Estimating Velocity Fields on a Freeway from Low Resolution Video

Young Cho
Department of Statistics
University of California, Berkeley
Berkeley, CA 94720-3860
Email: young@stat.berkeley.edu

John Rice
Department of Statistics
University of California, Berkeley
Berkeley, CA 94720-3860
Email: rice@stat.berkeley.edu

***Abstract*— We present an algorithm to estimate velocity fields from low resolution video recordings. The algorithm does not attempt to identify and track individual vehicles, nor does it attempt to estimate derivatives of the field of pixel intensities. Rather, we compress a frame by obtaining an intensity profile in each lane along the direction of traffic flow. The speed estimate is then computed by searching for a best matching profile in a frame at a later time. Because the algorithm does not need high quality images, it is directly applicable to a compressed format digital video stream, such as mpeg, from conventional traffic video cameras. We illustrate the procedure using a 15 minute long VHS recording to obtain speed estimates on a one mile stretch of highway I-80 in Berkeley, California.**

I. Introduction

Traffic cameras offer the potential to complement or substitute for loop detectors. Because they can provide finer spatial and temporal resolution, they have many advantages over loop detectors. In principle, video from cameras can also be used to detect lane-changing, accidents, and queuing patterns and to extract macroscopic traffic parameters, such as flow, speed, and density. Also, cameras are becoming less expensive to purchase and maintain.

However, in order to use images from cameras to study traffic, a large amount of video must be processed and an efficient and practical system to extract traffic parameters is thus essential. After being digitized, an hour video can be up to a several gigabytes.

The objective of this paper is to present a simple algorithm to estimate a velocity field, localized in space and time, from video data covering a wide area with limited spatial resolution. The localization is fine enough to reveal the temporal and spatial formation and dissipation of shockwaves. To demonstrate practicality of the algorithm, we present results from a 15 minute long video filmed by a Berkeley Highway Laboratory camera. Figure 1(a) shows the layout of the Berkeley Highway Laboratory. Figure 1(b) shows a single frame covering about one mile of freeway.

Kastrinaki *et al.* [1] provide an extensive survey of state of the art traffic applications of video processing, including road traffic monitoring. Our methodology falls into the general category of optical flow, techniques of which are reviewed in Beauchemin and Barron [2]. Applications of optical flow to traffic monitoring have been based on detecting and tracking individual vehicles to estimate speed, density, and flow. The following are representative examples. Autoscope [3] detects and tracks vehicles within a detection zone (roughly a rectangle the size of a vehicle) and integrates their spatial and temporal signatures to measure their speeds. The ACTIONS system [4], detects and tracks moving objects by estimating optical flow vectors which are then clustered to create candidate moving objects. The MORIO system [5] infers polyhedral models for objects moving relative to a stationary camera. The TITAN system [6] uses mathematical morphology to extract individual vehicle features, aggregates them into individual vehicles, and tracks them. It is capable of monitoring stretches of the motorway of up to about 1000 feet, depending on the height of the camera. The images of individual cars need to be separated. In Fathy and Siyal [7], a morphological edge detector and background differencing are used to identify and track vehicles and calculate traffic parameters. Coifman *et al.* [8] developed a feature-tracking algorithm to extract individual vehicle trajectories from video data by detecting predefined features from images, grouping them, and tracking the groups of features to produce trajectories. Dailey *et al.* [9] developed a method to estimate mean traffic speed, using an edge-detecting algorithm to find centroids and estimating mean speed from centroid movement in successive images.

However, algorithms that rely on identifying and tracking individual vehicles are not feasible for use with images of poor quality and over a wide area. If the spatial resolution is poor, vehicles in the frame do not show distinctive lines or features throughout the whole span of view and thus can be neither clearly identified nor tracked. If a vehicle only occupies a small number of pixels, its features may be hard to identify and furthermore can change as the precise position of the vehicle within the pixel grid changes. Vibration due to wind causes further difficulties, as do shadows and occlusions in congested traffic. Grant *et al.* [10] report on an extensive test of Autoscope on freeways in Atlanta, Georgia, showing that counts degraded in accuracy as the distance of the count location to the camera increased out to a maximum of about 400 ft. Although they did not directly measure the quality of speed estimation as a function of distance, they conjecture that it is similar to that of the volume counts.

