Vehicle Reidentification and Travel Time Measurement on Congested Freeways

Benjamin Coifman, PhD

Assistant Professor, Civil and Environmental Engineering and Geodetic Science
Assistant Professor, Electrical Engineering

Ohio State University
470 Hitchcock Hall
2070 Neil Ave
Columbus, OH 43210-1275

cofman.1@osu.edu
614 292-4282
Abstract

This paper presents a vehicle reidentification algorithm for consecutive detector stations on a freeway, whereby a vehicle measurement made at a downstream detector station is matched with the vehicle’s corresponding measurement at an upstream station. The method is illustrated using measured vehicle lengths from paired loop detector speed traps. Where speed traps are quite common, often placed at half mile spacings or less on urban freeways. In conventional operation, these detectors only monitor traffic conditions over the loops, leaving most of the freeway unmonitored. By taking the difference in known arrival times for a matched vehicle, it is possible to measure true link travel time and thus, quantify conditions between detector stations. This approach is significant because no one has attempted to use the existing detector infrastructure to match vehicle measurements between detector stations. The results of this work suggest that it is possible to extract a sufficient number of vehicle matches for traffic surveillance applications, while accepting few, if any, false positives. Thus, with the new algorithm, it will be possible to evaluate applications of travel time data without deploying new detector hardware.

**Keywords:** traffic surveillance, loop detectors, travel time measurement, vehicle reidentification
Introduction

This paper presents a vehicle reidentification algorithm for consecutive detector stations on a freeway, whereby a vehicle measurement made at a downstream detector station is matched with the vehicle's corresponding measurement at an upstream station. The algorithm should improve freeway surveillance by measuring the actual vehicle travel times; these are simply the differences in the times that each (matched) vehicle arrives to the upstream and downstream stations. Thus, it will be possible to quantify conditions between widely spaced detector stations rather than assuming that the local conditions measured at a single station are representative of an extended link between stations.

The method is developed using effective vehicle length\(^1\) measured at dual loop speed traps. These detectors are quite common, often placed at half mile spacings or less on urban freeways. The approach is significant because no previous work uses the existing detector infrastructure to match vehicle measurements between detector stations. The work is also transferable to other detector technologies capable of extracting a reproducible vehicle measurement, or "vehicle signature", such as video image processing.

Because the algorithm was developed with conventional loop detectors in mind, it uses the (effective) length measurements to distinguish vehicles. Notably, a length measurement may be accurate to 2 feet or worse due to resolution limitations, making difficult the task of matching vehicle measurements between the upstream and downstream detector stations. However, if the difference between two measurements exceed this measurement resolution, then the pair of measurements probably did not come from the same vehicle. After applying this resolution test to each pair of upstream and downstream measurements (for some specified group of vehicles), the remaining pair-wise comparisons that can not be eliminated are considered possible matches. For example, the upstream and downstream length measurements from the same vehicle should pass the resolution test and the pair will be labeled a possible match. Frequently however, one vehicle's measurement downstream will be a possible match to a different vehicle's measurement upstream because this pair of measurements likewise passes the resolution test. Clearly, these possible, but incorrect, matches are false positives. Toward eliminating the false positives, the algorithm uses a simple trick: it matches platoons whenever vehicles pass both detectors in the same relative order.

\(^1\) The effective vehicle length is the length as "seen" by the detectors; i.e., the sum of the physical vehicle length and the length of the detection zone.
The sequence of measured lengths in a platoon provides more information than do the individual measurements.

Because the measurement resolution at the speed traps improves as velocity decreases, the current implementation is limited to congested conditions, i.e., when the local velocity is less than 45 mph at one or both detector stations. This restriction is not as limiting as it may seem since free flow traffic is characterized by relatively constant velocities and thus, relatively constant travel times. To this end, the algorithm can be used alone as a research tool or it can be used in conjunction with extensions to free flow traffic, as discussed in [1], for active control.

**Overview**

To motivate the work, the following section presents several applications of vehicle reidentification and travel time data. The next section reviews preceding vehicle reidentification work. Then, the loop based vehicle reidentification system is presented in detail. Finally, the paper closes with a brief discussion and conclusions.

