Activity Analysis in Wide-Area Aerial Surveillance Video

Ryan Feather James W. Davis
Dept. of Computer Science and Engineering
Ohio State University
Columbus, OH 43210 USA
{feather,jwdavis}@cse.ohio-state.edu

Abstract

In this work, we present methods for tracking and activity analysis that are scalable and appropriate for challenges present in wide-area aerial video. After an initial preprocessing stabilization step, we use a constrained interest point matching algorithm to generate weak tracks of vehicles in the scene. We then present algorithms that exploit the tracks to recognize traffic activity patterns of flow direction, uni vs. bidirectional traffic, acceleration/deceleration zones, and bidirectional stops via analysis of characteristic velocity patterns. Lastly, we provide quantitative and qualitative results of our activity analysis algorithms using both synthetic and real imagery.

1. Introduction

Persistent aerial surveillance is an emerging domain with needs to assess ongoing activity in large areas for tasks such as force protection, traffic management, and urban planning. Automated analysis tools are important as the size of the area monitored and the number of objects to track are difficult to manage manually. Both tracking and activity analysis research in wide-area aerial surveillance (WAAS) video are recent and limited. Here, we introduce a scalable approach to handle the challenges inherent to tracking objects and analysis of traffic activity in such video.

In this paper we employ the publicly available Greene 07 WAAS image data set collected by AFRL over Greene County near Dayton, OH [1]. This data consists of approximately 8.5 minutes of mosaiced grayscale imagery from six video sensors at a frame-rate of approximately 1.2 Hz. The frames have a resolution that exceeds 10K $\times$ 10K pixels. To provide a constant top-down view, the aerial vehicle continuously circles the area of interest. Figure 1 provides an example of the imagery.

Challenges with this data set (and other similar data sets) include coarsely stabilized video, large number of pixels, stitching misalignments, occlusions, low spatial resolution, parallax motion, and intensity differences between stitched images (see Fig. 2). Additionally, the low frame-rate and low spatial resolution means that vehicles may appear small with large pixel displacements between frames, making tracking difficult.

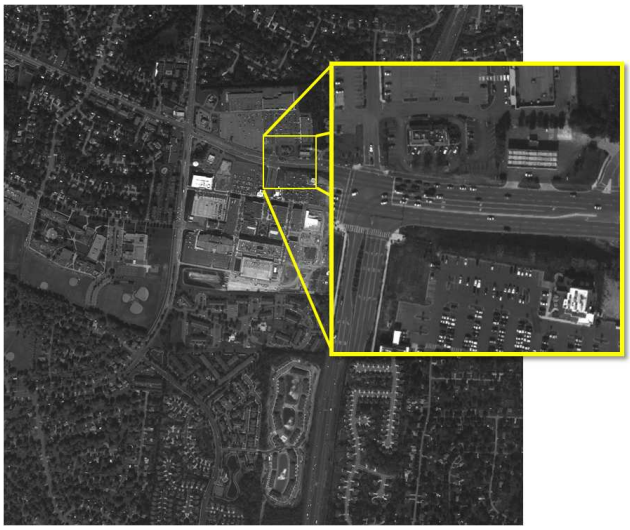

Figure 1. Example image portion from the Greene 07 data set showing approximately 4.7K $\times$ 4.7K pixels (full image is over 10K $\times$ 10K). The inset shows details in a smaller 500 $\times$ 500 patch.

Our tracking approach has been designed to be suitable for the WAAS imagery presented and is based on an interest point descriptor matching method augmented by motion masking and motion constraints. The output of our approach is "weak" tracking data, typically consisting of multiple track fragments per objects. We then provide a set of activity analysis algorithms for the weak tracks aided by a simple road map to recognize activity patterns including traffic flow direction, uni vs. bidirectional traffic, acceleration and deceleration zones, and areas where bidirectional traffic stops (*e.g.*, stoplight). We provide both quantitative and qualitative results using synthetic and real (Greene 07)

(a) (b) (c)

Figure 2. Imagery challenges present in the Greene 07 data set. (a) Intensity differences at seam. (b) Stitching misalignments. (c) Geo-registration/stabilization error between frames (150 pixel displacement in this example).

data to validate the approach.

