages: training and classification. In the training stage, we label the positive samples (identical pairs) as Arabic numeral 1 while negative samples (different pairs) are labelled as -1. A SVM classifier is trained based on $\mathrm{DIS}(U_i, D_j)$ and their labels. In the classification stage, for each pair of vehicles (U_i, D_j), the SVM classifier gives a confidence $\mathrm{Conf}(U_i, D_j)$ that describes the classification result. A value for $\mathrm{Conf}(U_i, D_j)$ that is larger than 0 means that (U_i, D_j) is matched, while a $\mathrm{Conf}(U_i, D_j)$ value that is less than 0 means that (U_i, D_j) is unmatched. Thus, the possibility that (U_i, D_j) are matched is proportional to $\mathrm{Conf}(U_i, D_j)$.

5.2 Linear Programming-based Mapping

$\mathrm{Conf}(U_i, D_j)$ describes how identical a vehicle pair is. However, it is possible for one upstream vehicle to be matched to more than one downstream vehicle. Mapping resolves this issue by formulating a global optimization problem that maximizes the overall confidence of all the matched vehicle pairs between two cameras by imposing several constraints. Mathematically, let $\mathrm{x}(\mathrm{U_i}, \mathrm{D_j})$ denote whether (U_i, D_j) is identical. If so, $\mathrm{x}(\mathrm{U_i}, \mathrm{D_j}) = 1$. If not, $\mathrm{x}(\mathrm{U_i}, \mathrm{D_j}) = 0$. Then, the maximization problem can be expressed as a standard linear programming problem:

$$\max(\sum_{\mathrm{U_i}}\sum_{\mathrm{D_j}} \mathrm{Conf}(\mathrm{U_i}, \mathrm{D_j})\mathrm{x}(\mathrm{U_i}, \mathrm{D_j}) + \sum_{\mathrm{U_i}} \mathrm{Conf}(\mathrm{U_i}, \mathrm{D_\varphi})\mathrm{x}(\mathrm{U_i}, \mathrm{D_\varphi}) + \sum_{\mathrm{D_j}} \mathrm{Conf}(\mathrm{U_\varphi}, \mathrm{D_j})\mathrm{x}(\mathrm{U_\varphi}, \mathrm{D_j})) \tag{10}$$

Subject to:

$$\mathrm{Conf}(\mathrm{U_i}, \mathrm{D_\varphi}) = \delta \text{ for all } \mathrm{U_i} \tag{10.1}$$

$$\mathrm{Conf}(\mathrm{U_\varphi}, \mathrm{D_j}) = \delta \text{ for all } \mathrm{D_j} \tag{10.2}$$

$$\textstyle\sum_{\mathrm{U_i}} \mathrm{x}(\mathrm{U_i}, \mathrm{D_j}) + \mathrm{x}(\mathrm{U_\varphi}, \mathrm{D_j}) = 1 \text{ for all } \mathrm{D_j} \tag{10.3}$$

$$\textstyle\sum_{D_j} \mathrm{x}(\mathrm{U_i}, \mathrm{D_j}) + \mathrm{x}(\mathrm{U_i}, \mathrm{D_\varphi}) = 1 \text{ for all } \mathrm{U_i} \tag{10.4}$$

$$\text{if } \mathrm{TM}_{\mathrm{D}_j} \notin [\mathrm{TM}_{\mathrm{U}_i} + \mathrm{TM_{min}}, \mathrm{TM}_{\mathrm{U}_i} + \mathrm{TM_{max}}], \text{then } \mathrm{x}(\mathrm{U_i}, \mathrm{D_j}) = 0 \tag{10.5}$$

$$\mathrm{x}(\mathrm{U_i}, \mathrm{D_j}) \in \{0,1\}, \mathrm{x}(\mathrm{U_i}, \mathrm{D_\varphi}) \in \{0,1\}, \mathrm{x}(\mathrm{U_\varphi}, \mathrm{D_j}) \in \{0,1\} \tag{10.6}$$

In equations 10.1 and 10.2, δ is the confidence of void object mapping, and this is set at 0 because 0 is the decision boundary of confidence of matched and unmatched vehicle pairs. Equations 10.3 and 10.4 ensure a one-to-one mapping, that is, each vehicle in Ucan only be mapped to a single vehicle (including void object) in D and vice versa. Note U_φ and D_φ are not subject to this one-to-one mapping restriction.