**Motivation and Applications**

Travel time data could improve existing traffic management applications such as incident detection, control at ramp meters, and traveler information. The travel time data could also serve as input to emerging technologies such as dynamic traffic assignment (DTA). More importantly, the data could be used to quantify the benefits from emerging technologies using real traffic data, off-line, before making significant infrastructure investments. Such analysis will allow for quantifying the necessary level of accuracy for a given application. As accuracy increases, the marginal costs for further improvements will likely increase. Thus, an operating agency can deploy the least expensive detection system (loop based or otherwise) that meets the specified requirements.

Finally, there are applications which might benefit from the vehicle reidentification or travel time data, although on their own, probably do not justify the deployment of a vehicle reidentification system. For example, the travel time data could be useful for planning applications and the reidentification algorithm could be used to study individual driver dynamics over time and space. The remainder of this section will examine three applications in detail: incident detection, DTA, and model validation and calibration.
Incident detection

A recent report from Caltrans [2] stated that, "Incidents are, by definition, perturbations in the normal operating characteristics of a transportation system, chief of which is travel time." The potential benefits of incident detection have been known for years [3-7]. Faster response to an incident can reduce the number of drivers affected and reduce the average delay for those who are affected. By reducing total delay, other costs associated with the incident, such as wasted fuel and increased emissions, will also decrease.

Countless automated incident detection strategies have been proposed, but most of these systems suffer from high false alarm rates and/or long detection times [8]. Lin and Daganzo [9] have demonstrated a reliable incident detection system using speed traps. The system uses two widely spaced detector stations to detect two "signals" that propagate through the traffic stream. The signals, a backward moving shock wave and a forward moving drop in flow, are indicative of an incident between the stations. As noted by the researchers, "Detection of an incident can happen only when both signals have been received...." Although the drop in flow travels at the prevailing traffic speed, this earlier work estimated the shock wave speed to be on the order of 8 mph.

Fortunately, the drop in flow reflects the fact that vehicles are being delayed behind the incident. All vehicles that arrive at the downstream station after the drop will experience increased travel times over the link. Thus, an incident detection system based on travel time may not have to wait for the slow moving shock wave to reach the upstream station before detecting the incident. By incorporating link data and point data, it should be possible to develop a fast and robust incident detection algorithm.

Dynamic Trip Assignment

Many researchers are investigating DTA as a means to reduce traveler delay. As proposed, a DTA system would observe current [10-17] and historical traffic conditions [10-19], estimate travel times over the network and then route vehicles with the goal of reducing traveler delay.

Typically, the travel time forecasts are based on traditional traffic parameters (such as flow, velocity, and occupancy) measured at discrete point detectors [10-17]. Usually, the point measurements are averaged over fixed time periods (20 seconds-15 minutes) to smooth out transients and then generalized to a link of significant length (0.5-5 miles long). Unfortunately, there is not a one-to-one relationship between travel time over an extended link and traffic parameters measured at a discrete point within that link. The DTA literature does not appear to consider the option of measuring travel time directly; but the use of direct travel time measurements
should improve the performance of a travel time forecasting algorithm both through real time data, and by providing a set of historical data.

Although the promoters of DTA systems forecast significant benefits, the systems have only been tested in simulation or in very limited field studies [20]. The proposed travel time measurement system could be used for much-needed evaluation under real-world conditions.

**Model validation and calibration**

Model validation and calibration is an important task for the traditional four step planning process as well as the on-going Travel Model Improvement Program which seeks to replace this process with microsimulation models. For example, the TRANSIMS designers at Los Alamos National Labs note that, "The most important result of a transportation microsimulation in [the planning] context should be the delays..." [21]. It will be important to verify and calibrate these models to real networks, a task that is well suited to the travel time measurement system.

**Preceding Work**

This section discusses preceding research related to vehicle reidentification or travel time measurement systems. First, complementary detector technologies are discussed, and then competing vehicle reidentification systems are presented.