We begin with a brief overview of related work in Sect. 2. We next describe the preprocessing stage in Sect. 3. The proposed tracking method is presented in Sect. 4, followed by an overview of the traffic analysis approach in Sect. 5. Experimental results are presented in Sect. 6, and a summary is given in Sect. 7.

2. Related Work

Current approaches to tracking in WAAS imagery typically focus on building tracks by matching motion blobs using cost minimization schemes. In [9], moving object blobs detected through use of median background modeling are given frame-to-frame assignments using the O(n^3) Hungarian algorithm, and neighboring object trajectories are employed as context to avoid track switching. In [11], blob detection is performed via three-frame subtraction, and GIS road network information is used to provided context. While these approaches may be appropriate for gathering tracks in WAAS imagery, they assume that the consumer of tracking output requires nearly perfect tracking results. Our approach does not require strong tracking data, which allows us to opt for a more efficient tracking means.

Gaining an understanding of road structure and traffic behavior through activity analysis is also an area of study. In [8], vehicular tracks are modeled as polynomials and tracks are clustered in polynomial coefficient space for lane detection. The work then categorizes lanes as entries or exits and as belonging to primary or secondary roads. While this approach could prove useful for discovering lane directionality, it assumes a higher quality of tracking data than we expect can be readily extracted from current WAAS data. The approach in [6] proposes an occlusion reasoning algorithm and learns vehicle count and speed information from tracking data. The occlusions deal specifically with vehicles occluding other vehicles, which is not present in our top-down view data set.

3. Preprocessing

We first need to re-stabilize the frames of the Greene 07 data set to produce accurate tracking results (see Fig. 2(c)). This is performed by matching SURF [3] interest points between frames and using RANSAC [5] to robustly fit a model consisting of rotation, translation, and scaling parameters. Each image in the sequence is registered/transformed back to the first frame in the sequence. To counter the possibility of SURF interest points clustering about certain highly structured areas, points are randomly selected across evenly spaced areas of the image.

This stabilization, and the subsequent tracking, is performed in a gridded fashion similar to that of [9] to allow our system to be easily run in a parallel manner. We divide the image into overlapping regions of 500 $\times$ 500 pixels and process each region independently.

4. Tracking

In contrast to strong tracking approaches which aim to find one track per object that persistently follows the object throughout the entire scene, we opt to collect "weak" tracks [10] which consist of multiple and frequently broken tracks per object. Weak tracking data can be obtained in a more efficient and real-time manner while still providing useful information about the motion of objects in the scene (as we will show). Our approach involves use of a motion mask and SURF interest point matching followed by the application of constraint priors. Again, a 500 $\times$ 500 pixel grid is employed to track objects in each grid independently.

4.1. Motion Mask

Our tracking algorithm exploits areas of motion to help reduce the number of feature points to track to only those belonging to moving objects (vehicles). We detect areas of motion by thresholding the magnitude of normal flow between consecutive images

$$\frac{|I_t|}{\sqrt{{I_x}^2 + {I_y}^2}} > Threshold \tag{1}$$

The use of normal flow is efficient and helpful in filtering out unwanted motion caused by the motion parallax of buildings (generating small normal flow magnitudes). We note that very tall structures may create greater parallax which can not be easily removed by the flow magnitude alone.