The time constraint is also considered in equation 10.5. The travel time for one vehicle moving from the upstream camera to the downstream camera under normal conditions is constrained. Let $\mathrm{TM_{min}}$ and $\mathrm{TM_{max}}$ denote the minimum and maximum time one vehicle needs to travel from the upstream camera to the downstream camera. $\left(\mathrm{TM}_{\mathrm{U}_i}, \mathrm{TM}_{\mathrm{D}_j}\right)$ is the timestamp denoting the time when $(\mathrm{U_i}, \mathrm{D_j})$ disappear from the upstream and downstream cameras, respectively. Thus, unreasonable travel time should be eliminated.

Charpter 6 Results

The VRI algorithm was tested using two case studies:

- Case 1:
  - A 3.7-km section of a three-lane freeway with no entrances or exits (figs. 6.1a and 6.1b) from camera W/O SIX FLAGS to camera E/O PACIFIC.
  - The upstream location was recorded from 15:41:30 to 16:11:30 on May 26th, 2014; the downstream location was recorded from 15:42:30 to 16:12:30 on the same day.
- Case 2
  - A 1.7-km section of four-lane freeway with one exit (figs. 6.1c and 6.1d) from camera N/O LINDBERGH to camera REAVIS BARRACKS.

The frame rate of the videos was 12 frames per second (FPS) and the video resolution was 360*240 pixels. The average size of each vehicle was about 40*40 pixels. Illumination and viewpoint changes were involved in both cases to test the effectiveness of the proposed VRI algorithm.