In contrast, the method set forth in this paper does not

depend on explicitly identifying and tracking individual vehicles. We demonstrate that it is robust to occlusion and shadows, which can be seen in Figure 1(b), and to the camera motion that is evident in the videos. We compare the estimates obtained from the video to those from high frequency loop detectors.

II. Data

The data used in this study were generated from a 15-minute video of about a one mile stretch of highway I-80 in Berkeley, California. A video test bed, the Berkeley Highway Laboratory, consists of 12 cameras on the roof of Pacific Park Plaza, a 30-story building beside the highway. The analog cameras have S-VHS video recorders attached. Figure 1(a) illustrates the setup and the coverage of each camera.

Among six cameras looking north, the field of view of camera N6 is furthest down I-80E. It covers the longest stretch (about one mile long), from the Ashby Avenue on-ramp to University Avenue off-ramp. But it produces relatively poor quality images due to the poor angle and resolution. Figure 1(b) is a still frame from camera N6. The spatial resolution is such that a pixel in the near field of view is about 5 feet, whereas those furthest from the camera are about 15 feet. The images also suffer from occlusions by vehicles and their shadows and from camera motion.

Despite the poor quality of the images, the data from camera N6 is potentially informative, for example for studying the effect of on-ramp flow on highway performance. Also, it covers the longest stretch, about one mile, and in principle, information extracted from this camera can be combined with that from the higher resolution cameras, which have smaller fields of view. Algorithms developed to analyze images from camera N6 should be applicable to other conventional traffic cameras.

For the analysis, a 15 minute long tape was digitized at the rate of 10 frames per second and the results saved in ppm file format. Each frame, like Figure 1(b), has 800x640 pixels and each pixel has intensities for red, green, and blue channels.

III. Methodology

Because of the camera resolution, it would be at best extremely difficult to apply a feature detecting and tracking algorithm. Hence, it is necessary to develop a new way to extract information from the images. We developed an algorithm for this purpose. The algorithm proceeds as follows: First, an "*intensity profile*" of each lane in the direction of traffic flow is extracted from each frame. Arranging the profiles in time order gives intensity flow in the time-space domain, in which vehicles appear as stripes, or moving peaks. Second, the speed estimate at (t, x) in the time-space domain is computed via searching a pair of best matching patterns at time $t + \tau$ and $t - \tau$ in terms of the L_1 norm (sum of absolute values of differences) and estimated as the slope of the line connecting the two centers of the pair. For discussion and references to such correlation-based matching methods, see Beauchemin and Barron [2].