4.2. Constrained SURF Point Matching

Within the motion mask, we next track feature points. One popular real-time approach to weak tracking is

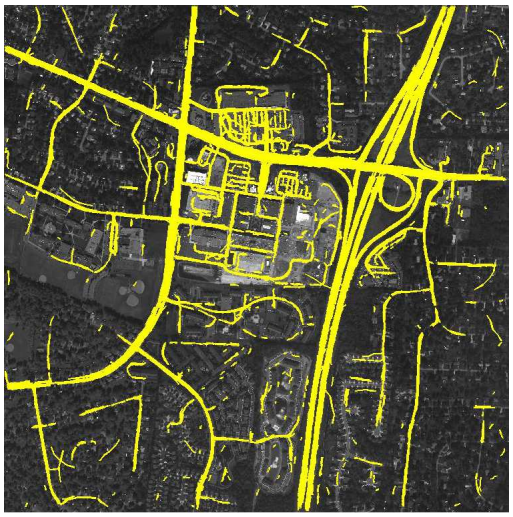

Figure 3. Image of all valid track segments.

Figure 4. Mask of major roads in the Greene 07 data set.

the well-known Kanade-Lucas-Tomasi (KLT) tracker [7]. However, as vehicles in WAAS video may often move many pixels (over 60 pixels in Greene 07) between frames and are small (≈20 pixels), the KLT hierarchical optical flow method does not perform well. Hence, we instead employ a fast SURF-based method.

SURF [3] interest points are detected on blob-like structures and use scale-space localization to build scale invariant description vectors. We limit the detection of SURF points to those within the previously described motion mask. We also subject all points to a maximum spatial matching distance. The maximum matching distance is determined by the object's previous velocity plus a constant factor to allow for acceleration (initial track points use a default distance of 100 pixels).

To disambiguate matching when a set of tracks P in one frame match a single point r in the next frame, and the ratio between the best two descriptor distances is small (¡ 0.65), we choose the match P_c such that

$$c = \underset{j}{\operatorname{argmin}}\{\|r - (P_j + v_j)\|\} \tag{2}$$

where v_j is the velocity computed from the last two observations in the corresponding track. When choosing between a potential match from a set including a single-observation track and a multi-observation track, the single-observation track point is chosen to avoid extending longer tracks with possibly ambiguous information. A track is terminated if no point match is found in the following frame or if the motion between frames for the point is too small (object stopped moving). We also enforce that tracks exist for a minimum number of frames (4 time-steps). A track switching mismatch from one object to another often creates a sharp change in velocity, so a maximum angle ($\frac{\pi}{6}$ radians) and a deceleration (8 pixels per frame) criterion are employed to identify this behavior and split the track.

While our activity analysis methods will ultimately use only frame-to-frame tracking information (velocity), we still employ multi-frame tracking data (as opposed to using only optical flow), as more reliable evidence is obtained to discriminate true object motion from non-desirable motion due to stitching errors, sensor noise, parallax motion, and geo-registration errors.

The output of our approach on the Greene 07 data set is shown in Fig. 3. This figure shows an aggregate image of all tracks collected over the sequence (over 38K tracks).

5. Activity Analysis

With the particular data set employed, we are constrained to only 8.5 minutes of data. As the duration is not long enough to learn reliable behavioral trends and relationships, we instead show the reliability of detecting traffic direction, uni vs. bidirectional flow, acceleration/deceleration zones, and bidirectional stop locations. While our tracking approach was designed to be more general purpose, our activity analysis algorithms focus on traffic activity patterns occurring in the context of an established road network, hence we employ a simple mask of selected roads. We use only the track information from these roadways (though the mask could be incorporated into the tracking algorithm). We do not believe that the availability of such a road mask is an unreasonable assumption as such GIS information is widely available, and in lieu of such, many approaches exist to automatically extract road structures from aerial imagery (*e.g.*, [4, 2]). We manually created the mask based on existing information (Google Maps). Note that we treated on/off-ramps as separate roads. The road mask employed is shown