**Complementary technologies**

Although this paper focuses on measured vehicle lengths from speed traps, the reidentification algorithm could be applied to other signature based detector systems. There are four emerging detector systems under Caltrans sponsorship that promise to yield more robust vehicle signatures while being compatible with the reidentification algorithm:

1. Magnetic Vehicle Signatures from Loop Detectors, [22].


3. Vehicle Dimensions and Velocity From Overhead Video Detectors, [24].

4. Visual Vehicle Signatures from Wayside Cameras, [25].

For example, item 2 above is designed to measure vehicle length with an error of 1 inch at free flow traffic speeds (versus 24 inches with the speed traps).
Competing technologies

Several systems have been proposed for measuring travel time directly using vehicle signatures [22, 26-29]. These emerging technologies use specialized hardware to extract vehicle signatures that are more descriptive than effective length. As a result, the hardware must be deployed before quantifying the benefits. In most cases, the systems have only been installed on small test sites. Other systems use automatic vehicle identification (AVI), e.g., machine readable "license plates", [30-36] that make vehicle reidentification trivial, but the systems may compromise personal privacy. Furthermore, the AVI systems do not measure local conditions at the detectors, this omission can impact surveillance and control.

Other surveillance systems have been proposed for estimating travel time from aggregate traffic parameters [37-38]. Although these systems appear promising for free flow and lightly congested conditions, they currently do not perform well under heavy congestion.

Another approach for measuring travel time is to match vehicles simply based on the cumulative arrivals at successive detector stations [39-40], i.e., the n-th vehicle at one station is matched to the n-th vehicle at the next station. To counter detector drift between stations, these systems use aggregate measurements to recalibrate during free flow conditions. Unfortunately, congestion can last several hours, leading to significant measurement drift between recalibrations.

Vehicle Reidentification

This section presents the vehicle reidentification algorithm in detail. The work builds off of simpler algorithms presented in [41]. An ongoing example will be used to illustrate the various steps of the algorithm throughout the section. This example uses data from two speed traps on Interstate 80 in Berkeley, California, as shown in Figure 1. Concurrent video was collected at each station to serve as ground truth and approximately 1,500 vehicles were observed at each station.

The section begins by defining the simple vehicle signatures that serve as input parameters, then proceeds through the reidentification algorithm, and finally the algorithm performance is compared against the ground truth data.

Simple Vehicle Signatures

As a vehicle passes over a speed trap, e.g., Figure 2A, the controller normally records four transitions, as shown in Figure 2B. After accounting for any unmatched transitions, the following
parameters are calculated for each vehicle: speed trap traversal time via the rising edges (TT_r), speed trap traversal time via the falling edges (TT_f), total time the upstream detector is on (OT_u) and total time the downstream detector is on (OT_d), where:

\[
TT_r = t_{\text{RISE, down}} - t_{\text{RISE, up}}
\]

\[
TT_f = t_{\text{FALL, down}} - t_{\text{FALL, up}}
\]

\[
OT_u = t_{\text{FALL, up}} - t_{\text{RISE, up}}
\]

\[
OT_d = t_{\text{FALL, down}} - t_{\text{RISE, down}}
\]

Vehicle velocity is simply the loop separation divided by the traversal time,

velocity from rising edge:
\[
V_r = \frac{20}{TT_r} \text{ [ft/sec]}
\]

velocity from falling edge:
\[
V_f = \frac{20}{TT_f} \text{ [ft/sec]}
\]

where 20 (ft) represents the loop separation, i.e., the spacing between corresponding points on the two loops. Effective vehicle length is the measured velocity multiplied by the time the detector was on, but because there are two measurements for each of these parameters, two length measurements are made:

length measurement #1:
\[
L_1 = V_r \cdot OT_u \text{ [ft]}
\]

length measurement #2:
\[
L_2 = V_f \cdot OT_d \text{ [ft]}
\]

and the average of the two measurements is recorded as the effective vehicle length. The controller samples the loops at 60 Hz, so at best, each parameter in Equation 1 is accurate to ±1/30 seconds. To capture this resolution constraint, the measurement uncertainty is defined as the range between the two length measurements after accounting for the parameter measurement accuracy. Misdetections will impact these calculations, but these errors are addressed in later steps of the algorithm. To ensure the best measurements possible, any hardware problems, such as cross talk between detectors, are identified using [42] and corrected.
**Possible Matches**