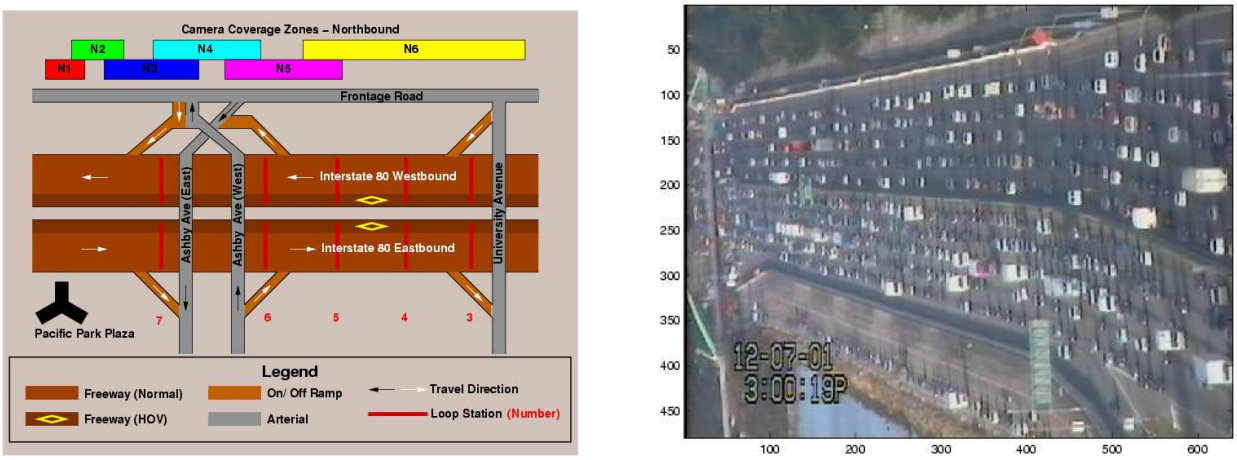
The algorithm has a number of advantages. First, it does not involve vehicle tracking, which can be computationally very expensive, and hence makes it more efficient to process a large amount of data. Secondly, in estimating local speed it does not compute a gradient, or a weighted average of speeds of moving features around the location of interest. It uses the L_1 norm in finding the best matching pattern and hence is robust to noise. Later in this section, we show that under certain conditions the algorithm is equivalent to finding a weighted median of the speeds of moving peaks.

A. Intensity Profiles

To generate the intensity profile for a lane, a mask (M) is created on a particular frame, a so-called reference frame. The mask passes intensities of pixels in a region of interest. A masked image is presented at the top of Figure 2. Once masked, a frame contains intensities of pixels in the region. The pixels have three integers between 0 and 255 for red, green, and blue channels. Because these three intensities are strongly correlated at the pixel level, we take the average of red, green, and blue intensities. Then, we scan across the lane (orthogonal to the direction of traffic) and calculate the maximum average intensity along the direction of traffic, which we refer to as a maximum intensity profile. The maximum intensity profile corresponding to the top image of Figure 2 is presented in the bottom of Figure 2.

The mask should be created for each frame, because the camera may be constantly shaking due to strong wind. To create masks automatically, we find mappings (Ψ_t) from the reference frame to every frame during the time period and use the new transformed masks ($\Psi_t(M)$). First, we choose four fixed objects (reference objects) on the reference frame and place square windows centered at them. Then, in each frame we search for the best matching patterns corresponding to the squared areas centered at the reference objects. Once the patterns are found, we use the coordinates of the center pixels of the square windows to compute the projection matrix. For more information, refer to chapter 5 of Hartley and Zisserman [11].

We repeat this process for each frame and stack the maximum intensity profiles in time order to obtain an intensity flow on the time-space domain. The intensity fades as the vehicle moves away from camera, and the road surface intensity varies depending on the location. To correct for this, at each location of the highway we determine the maximum and minimum intensity during the 15 minutes and form the ratio of the difference between the intensity and minimum to the difference between maximum and minimum. After this background correction, the intensity is standardized between 0 and 1. To change units from pixels to feet, we computed the projection matrix from image to the real world, using the real dimension of I-80 (For a detailed computation, also refer to chapter 5 of Hartley and Zisserman [11]). As a final step, we interpolate the intensities on a finer grid, which are equally spaced by about 5 feet. Two examples are presented in Figure 3.



(a) Berkeley Highway Laboratory Layout

(b) A frame from Camera N6

Fig. 1. Camera Setup

The resulting intensity flow is similar to a trajectory plot in the time-space domain in that the stripes, or moving peaks, contain information about the traffic flow on the highway. But it differs in that one curve does not necessarily correspond to one vehicle. Rather one vehicle can be shown as two lines, or two vehicles traveling closely together can appear as one moving peak. Small or dark vehicles may be barely perceptible. Fine discrimination is not needed for subsequent speed estimation.