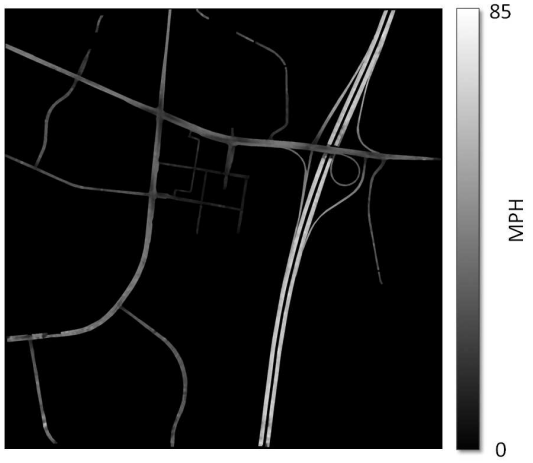

Figure 5. Velocity road map.

in Fig. 4.

Given the road mask and tracking data, we compute the average velocity at each location of the roads. For each pixel location (i, j) in the velocity map M, we compute the average velocity of nearby track observations p using a Gaussian kernel K

$$M_{i,j} = \frac{\sum_k v_k \cdot K(\|p_k - \langle i, j \rangle\|, \sigma)}{\sum_k K(\|p_k - \langle i, j \rangle\|, \sigma)} \quad (3)$$

where v_k is the corresponding velocity and $\sigma = 10$. The resulting velocity mapping is shown in Fig. 5. From this map, we see the expected distribution of speed values of traffic on the interstate within the 60-70mph range whereas side road traffic is near 45 or 25mph.

We also will need an effective way to analyze data *along* a given road. To do this, we employ the road center lines (spine for each road). Center lines are often directly available from GIS data, however, we used morphological operations on our road mask to obtain the paths. We performed skeletonization followed by a cleanup/removal of spurs until a single beginning and ending point of the skeleton existed.

5.1. Road Traffic Flow

Knowing information about the lane structure and traffic flow (uni vs. bidirecitional traffic) on a road is useful and helpful for further activity analysis. We characterize traffic flow on a road using a two-bin angular histogram. For each road, we traverse the center line of the road, and at each position, a cross section of the road normal to the center line is taken from the velocity map. For each location in this cross section, the velocity angle with respect to the center line direction is calculated, quantized $(0, \pi)$, and the corresponding bin in the angular histogram is incremented. After the road is fully traversed, if the ratio of the minimum to maximum bin count is greater than 0.75, we consider the road to be bidirectional (else it is unidirectional).

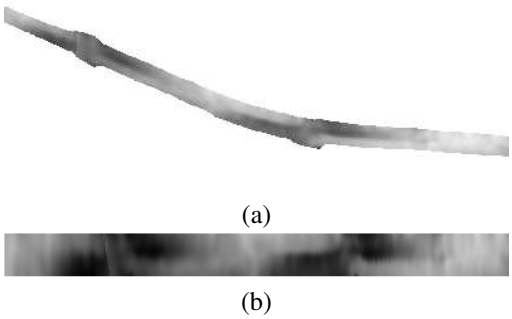

Figure 6. Canonical road projection. (a) Original velocity mapped road segment. (b) Canonical straightened representation.

5.2. Canonical Road Projections

Using the road mask and center lines as a guide, we next create a straightened road projection of selected major roads in the velocity map to facilitate further activity analysis.

First, we standardize the roads such that traffic flow moves from left to right in the projection. With a bidirectional road, this corresponds to the flow on the bottom half of the projection. We then straighten the road by sampling cross sections normal to the center line and normalize them to a fixed height (100 pixels). Some distortion is unavoidable when straightening a highly curved path.

To address areas with unreliable (few) track observations, we do not include in our sampling any pixel where the track density is within the lowest 5% for that road. An example road and its canonical projection are shown in Fig. 6.