For a given downstream measurement, the algorithm attempts to find the upstream measurement that corresponds to the same vehicle. Assuming that the vehicle did not change lanes between detector stations, it is possible to select a reasonable set of upstream measurements from some time earlier that probably includes the matching measurement. Unfortunately, it is not possible to find the upstream match directly for several reasons: a vehicle's measured length is not unique, it may be subject to measurement errors, and as noted above it is subject to resolution constraints. To illustrate this problem, Figure 3A shows a sequence of upstream length measurements with dark rectangles. The measurements are plotted in the order that the vehicles were observed and the height of each rectangle reflects the measurement uncertainty. Superimposing the downstream measurement, subject to its measurement uncertainty, yields the lighter rectangle. Comparing this measurement to all of the upstream measurements, one can not identify a unique match with a particular upstream vehicle. But it is possible to eliminate many unlikely matches via this resolution test, because, even allowing for the measurement uncertainty, most upstream measurements do not intersect the downstream measurement. All upstream vehicles that can not be eliminated are considered possible matches, and these results are indicated with open circles in Figure 3B.

**Sequences, Lane Changes and Matches**

Although it is not possible to identify a unique match for an individual vehicle, a sequence of measured lengths rapidly becomes distinct and the sequence can potentially be reidentified at successive detectors. The algorithm looks for short sequences of measured vehicle lengths that exhibit a strong correlation between two stations. Lane changes and measurement errors disrupt the sequences, so the algorithm is specifically designed to match vehicles between these disruptions.

Vehicles are assigned successive arrival numbers as they pass each detector station and these numbers are assigned independently at each station. Next, a *set of reasonable upstream matches* is identified for each downstream vehicle; where this set is the last \( n \) successive upstream vehicles in the same lane ending with the last vehicle to pass the upstream speed trap before the downstream observation. The constant, \( n \), should be set large enough to ensure that the true match will always fall within in the *set of reasonable upstream matches*, while being small enough to allow the computer to process the data. For this paper, \( n \) was set arbitrarily to 100 vehicles; then, after running the algorithm, it was verified that the true matches were always within the *set of*
reasonable upstream matches. In practice, a conservative value of n could be set from estimated jam density and the distance between stations.

Next, the algorithm applies the resolution test to each downstream vehicle and the corresponding set of reasonable upstream matches. The outcomes are stored in the vehicle match matrix, e.g., Figure 4 shows the results for 100 downstream vehicles in the sample data. The dots indicate possible matches and each row of the vehicle match matrix represents the outcome from one resolution test. Note that the columns are indexed by the difference between upstream and downstream vehicle numbers.

Many false positives are clearly evident in Figure 4 since each vehicle can only have, at most, one true match, yet most rows have more than one possible match for the given downstream vehicle. Figure 5 shows the cumulative distribution of the percentage of possible matches for all 1,500 vehicles. The frequency of false positives is less than 50 percent for seven out of 10 vehicles in this example.

Assuming that any two successive length measurements at a detector station are independent of each other, the false positives are manifest as random noise in the vehicle match matrix. If a false positive occurs with probability less than 0.5, a false positive should usually be preceded (moving up one row) by an unlikely element. On the other hand, if vehicles maintained their order between the two stations, a true match should usually be preceded by a possible match. Relaxing the order constraint somewhat, the work of John Windover on driver memory [43] has shown that long sequences of drivers often maintain their headway, and thus, their order for extended distances. So, if vehicles usually maintain their order between stations, the true (but unknown) matches should manifest themselves as sequences of possible matches in the same column. In other words, false positives will typically form short sequences while the true matches will usually form longer sequences in the vehicle match matrix. To exploit this property, the algorithm looks for sequences of possible matches in the vehicle match matrix and tallies how many sequential vehicles matched at both stations. Using the data from Figure 4, all sequences longer than one possible match are shown in Figure 6. Because the algorithm looks for sequences, even the vehicles with a high frequency of false positives in Figure 5 are informative since the unlikely elements (i.e., the non-positive results) can break false sequences.