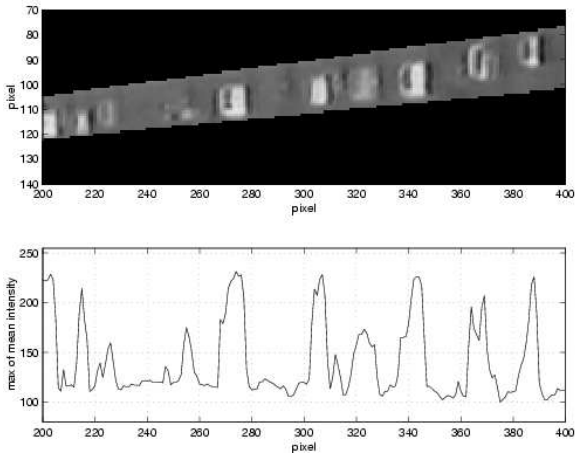


Fig. 2. A filtered image and the corresponding intensity profile

B. Speed Estimation

The idea behind the estimate is that a locally constant intensity pattern represents travel at a locally constant speed. Hence, after a short period of time, the same pattern will show up again at the travel distance above the previous location in the time-space domain.

To estimate the local speed at (t, x) in the time space domain, we choose two rectangular windows of size w_t by w_x and center one at $(t + \tau, x)$ and the other at $(t - \tau, x)$. We then move the window centered at $t + \tau$ up (increase x) and the other one at $t - \tau$ down (decrease x), computing the L_1 norms for the pair. The norm will be minimized at the travel distance. By the nature of the image, the shifts are integer-valued. To deal with this discretization, we fit the L_1 norms near the minimizer to a quadratic function to find the interpolated distance which minimizes the norm. The speed is estimated as the ratio of the distance to the shifting parameter, τ, which is equivalent to the slope between the two centers of the pair of the best matching patterns.

In the examples presented here, we use the L_1 norm criteria and set the window size to 30 seconds by 90 feet, τ as 3 second, and the searching area as 0 to 80 MPH. The step by step outline of the algorithm is as follows.

Let $I(t, x)$ denote the intensity at location (t, x) in the time-space domain.

1) Fix the local window size, $w_t \times w_x$, and the shift parameter τ,
2) For $d = 0, \ldots, m$, compute
$$D(d) = \sum_{t=t_0-w_t}^{t_0+w_t} \sum_{x=x_0-w_x}^{x_0+w_x} |I(t+\tau, x+d) - I(t-\tau, x-d)|.$$
3) Fit $D(d)$ near the minimum to a quadratic function of d.
4) Find d_0, which minimizes D.
5) The speed is estimated as d_0/τ.

Note that the algorithm does not involve gradient computation or slopes for individual lines on the intensity flow. After running the algorithm, we run a 2-dimensional median filter to remove noise, of size 40 second by 370 feet.

Also note that it simultaneously shifts two square windows (*centered shifting*), instead of fixing one window centered at (t, x) and shifting the other window at either $t + \tau$ (*forward shifting*) or $t - \tau$ (*backward shifting*) in time. As can be seen

from a Taylor series expansion, derivatives are estimated more accurately by central differences than by un-centered ones. The derivative of a quadratic function is estimated exactly by central differencing, but not by one-sided differencing.

Because the algorithm involves computing the sums of absolute difference over 2 square windows, it can be quite slow. To speed up, we use a subgrid within the square window instead of using all the intensities. Based on the empirical comparisons, we find even a coarse grid of resolution one second by about 10 feet (using 5% of data points) is sufficient to produce an estimate equally good as using all the data. Also, we need not evaluate the estimate at every location. Instead, we estimate the speed on a sparse grid and then interpolate.