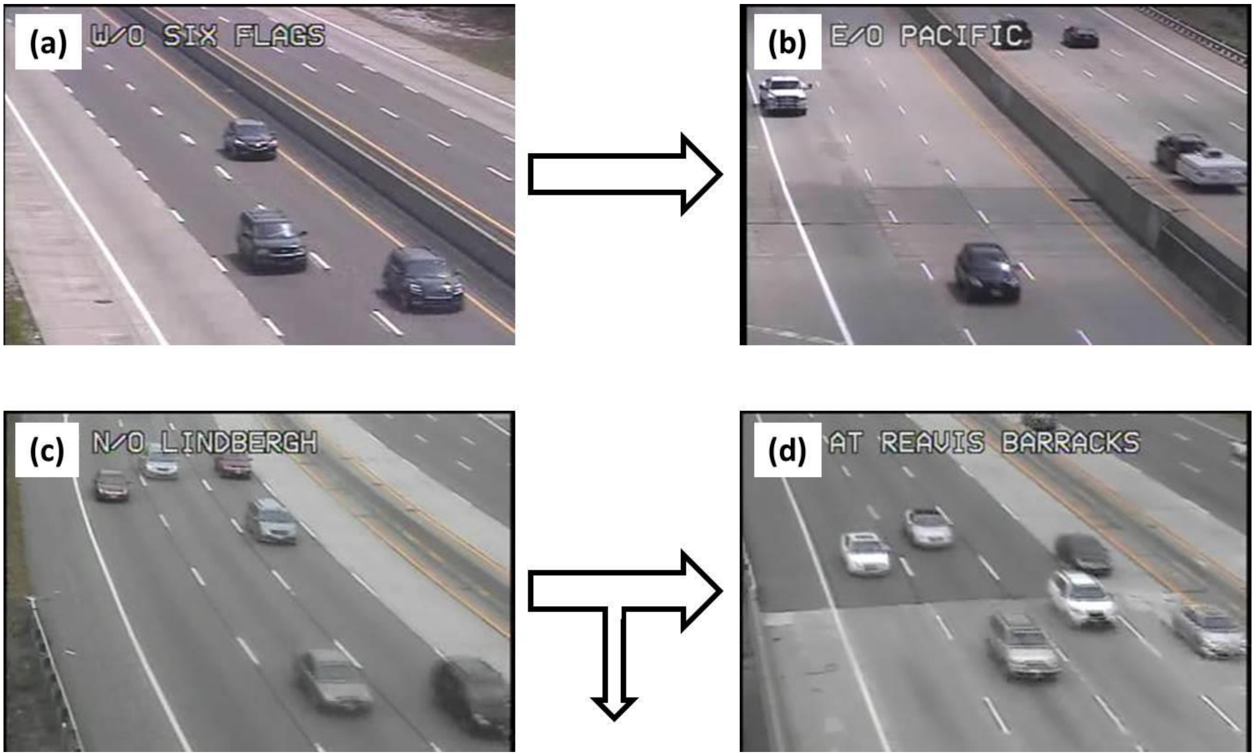

Figure 6.1 Screenshots of recorded videos: (a) upstream frame in case 1, (b) downstream frame in case 1, (c) upstream frame in case 2, and (d) downstream frame in case 2. Case 1 involves no entrances or exits while case 2 has one exit.

The ground truth was obtained by manually detecting and re-identifying vehicles in the upstream and downstream videos. In case 1, 776 vehicles were detected in the upstream video and 804 vehicles in the downstream video during the 30-min period, of which 713 pairs of vehicles were manually matched. In case 2, 961 vehicles were detected in the upstream video and 775 vehicles in the downstream video during the 20-min period, of which 750 pairs of vehicles were manually matched. Copies of these videos were also recorded for training; 683 and 692 pairs of vehicles were manually matched for cases 1 and 2 in the training stage, respectively.

6.1 Performance Metrics

Three metrics are utilized here to evaluate the performance of the VRI system:

$$\text{Precision} = \frac{TP}{TP+FP} \tag{11}$$

$$\text{Recall} = \frac{TP}{TP+FN} \tag{12}$$

$$\text{Fscore} = 2 \cdot \frac{\text{precision.recall}}{\text{precision+recall}} \tag{13}$$

where true positives (TP) represent the number of correctly matched vehicle pairs. False Positives (FP) are the number of pairs of different vehicles mistakenly regarded as matched by the algorithms, and false negatives (FN) are the number of identical vehicle pairs mistakenly regarded as different vehicles by the algorithms. Recall concerns the proportion of correctly matched vehicles among the ground truth sample, while precision focuses on the proportion of correctly matched vehicles among all vehicle pairs matched by the proposed VRI algorithm. The F-score is a comprehensive evaluation.

6.2 Quantitative Performance Evaluation

In case 1, 492 pairs of vehicles were correctly matched while 239 pairs were deemed to be FP. Thus, the precision, recall, and F-score of case 1 are 67.31%, 69.00%, and 68.14%, respectively. In case 2, 430 pairs of vehicles were correctly matched while 337 pairs were deemed to be FP. Thus, the precision, recall, and F-score of case 2 are 56.06%, 57.33%, and 56.69%, respectively. Sumalee et al. [9] achieved 54.75% re-identification precision in videos with a much higer resolution of 764*563 pixels, thus the present algorithm ourperforms their proposed probabilistic fusion method.

Note that the performance for case 2 was much lower than case 1. This might be because there is a freeway exit between the two cameras in this case, so some vehicles in the upstream video will not appear in the downstream video, introducing noise when they are re-identified. To validate the analysis, we manually removed all the vehicles that exited before reaching the downstream camera, at which point the re-identification result improved to 70.82% in precision, 68.93% in recall, and 69.86% in F-score. These results are comparable to those for case 1. Placing surveillance cameras on the exit would thus enhance the algorithm's matching performance by making it possible to re-identify vehicles using three cameras (upstream, downstream, and exit).

6.3 Comparison

To validate the effectiveness of the specific components in the proposed VRI algorithm, a series of comparisons were performed. The decrease in performance when one component is not considered indicates that this component makes a contribution to the VRI algorithm and the degree of the decrease can indicate the relative importance of that component.

Comparing the first two rows of table 6.1 with the last row of table 6.1 (the proposed VRI algorithm) clearly demonstrates that the operations of segmentation and warping applied to every vehicle image are helpful in improving the performance.

The next three rows in table 6.1 indicate the relative importance of three features (size, color, texture). For example, in case 2, the F-score decreases by 1.86%, 11.91%, and 17.82% when the size, color, and texture features, respectively, are not considered. This indicates that the texture feature is more important than the size and color features, which can be explained in several aspects. First, when vehicles are viewed from a long distance, their size difference is not obvious, thus it is not enough to distinguish one vehicle from others based on size. Second, in both case 1 and case 2, the colors of over 60% of the vehicles were either white or black, and this limited color palette makes it harder to distinguish between the many vehicles with similar coloration. Third, the texture feature is sensitive to quite subtle differences between vehicles, such as whether the lights are on and whether the sunroof is open, which makes it well suited to the identification of individual vehicles.