Using concurrent video to visually match every vehicle that remained in the lane between the two stations for all 1,500 vehicles in the example, the solid line in Figure 7 shows the distribution of sequence lengths as measured by the algorithm for the true matches and the dashed line shows the distribution for all of the false positives. The true sequences are, on average, over seven vehicles
long, while the false sequences are, on average, about two vehicles long. The few long sequences of false positives are due to common vehicles and usually occur when there is a longer sequence of true matches.

Next the algorithm allows for a small set of lane changes and/or misdetections between sequences. Figures 8A-C show the three lane change maneuvers searched for by the algorithm:

(A) one vehicle exits the lane between stations or a vehicle is not detected at the downstream station,

(B) one vehicle enters the lane between stations or a vehicle is not detected at the upstream station,

(C) one vehicle enters and one vehicle leaves the lane between stations or there is a measurement error at one station.

For each sequence of vehicles, the algorithm checks the first element to see if it can be linked to an earlier sequence (i.e., a sequence starting with a lower vehicle number) via a lane change maneuver. The procedure is demonstrated using the sequence starting with element (m,n) in Figure 8D, the algorithm checks to see if there are any earlier sequences passing through one of the three shaded elements, where each element corresponds to one of the lane change maneuvers shown in Figures 8A-C. If so, the algorithm produces a modified sequence by adding the earlier possible matches to the current sequence and then applies a length penalty of one vehicle for the lane change. Otherwise, the current sequence is stored as a modified sequence without a lane change maneuver. Note that the algorithm only joins pairs of sequences via lane change maneuvers, so one sequence (or portions thereof) may be part of many modified sequences, but a modified sequence will never include more than two sequences. To illustrate this process, Figure 8E shows a case where there are two possible lane changes for the sequence starting at (m,n). The algorithm selects the entry maneuver, as shown in Figure 8F, since this yields the longest modified sequence. Observe that the union only includes a portion of the earlier sequence.

Finally, the algorithm identifies the upstream match for each downstream vehicle, or each row in the vehicle match matrix. This upstream match corresponds to the column with the longest modified sequence passing through the given row. The resulting matches from Figure 6 are shown in Figure 9. Note how most matches fall near the same column in this figure. Extending to the entire sample population, the algorithm found a match for about 90 percent of the vehicles. The corresponding travel times for these matches are shown in Figure 10. Clearly, some of the
matches are due to false positives simply because the large discontinuities between successive travel time measurements are not feasible.

**Cleanup**

To eliminate additional false positives, the algorithm follows three steps to "cleanup" the matches. These steps, based on [44], are as follows:

**Step 1, the Upstream Vehicle**

By selecting the longest *modified sequence* passing through a given row, the algorithm has ensured that each downstream vehicle is matched to, at most, one upstream vehicle. But an upstream vehicle may be matched to multiple downstream vehicles. The possible matches for an upstream vehicle fall on a diagonal at +45 degrees in the chosen coordinate system, as shown in Figure 11. It may take several minutes to observe all of the possible matches for a given upstream vehicle. Rather than waiting to observe the outcomes from all of the tests, the algorithm uses as much information as possible immediately after identifying a match for a downstream vehicle. To illustrate with match (m,n) in Figure 11, the match is discarded if its sequence length is less than that of an earlier match for the same upstream vehicle. The comparison does not consider matches from later downstream vehicles since the algorithm runs in real time. So the upstream vehicle may still be matched to multiple downstream vehicles, but the later matches (with respect to downstream vehicle number) are stronger than any earlier ones.

**Step 2, Maximum Link Velocity**

Next, recall the fact that, for a given downstream vehicle, the most recent upstream vehicle bounds the *set of reasonable upstream matches* on the right. Under normal traffic conditions, several vehicles can be eliminated from the right hand side of this set simply because they would have to travel at excessive speeds to be a true match. By waiting until this step to exclude these vehicles, some false positives corresponding to common vehicle lengths will fall into this infeasible range. For the current implementation, the algorithm discards any match that would require a link velocity in excess of 85 mph.