To gain some insight into the nature of the estimate produced by the algorithm we now consider an idealized continuous version. Suppose features, $A_j(t,x)$, have disjoint supports I_j, are parabolic on I_j (3rd and higher order derivatives are zeros), and travel at the speed of v_j. The features correspond to the contributions of individual vehicles to the intensity profile, which we write as

$$f(t,x) = \textstyle\sum_j A_j(t,x) = \sum_j A_j(x - v_j t).$$

Now consider the minimizer of

$$\tfrac{1}{2\tau} \int_{t=t_0-w_t}^{t_0+w_t} \int_{x=x_0-w_x}^{x_0+w_x} |f(t+\tau, x+d) - f(t-\tau, x-d)|\, dxdt.$$

For small d and τ

$$\begin{aligned}
\hat{v} &= \arg\min_d \frac{1}{2\tau} \int_t \int_x |f(t+\tau, x+d) \\
&\quad - f(t-\tau, x-d)| dxdt \\
&= \arg\min_d \frac{1}{2\tau} \int_t \sum_j \int_x |A_j(t+\tau, x+d) \\
&\quad - A_j(t-\tau, x-d)| dxdt \\
&= \arg\min_d \frac{1}{2\tau} \int_t \sum_j \int_x \left|2(d - v_j\tau) A_j'(x - v_j t)\right| dxdt \\
&= \arg\min_d \sum_j \left|\frac{d}{\tau} - v_j\right| \int_t \int_x \left|A_j'(x - v_j t)\right| dxdt \\
&= \arg\min_d \sum_j \frac{S_j}{S} \left|\frac{d}{\tau} - v_j\right|
\end{aligned}$$

where $S_j = \int_{t_0-w_t}^{t_0+w_t} \int_{x_0-w_x}^{x_0+w_x} |A_j'(x - v_j t)| dxdt$ and $S = \sum_j S_j$. The minimizer of the final expression above is a weighted median of the individual velocities v_j in which the weights are S_j/S. That is, it is the median of a discrete probability distribution which has masses S_j/S on the values v_j . Vehicles with large derivatives of their individual intensity profiles thus contribute most heavily to the estimate. The median, however, is insensitive to extreme velocities. By contrast, if we were to use the sum of squared deviations rather than the sum of absolute deviations, the argument above shows that the estimate would be a weighted mean, and less robust to extreme v_j . This argument formalizes the notion that the shifting and matching algorithm estimates a weighted median velocity over a region of space and time.

IV. Results

Two intensity flows during 15 minutes from 3:00pm on 17th of December 2001 to 3:15pm on the same day are presented in Figure 3. We picked two lanes; the right-most (5th) lane of I-80E and the 3rd lane of I-80W. We chose the two lanes for the following reasons. The 5th lane merges with the Ashby on-ramp at the near field of the frame (at around 500 feet) and the inflow creates congestion. The 3rd lane of I-80W experienced the worst stop-and-go traffic and had more trucks than any other lanes during the 15 minutes.

During the 15 minutes, the east bound traffic experienced moderate congestion, shown in the intensity flow as changes in slopes of lines. Examining the figure carefully, one can see some lines disappear and appear, caused by lane-changing and occlusions from the shadows of vehicles traveling in the next lane. The west bound lanes experienced very heavy traffic. Also recall that in the 3rd lane there were the most trucks. In the intensity flow, trucks appear as broad stripes. In the intensity flow, we observe flat patterns lining up, which shows shockwaves propagating against the traffic. The lane also experienced the most frequent occlusion from vehicles and their shadows, due to the stop and go traffic in the next lane. In this lane, there are marks on the road to signify the off-ramp and they create horizontal stripes around 30, 60, 460 pixels even after background correction. However, the speed estimate is robust to these artifacts.

From the speed estimate of I-80E, we observe congestion due to the inflow from the Ashby on-ramp and corresponding shockwaves. We suspect that a traffic signal on Ashby Avenue caused periodic fluctuations in inflow and hence the pulsating series of shockwaves. Also note a pronounced shockwave originating at around 360 second and 0.5 mile from the University exit and travelling against the traffic at about 10 MPH.

The I-80W speed estimate shows even stronger oscillations shockwave evolution, and some variation in their velocities of propagation. The figure shows that the shockwaves typically travel at about 10 MPH. Because we do not observe where they originated and dissipated, we cannot verify how long the shockwaves traveled before dissipating, based solely on camera # 6. For now, we conjecture that the shockwaves were created further downstream on I-80W, about 1.3 miles south of the Ashby off-ramp, at the notorious split of I-80W into I-580S, I-880, and I-80W. Further investigation using tapes from cameras # 1-5 would reveal more information.