The last three rows in table 6.1 compare the classifier adopted in this paper with other classifiers implemented in previous work. A decision tree is used in Tawfik et al. [1], while a Bayes based classification model is utilized in Sumalee et al. and Cetin and Nichols [9, 12]. The results clearly show that SVM outperforms other classifiers in the VRI classification problem.

Table 6.1 Algorithm Comparison (√: considered –: not considered DT: Decision Tree)

Preprocessing		Feature			Classifier			Results for Case1			Results for case 2		
Segmentation	Warping	Size	Color	Texture	SVM	DT	Bayes	%Precision	%Recall	%Fscore	%Precision	%Recall	%Fscore
√	–	√	√	√	√	–	–	62.88	64.38	63.62	54.68	56.13	55.39
–	√	√	√	√	√	–	–	60.94	61.71	61.32	54.12	53.47	53.79
√	√	–	√	√	√	–	–	57.50	58.06	57.28	54.09	55.60	54.83
√	√	√	–	√	√	–	–	57.99	60.03	58.99	44.11	45.47	44.78
√	√	√	√	–	√	–	–	36.02	37.59	36.79	38.29	39.47	38.87
√	√	√	√	√	–	√	–	29.88	25.95	27.82	30.40	27.60	28.93
√	√	√	√	√	–	–	√	45.37	47.41	46.36	41.94	43.33	42.62
√	√	√	√	√	√	–	–	**67.31**	**69.00**	**68.14**	**56.06**	**57.33**	**56.69**

6.4 Travel Time Estimation

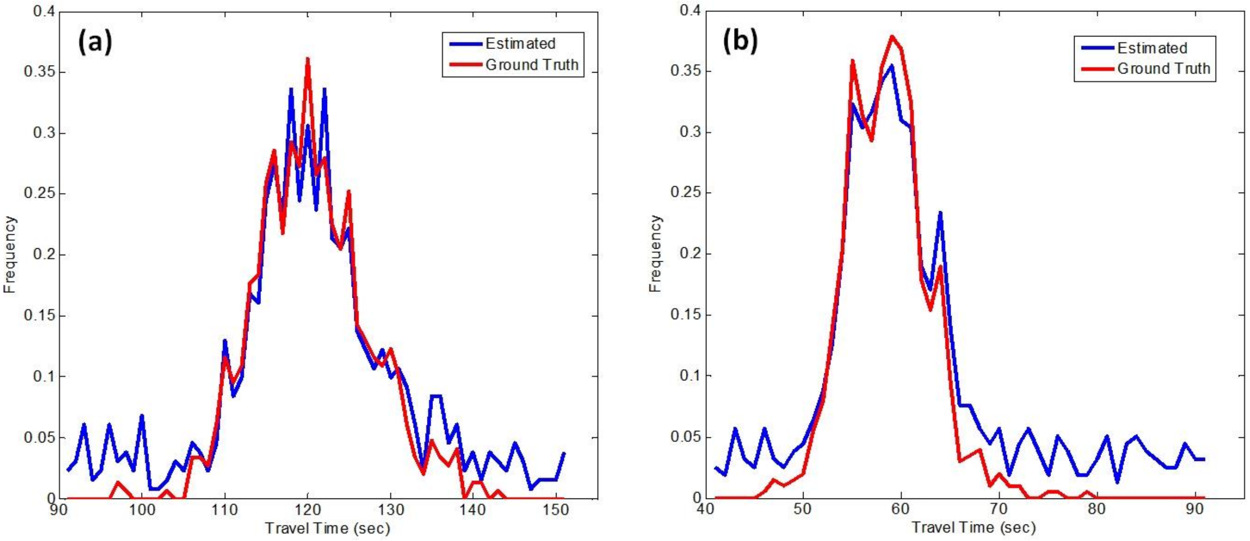

Figure 2.2 Comparison of the estimated and manually observed travel time distributions of (a) case 1 and (b) case 2