**Step 3, Platoon Information**

Return to Figure 9, in this example most possible matches fall near column 80, but a small number of matches fall far away from this column. Maintaining the assumption that lane changes are relatively infrequent, a sequence of true but unknown matches in one column should be preceded
by other sequences of true matches in nearby columns. Converting this heuristic to a constraint, first the algorithm identifies platoons of matches, where one platoon consists of all consecutive downstream vehicles whose matches have the same upstream offset. This calculation is necessary because earlier steps in the cleanup will likely disrupt the sequences. Then, the algorithm compares the upstream offset of a given platoon against that of the preceding eight platoons. If at least three of these comparisons are within ±5 and the current platoon is more than one vehicle long, the matches in the platoon are retained as final matches. In any event, the platoon is kept for later comparisons at this step.

**Results**

Applying the cleanup steps to the on-going example, approximately 25 percent of the matches were eliminated. Table 1 shows how many matches were retained after each step while the X’s in Figure 12A indicate the travel times from the final matches. Superimposing the travel times from the ground truth matches, we see the algorithm performed quite well; the algorithm follows the increasing and decreasing travel time as disturbances pass through the link. Local velocity at the detector stations ranged between 0 and 40 mph for this example and the algorithm matched approximately 65 percent of the vehicles that passed the upstream site. One false platoon of 12 vehicles and four false matches on the ends of true platoons remain, yielding an error rate of 1.6 percent for the example. It is important to note, however, that the algorithm recovered after making these errors. Finally, the time between successive matches is typically on the order of a few seconds for the example, as shown in Figure 12B, with 1.3 minutes being the longest period without a reidentification.

**Discussion**

The preceding section presented detailed results for just over one hour of data from one pair of detector stations, 1/3 of a mile apart. Previous work using related algorithms have examined other locations with up to two miles between detector stations (e.g., [41]). One of the most taxing challenges of this analysis is generating the ground truth data. It is left to future research to determine the optimal parameters for the algorithm, as well as the maximum detector spacing before the algorithm breaks down. To this end, work is underway to develop the Berkeley Highway Laboratory (BHL), which includes eight dual loop speed trap detector stations as well as machine vision vehicle tracking tools to ease the ground truth data collection [45]. The first
application of the BHL will be using the algorithm presented above to measure travel time in real time.\(^2\)

Before closing, it is worth discussing the quantity of matches needed for surveillance, i.e., what is a sufficient percentage of vehicles matched? The answer depends on the application. For example, Origin/Destination studies would likely require near perfect performance from non-AVI vehicle reidentification systems. Travel time measurement is not as demanding, Van Aerde, et al, [46] estimated that matching 20 percent of the population is sufficient for such measurements and Holdener and Turner, [47] suggest that the percentage may even be smaller.

**Conclusions**

This paper has presented a new algorithm to match a vehicle's length measurement at a downstream detector station with the vehicle's corresponding measurement at an upstream station. The algorithm rules out unlikely matches, looks for sequences of possible matches between measurements at the two stations and then eliminates unlikely sequences.

The beauty of the approach is in its simplicity. Matching vehicles between detector stations is a difficult task and some of the best minds have tried to tackle the problem with varying degrees of success. Preceding work emphasized computationally intensive strategies and/or hardware intensive strategies. By creating the solution space of possible matches, this research has enabled vehicle reidentification using existing detector hardware and inexpensive computers.

The contribution to the field of traffic surveillance should prove to be significant since the vehicle reidentification algorithm will allow the study of travel time applications without deploying new hardware and thereby enable cost-benefit analysis before investing in a new detection system. If travel time measurement proves to be beneficial, the algorithm could be deployed using speed traps\(^3\), or it could be transferred to emerging detector technologies that have better measurement resolution. The methodology should prove beneficial for research purposes as well; yielding better insight into vehicle dynamics between widely spaced detector stations without the host of assumptions necessary with simulation.

\(^2\) The real time system can be viewed at: http://www.its.berkeley.edu/projects/freewaydata

\(^3\) It is worth noting that the additional hardware cost per station to deploy the algorithm across all lanes in the BHL is less than one percent of the cost of installing a detector station for conventional traffic surveillance.
Acknowledgments

This work was performed as part of the California PATH (Partners for Advanced Highways and Transit) Program of the University of California, in cooperation with the State of California Business, Transportation and Housing Agency, Department of Transportation; and the United States Department of Transportation, Federal Highway Administration.