To check our estimates, we compared them to loop detector data. Loop detectors are located at stations 3, 4, and 5 in the order of distance from the University exit; refer to Figure 1(a). Unfortunately, the stretch had been paved recently and we could not locate precisely where the loops were. So, we approximated the loop locations by those of the cabinets and pull-boxes of the loop counter stations, which are located at the side of the wall of I-80E. In Figure 5, the dots are the point

estimates(vehicle by vehicle) from the loop data. The speed estimates corresponding to the cabinet(pull-box) locations are shown as the solid lines.

TABLE I
MEANS AND STANDARD DEVIATIONS (MPH)

	East bound	West bound
Station 3	-3.7 (2.1)	-2.0 (3.8)
Station 4	0.3 (1.6)	0.0 (1.8)
Station 5	4.3 (1.8)	1.6 (3.1)

The figures show that the estimates are very close to the loop data during the 15 minutes and pick up most of the oscillations. There are some systematic differences, which may be attributable to the imprecision of loop detector locations. Note that the speed ranges and traffic conditions for the west and east bound lanes are very different, yet the estimates are very consistent in both cases. The means and standard deviations of the errors between the estimates and the loop data are reported in Table I.

V. CONCLUSIONS AND DISCUSSION

The results above demonstrate the potential of our algorithm for processing a video recording from a traffic camera, providing a useful tool to study numerous traffic issues, such as the effect of an on-ramp, the evolution and dissipation of queuing and congestion, and for monitoring highway performance. Despite its poor quality image, camera N6 provides very useful information in these regards. For some purposes, simple functionals of the estimated velocity field may be sufficient. For example, travel times can be estimated by tracing through the field, or the average velocity over space at a given time can be computed.

Although the results we have shown are quite reasonable, we will study several issues in more detail in the future. One is the choice of the region on which to base shifting and matching. In principle, the rectangle could be as small as one pixel in time and several pixels in space, or vice-versa. The computing time is faster for smaller rectangles, but the results are noisier (a defect which can be ameliorated, however, by smoothing the estimates). Smaller rectangles yield a finer resolution in space and time, but again at the cost of noise. Larger rectangles localize less and are computationally more expensive, but produce less noisy estimates. In principle, the regions need not be rectangular and weight functions, such as Gaussian kernels, can be used instead of uniform weighting. Initial experimentation indicates that the final results are quite insensitive to these choices, but further study is necessary to optimize the algorithm for speed and accuracy.

In addition to further improving speed estimation, we are developing algorithms to extract the other macroscopic parameters, flow and density, from the intensity profiles. This is more difficult than velocity estimation. Counting is more feasible in the near field of view, and the results can be propagated through the estimated velocity field to obtain estimates of density and flow in the far field of view. We will also investigate the still more challenging problem of detecting lane changing.

Finally we mention that we have used our method on MPEG and AVI compressed video, with little degradation of the results. This may be useful if data are to be transmitted prior to analysis.

ACKNOWLEDGEMENTS

This research was supported by Partners in Advanced Highways and Transportation and by a grant from the National Science Foundation. We wish to thank Dan Lyddy for assistance with the data, Ryan Lovett and Phil Spector for assistance with computation and Peter Bickel, Z Kim, Jaimyoung Kwon, and Erik van Zwet for helpful discussions.

REFERENCES

[1] V. Kastrinaki, M. Zervakis, and K. Kalaitzakis, "A survey of video processing techniques for traffic applications," *Image and Vision Computing 21*, pp. 359–381, 2003.

[2] S. Beauchemin and J. Barron, "The computation of optical flow," *ACM Computing Surveys*, vol. 27, no. 3, pp. 433–467, Nov. 1995.

[3] P. Michalopoulos, "Vehicle detection video through image processing: the autoscope system," *IEEE Trans. Veh. Technol.*, vol. 40, pp. 21–29.