Travel time can be obtained by calculating the time differences between vehicle pairs in the upstream and downstream videos. Ground truth is obtained by manually matching vehicle pairs while the estimated travel time distribution is obtained by TP and FP. Figure 6.2a and 6.2b show the travel time distributions for case 1 and case 2, respectively. To verify the validity of the estimated travel time distributions, Root Mean Square Deviation (RMSD) is adopted as a performance metric here, which is calculated by

$$\mathrm{RMSD} = \left(\frac{1}{K}\sum_{k=1}^{K}(X_k - X_k^*)^2\right)^{\frac{1}{2}} \tag{14}$$

where K is the number of bins of the time distribution, X_k is the estimated frequency of travel time, and X_k^* is the ground truth of frequency. The RMSD for the estimated travel time distributions are

0.0270 and 0.0324 for cases 1 and 2, respectively. These relatively small values for RMSD indicate that the results of the proposed VRI algorithm do indeed offer a reliable way to estimate the travel time distribution between the upstream and downstream cameras.

In addition to the estimation of travel time distribution, the average travel times can also be compared. Relative Error (RE) is adopted to measure the performance here, which is defined as

$$\mathrm{RE} = \frac{|\bar{Y}-\bar{Y}^{*}|}{\bar{Y}^{*}} \tag{15}$$

where $\bar{Y}$ is the estimated average travel time, while $\bar{Y}^{*}$ is the ground truth. The RE for case 1 and 2 are $\frac{|119.575-119.891|}{119.891} = 0.26\%$ and $\frac{|60.478-60.083|}{60.083} = 0.66\%$, respectively, which also shows the accuracy of travel time estimation.

Chapter 7 Conslusion and Future Work

This paper proposes a new Vehicle Re-identification (VRI) algorithm applied to low-resolution traffic surveillance cameras on freeways with applications of travel time estimation. The contribution of this paper includes setting up a complete vehicle detection and feature extraction system when the quality of vehicle images is low. By considering VRI as a classification and mapping problem, Support Vector Machine (SVM) and linear programming are adopted to solve the vehicle matching problem. The approach was tested on two cases. One is closed without any entrance or exit, while the other has one exit. The re-identification F-score is about 68% and 57% for the two cases, respectively, which outperforms the state-of-the-art methods. The estimation of travel time distribution also shows that the proposed VRI system is reliable.

In the future work, the proposed VRI algorithm will be tested in a larger scale traffic surveillance camera network. Also, the probability that one vehicle changes its lane and the probability that one vehicle exits from the fork can be expected as context information to improve VRI accuracy.

References

1. Tawfik, A. Y., B. Abdulhai, A. Peng, and S. M. Tabib. Using Decision Trees to Improve the Accuracy of Vehicle Signature Re-identification. In *Transportation Research Record: Journal of the Transportation Research Board, No. 1886*, Transportation Research Board of the National Academies, Washington, D.C., 2004, pp. 24–33.

2. Kwon, T. M., and A. Parsekar. Blind Deconvolution Processing of Loop Inductance Signals for Vehicle Re-identification. In *Proceedings of the 85th Annual Meeting of the Transportation Research Board*, Transportation Research Board of the National Academies, Washington, D.C., 2006, pp. 06-2581.

3. Barceló, J., L. Montero, L. Marqués, and C. Carmona. Travel Time Forecasting and Dynamic Origin–Destination Estimation for Freeways Based on Bluetooth Traffic Monitoring. In *Transportation Research Record: Journal of the Transportation Research Board, No. 2175*, Transportation Research Board of the National Academies, Washington, D.C., 2010, pp. 19–27.

4. Sanchez, R.O., C. Flores, and R. Horowitz. Vehicle Re-identification Using Wireless Magnetic Sensors: Algorithm Revision, Modificationsand Performance Analysis. In *IEEE International Conference on Vehicular Electronics and Safety*, 2011, pp.226-231.

5. Ozbay, S., and E. Ercelebi. Automatic Vehicle Identification by Plate Recognition. In *World Academy of Science, Engineering and Technology*, 2005, pp. 222-225.

6. Wang, Y., S. Velipasalar, and M. C. Gursoy. Wide-area Multi-object Tracking with Non-overlapping Camera Views. In *International Conference on Multimedia and Expo*, 2011.