The Contents of this report reflect the views of the authors who are responsible for the facts and accuracy of the data presented herein. The contents do not necessarily reflect the official views or policies of the State of California. This report does not constitute a standard, specification or regulation.

References


[31] Balke, K., Ullman, G., McCasland, W., Mountain, C., Dudek, C., *Benefits of Real-Time Travel Information in Houston, Texas*, Southwest Region University Transportation Center, Texas Transportation Institute, College Station, TX, 1995.


Figure 1. The segment of Interstate-80 in Berkeley, California used for this study.
Figure 2. One vehicle passing over a speed trap, (A) detection zones and the vehicle trajectory as shown in the time space plane. The height of the vehicle’s trajectory reflects the non-zero vehicle length. (B) The associated detector transitions, including the upstream and downstream, rising and falling edges at the indicated times.
Figure 3. (A) "Resolution test" between one downstream vehicle and a set of upstream vehicles. If the upstream measurement range does not intersect the downstream measurement range, the upstream vehicle can be dismissed as being an unlikely match for the downstream vehicle. (B) Everything that can not be eliminated are considered a possible matches, as indicated with open circles. Note that each outcome is shown directly below the given comparison from part (A).
Figure 4. Consider a set of 100 downstream vehicles, applying the resolution test to each of these vehicles produces a set of possible matches, or one row in this matrix. Note that the columns are indexed by the upstream offset, so, if there were no lane changes in this coordinate system, all of the true matches would fall into a single column.

\[ \text{boundary line indicating the last feasible match for each downstream vehicle} \]

\[ \text{possible match} \]
Figure 5, The cumulative distribution (CDF) of the percentage of possible matches for all 1495 downstream vehicles.
Figure 6. All sequences of two or more possible matches for the on-going example.
Coifman, B.

Figure 7. Using concurrent video to calculate the true matches, the solid line shows the distribution of sequence lengths measured by the algorithm for these matches. While the dashed line shows the distribution of sequence lengths for all other possible matches, i.e., the false positives.
Figure 8. A simple example illustrating the possible lane change maneuvers recognized by the Basic Algorithm: (A) One vehicle exits the lane between stations, (B) One vehicle enters the lane between stations, (C) One vehicle enters and one vehicle exits the lane between stations, (D) The search region for the sequence starting at element \((m,n)\). (E) Three sample sequences, one in each column \(n-1\) to \(n+1\). (F) In this case, the sequence starting at \((m,n)\) is joined via an entrance (part (B)) to a portion of an earlier sequence.

\(\odot\) = possible match

\[\text{element to check for an earlier sequence}\]
Figure 9, The resulting matches for the on-going example after allowing for lane changes. Each match corresponds to the longest modified sequence for the given downstream vehicle. Note how most matches fall near column 80 with small column shifts due to lane change maneuvers.
Figure 10. After applying the algorithm to 1495 vehicles, 1345 vehicles were matched by the algorithm. This figure shows the resulting travel times for the matches. Most of the travel times seem plausible; but clearly, there are a significant number of erroneous matches, manifest as random noise.

Coifman, B.
The longest modified sequence passing through the given row was identified as the match for the downstream vehicle. Each of these matches is then compared to any earlier matches for the upstream vehicle (the diagonal line). The match is discarded if its sequence length is less than that of an earlier match for the same upstream vehicle. The comparison does not consider matches from later downstream vehicles since the algorithm runs in real time.

Coifman, B.
Figure 12. (A) Compare the travel times for the 1008 final matches against those from the ground truth matches. (B) The time between successive final matches.
Table 1, Number of matches after each cleanup step for the on-going example.

<table>
<thead>
<tr>
<th>Step</th>
<th>Number of Matches</th>
</tr>
</thead>
<tbody>
<tr>
<td>pre-cleanup</td>
<td>1345</td>
</tr>
<tr>
<td>step 1</td>
<td>1212</td>
</tr>
<tr>
<td>step 2</td>
<td>1202</td>
</tr>
<tr>
<td>step 3, final matches</td>
<td>1008</td>
</tr>
</tbody>
</table>