[4] W. Enkelmann, "Interpretation of traffic scenes by evaluation of optical flow fields from image sequences," *IFAC Control, Computers, Communications in Transportation*, 1989.

[5] L. Dreschler and H.-H. Nagel, "Volumetric model and 3d-trajectory of a moving car derived from monocular tv-frame sequences of a street scene," *Computer Vision, Graphics, and Image Processing*, vol. 20, pp. 199–228, 1982.

[6] J. Blosserville, C. Krafft, F. Lenoir, V. Motvka, and S. T. Beucher, "New traffic measurements by image processing," *IFAC Control, Computers, Communications in Transportation*, pp. 35–42, 1989.

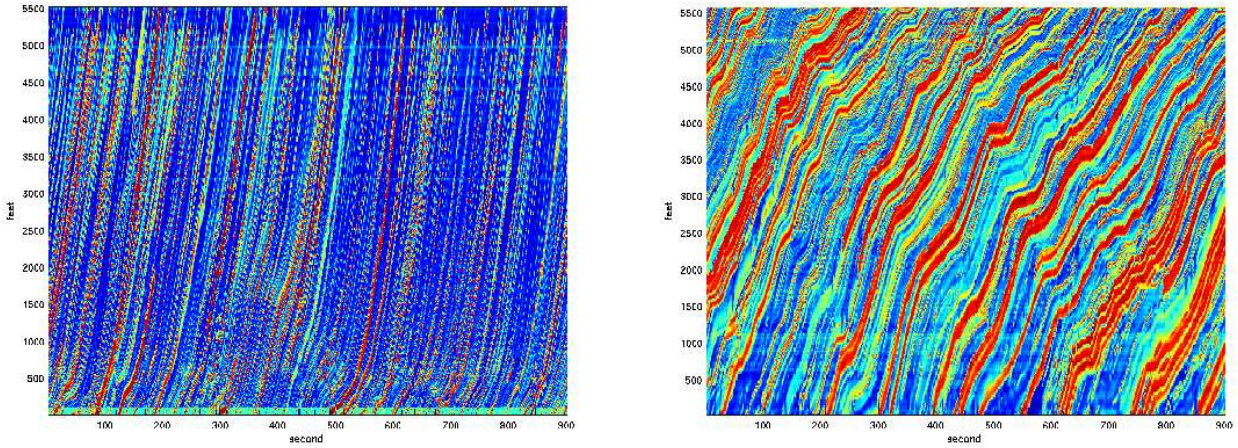
[7] M. Fathy and M. Siyal, "An image detection technique based on morphological edge detection and background differencing for real-time traffic analysis," *Pattern Recognition Letters*, vol. 16, pp. 1321–1330, 1995.

[8] B. Coifman, D. Beymer, P. Mclauchlan, and J. Malik, "A real-time computer vision system for vehicle tracking and traffic surveillance," *Transportation Research, Part C*, vol. 6C(4), pp. 271–288, Aug. 1998.

[9] D. Dailey, F. Cathey, and S. Pumrin, "An algorithm to estimate mean traffic speed using uncalibrated cameras," *IEEE Trans. Intell. Transport. Syst.*, vol. 1, no. 2, pp. 98–107, June 2000.