7. Wenbin, J., C. Xiao, H. Jin, S. Zhu, and Z. Lu. Vehicle Tracking with Non-overlapping Views for Multi-camera Surveillance System. In *High Performance Computing and Communications & 2013 IEEE International Conference on Embedded and Ubiquitous Computing*, 2013.

8. Hou, T., S. Wang, and H. Qin. Vehicle Matching and Recognition Under Large Variations of Pose and Illumination. In *Computer Vision and Pattern Recognition Workshops*, 2009, pp. 290–295.

9. Sumalee, A., et al. Probabilistic Fusion of Vehicle Features for Re-identification and Travel Time Estimation Using Video Image Data. In *Transportation Research Record: Journal of*

the Transportation Research Board, No. 2308, Transportation Research Board of the National Academies, Washington, D.C., 2012, pp. 73-82.

10. Sun, C. C., G. S. Arr, R. P. Ramachandran, and S. G. Ritchie, Vehicle Re-identification Using Multidetector Fusion. In *IEEE Transactions on Intelligent Transportation Systems,* 2004, pp. 155–164.

11. Kamijo, S., T, Kawahara, and M, Sakauchi. Vehicle Sequence Image Matching for Travel Time Measurement between Intersections. In *IEEE International Conference on Systems Systems,* Man and Cybernetics, 2005, pp. 1359–1364.

12. Cetin, M., and A. P. Nichols. Improving the Accuracy of Vehicle Re-identification Algorithms by Solving the Assignment Problem. In *Transportation Research Record: Journal of the Transportation Research Board, No. 2129,*Transportation Research Board of the National Academies, Washington, D.C., 2009, pp. 1–8.

13. Dalal, N., and B. Triggs. Histograms of Oriented Gradients for Human Detection. In *Computer Vision and Pattern Recognition,* 2005.

14. Guo, Z., L. Zhang, and D. Zhang. A Completed Modeling of Local Binary Pattern Operator for Texture Classification. In *IEEE Transactions on Image Process*, 2010, pp. 1657–1663.

15. Lowe, D. G. Distinctive Image Features From Scale-invariant Keypoints. In *International Journal of Computer Vision*, 2004, pp. 91-110.

16. Yin, Z., and R. Collins. Moving Object Localization in Thermal Imagery by Forward-backward Motion History Images. In *Augmented Vision Perception in Infrared*, 2009, pp.271-291.

17. Torralba, A., K. P. Murphy, and W. T. Freeman. Sharing Features: Efficient Boosting Procedures for Multiclass Object Detection. In *Computer Vision and Pattern Recognition,* 2004.

18. Stauffer, C., and W. Eric L. Grimson. Adaptive Background Mixture Models for Real-time Tracking. In *Computer Vision and Pattern Recognition*, 1999.

19. Felzenszwalb, P. F., and D. P. Huttenlocher. Efficient Graph-based Image Segmentation. In *International Journal of Computer Vision*, 2004, pp. 167-181.

20. Kanhere, N. K., S. T. Birchfield, W. A. Sarasua, and T. C. Whitney. Real-Time Detection and Tracking of Vehicle Base Fronts for Measuring Traffic Counts and Speeds on Highways.

In *Transportation Research Record: Journal of the Transportation Research Board, No. 1993,*Transportation Research Board of the National Academies, Washington, D.C., 2007, pp. 155–164.

21. Finlayson, G., and R. Xu. Non-iterative Comprehensive Normalisation. In *Conference on Colour in Graphics, Imaging, and Vision*, 2002.

22. Mukkamala, S., and A. H. Sung. Feature Selection for Intrusion Detection with Neural Networks and Support Vector Machines. In *Transportation Research Record: Journal of the Transportation Research Board, No. 1822,* Transportation Research Board of the National Academies, Washington, D.C., 2003, pp. 33-39.

23. Zhang, N., Y. Zhang, and H. Lu. Seasonal Autoregressive Integrated Moving Average and Support Vector Machine Models. In *Transportation Research Record: Journal of the Transportation Research Board*, No. 2215, Transportation Research Board of the National Academies, Washington, D.C., 2011, pp. 85-92.

24. Scholkopf, B., and A. J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*, MIT press, 2001.