5.3. Acceleration/Deceleration Segmentation

From the canonical road projections, we first find the areas of acceleration/deceleration. We begin by creating a 1D velocity profile V of a lane (or road for one-way roads) by taking a 2D Gaussian weighting of velocities (centered on the lane/road) in the canonical road projection with $\sigma_y = (width)/6$ and $\sigma_x = 25$ where, in our case, the lane width is 50 for the canonical road projection (road width is 100). For the top lane in Fig. 7(a), we show the velocity profile in Fig. 7(b). An acceleration profile, A, is then created by a first-difference of the velocity profile (see Fig. 7(c)). Zero-crossings are then found in A (where A_i and A_{i+1} do not share the same sign) to identify the boundaries of the accel/decel zones. To handle areas where values may remain slightly positive or negative yet represent an

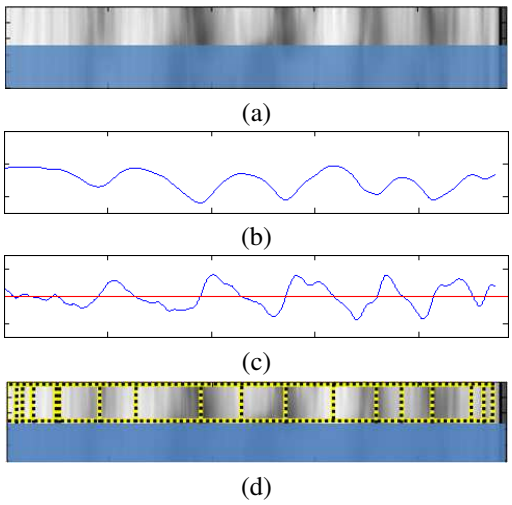

Figure 7. Segmentation process for one lane of a bidirectional road. (a) Canonical road projection for a two lane road with the top lane visible. (b) Velocity profile. (c) Acceleration profile. (d) Final lane segmentation results.

area of constant velocity, we augment the set of boundaries to also include cases such that ($|A_i| > \epsilon$ and $|A_{i+1}| \leq \epsilon$) or ($|A_i| \leq \epsilon$ and $|A_{i+1}| > \epsilon$) where $\epsilon = 0.01 * max(|A|)$. The endpoints of the profile are also included in the boundary set.

Given the set of boundaries B for a lane/road, the set of accel/decel segments S is defined as

$$S_i = V_{j \to k} : j = B_i, k = B_{i+1}. \quad (4)$$

Figure 7(d) shows the final accel/decel segment partition.

For each segment, we then calculate the difference in velocity at the opposing endpoints in the direction of the traffic flow. The +,- signs of the differences (denoting acceleration/deceleration) will be used in the following bidirectional stop detection, and the absolute difference in velocity can be used to rank and threshold segments to yield only the strongest acceleration and deceleration zones if desired.

5.4. Bidirectional Stop Detection

Bidirectional stops, as caused by traffic lights at crossroads, give a distinctive "checkerboard" pattern of traffic velocity in a canonical road projection (see Fig. 6(b)) with a particular alignment of accel/decel zones in each lane (as shown in Fig. 8). For detecting bidirectional stops, we require four accel/decel segments (two from each lane on a road) where each segment touches at least one segment in the other lane. Segments in a lane must display a deceleration followed by an acceleration in the direction of traffic flow.

If the acceleration/deceleration zone spatial context is met for a given portion of a road (Fig. 8), we then score and threshold the potential bidirectional stops by calculating the absolute velocity difference for each composing segment and threshold based on the median of those four values (removing the potential stop if the median absolute value is less than 10mph).

6. Experimental Results

Our approach was tested on both synthetic and real data. As the acceleration/deceleration zone and bidirectional stop detections are the primary goal of this work, we focus the majority of the results on these tasks.

6.1. Synthetic Experiments

Estimating locations where vehicles tend to accelerate or decelerate from real data is possibly prone to arbitrary judgements, therefore, we first provide synthetic data for quantitative analysis of our approach. Our synthetic experiments are formed using a road having a single bidirectional stop (see Fig. 8). We model the ideal paths of objects on our road as tracks spanning the length of the road (1000 $\times$ 70 pixels) in opposing directions. Tracks are randomly initialized in the first 50 pixels of a lane, and observations of the track are assigned locations based on the ideal average velocity that an object in the zones would travel. We process these synthetic tracks through the road mapping, width normalization, segmentation, and bidirectional stop detection stages of our algorithm. The output is the set of pixel segments corresponding to the accel/decel zones comprising the bidirectional stop.