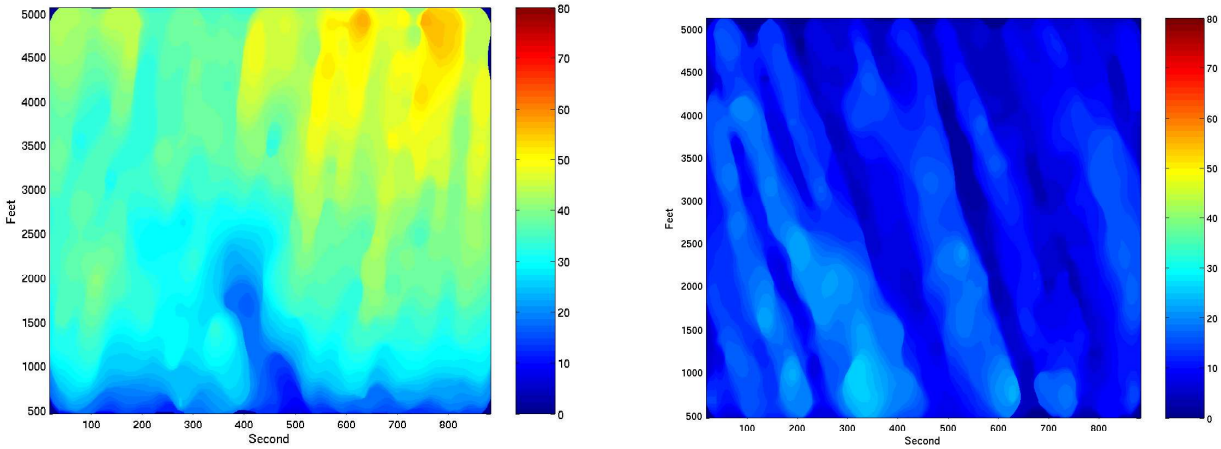
[10] C. Grant, B. Gillis, and R. Guensler, "Collection of vehicle activity data by video detection for use in transportation planning," *ITS Journal 5*, pp. 342–361, 2000.

[11] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge, UK: Cambridge University Press, 2000.



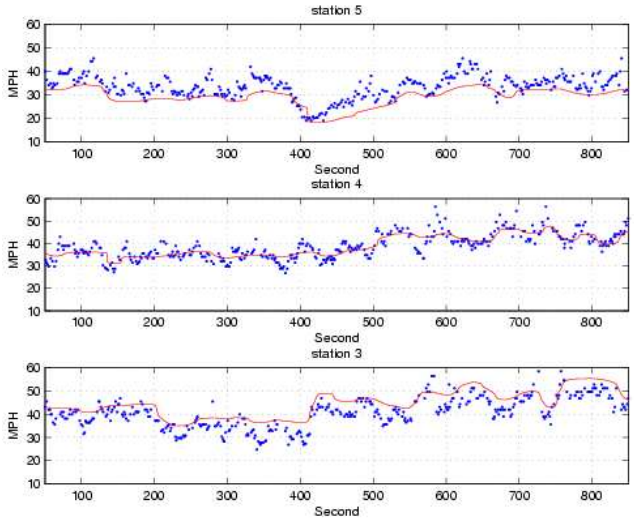
(a) I-80 East Bound (b) I-80 West Bound

Fig. 3. Intensity flows

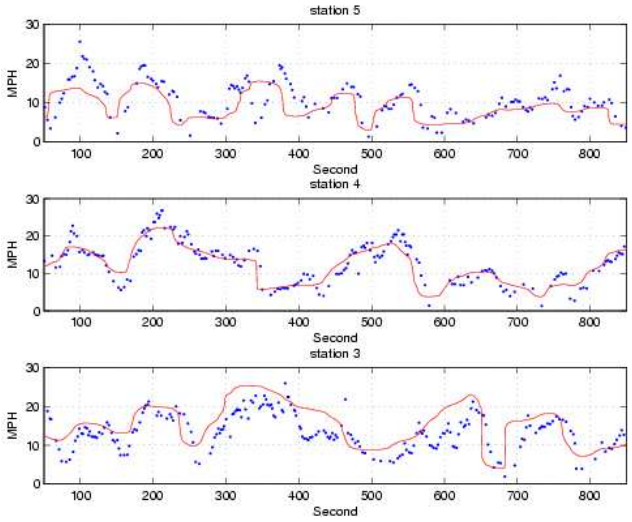


(a) I-80 East Bound (b) I-80 West Bound

Fig. 4. Estimated velocity fields



(a) I-80 East Bound



(b) I-80 West Bound

Fig. 5. Comparsion between the estimate and the loop data