Using this setup, we vary the number of tracks (per lane), observation location noise (σ), and track breakage probability (b). To measure the localization performance of our algorithm, we calculate precision, recall, and F1 score based on the number of pixels correctly assigned to the appropriate ground truth zone. Due to the random nature of the input, we report the average results from running each experiment 100 times.

As a default, we set σ=2, b=0.1, and the number of tracks equal to 1000. In the first experiment, we varied the track count from 5 to 1000. The results in Table 1 show that our algorithm's F1 Score is largely independent of the number of tracking observations over orders of magnitude. Next, we varied the probability that tracking observations may be omitted (*i.e.,* breaking tracks) which in Table 2 shows a predictably low effect on the observed F1 score. When varying location noise, we found an expected decrease in performance with larger σ, as shown in Table 3, due to a decrease in accurate segmentation with high location noise.

For all tests, the lower precision scores can generally be attributed to an extension of acceleration/deceleration segments (into areas where constant velocities should exist) due to the smoothing involved in the mapping and calculation of the velocity profiles. The few percent recall er-

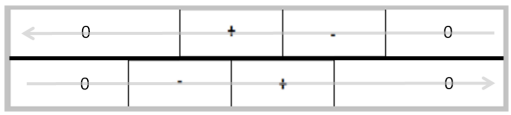

Figure 8. Ideal accel/decel zones of a bidirectional stop.

Track Count	F1 Score	Precision	Recall
5	0.759	0.624	0.971
10	0.752	0.615	0.972
25	0.763	0.630	0.971
50	0.769	0.649	0.971
100	0.778	0.655	0.971
250	0.782	0.655	0.971
500	0.783	0.657	0.971
1000	0.783	0.657	0.970

Table 1. Results for bidirectional stop localization across the number of tracks.

Break Chance	F1 Score	Precision	Recall
0%	0.784	0.657	0.971
10%	0.784	0.657	0.971
20%	0.784	0.657	0.971
30%	0.784	0.657	0.971
40%	0.783	0.657	0.971
50%	0.783	0.656	0.971
60%	0.782	0.654	0.971
70%	0.779	0.650	0.971

Table 2. Results for bidirectional stop localization across track breakage probabilities.

σ	F1 Score	Precision	Recall
0	0.775	0.645	0.970
1	0.783	0.657	0.970
2	0.783	0.657	0.970
4	0.781	0.654	0.971
8	0.760	0.623	0.972
16	0.736	0.595	0.970
32	0.655	0.512	0.917

Table 3. Results for bidirectional stop localization across observation noise levels.

ror was typically caused by ambiguity where a deceleration zone ends and an acceleration zone begins. In these areas, track velocities are effectively zero for a few pixels, which makes choosing the precise end/start location difficult. In all experiments, the bidirectional stop was detected with a conservative threshold (10mph, see Sect. 5.4).

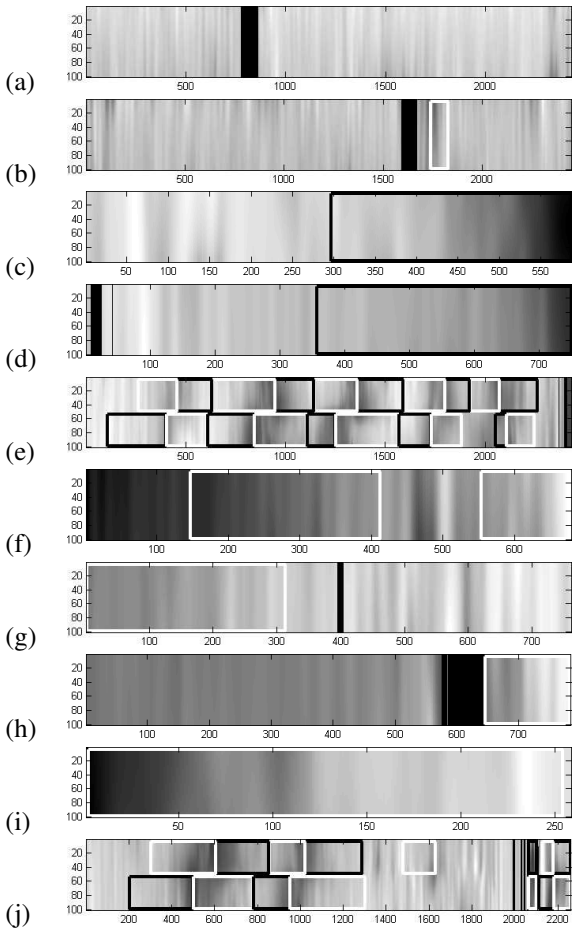

Figure 9. Detected acceleration and deceleration zones. Light boxes indicate an increase in velocity and dark a decrease (in the direction of traffic).

6.2. Greene 07 Results

We chose ten of the most highly traveled roads in our area of interest in the Greene 07 dataset for analysis. Out of the ten roads, both of the bidirectional roads were correctly detected with all other roads being detected as unidirectional. We measured the success of the bidirectional stop detection approach by comparing the results to the number of known stops in the actual road network. On the roads analyzed, there were seven true bidirectional stops in the scene (caused by traffic lights). We detected six correct bidirectional stops with one stop missed and one false detection. The missed stop comes from a section where two such stops are close enough together that track velocity observations blended together in a way that masked a set of acceleration and deceleration zones. The false detection corresponded to the presence of a side road combined with poor tracking in that area due to occlusions.

Figure 9 displays the strongest acceleration/deceleration zone detections in the canonical road projections. Strong

acceleration/deceleration zones are detected by scoring segments based on the absolute velocity segment difference and retaining those with an absolute difference greater than 10mph. We can also use detected bidirectional stops as strong evidence of true, but weak magnitude, accel/decel zones. Each section detected as part of a bidirectional stop is also treated as an acceleration or deceleration detection (three additional zones were added to Fig. 9). From Fig. 9, it is clear that our algorithm is able to detect areas where the majority of vehicles change velocity significantly, such as on-ramps and off-ramps. A pictorial overview of our acceleration/deceleration and bidirectional stop results with the caption labels from Fig. 9 is shown in Fig. 10.

There was a deceleration zone found on both off-ramps (see Fig. 9(c),(d)). The exit zones detected do not span the entire canonical projections due to a tendency of vehicles to maintain a relatively constant velocity then decelerate quickly near the end. At least one acceleration was found on each on-ramp (see Fig. 9(f)-(i)). Again, vehicles do not always evenly accelerate over the entire length of an on-ramp. Additionally, a zone of acceleration was found on the "J" shaped road where it intersects another road (see Fig. 9(j), right of the middle in the top lane). A significant amount of traffic entered the road and accelerated at that location.

Some areas with false detections, including an area on the interstate (see Fig. 9(b)) and areas at the bottom end of the "J" shaped road (the right end of Fig. 9(j)), correspond to areas where poor tracking data was collected due to occlusions. These problems could potentially be alleviated with use of further GIS information (known occlusion areas). One additional zone was found in Fig. 9(f). This is likely due to blending of vehicle velocities from the on-ramp with those of vehicles on the interstate when the on-ramp thins to only a few pixels wide.

In addition to the bidirectional stop and acceleration/deceleration zones found in the Greene 07 data set, the effects of other phenomena (*e.g.*, checkpoints or traffic accidents) could potentially be detected by our approach as we are able to find such activities with short duration data (within 8.5 minutes in this case).

7. Summary

We presented techniques applicable for tracking vehicles and analyzing activity in wide-area aerial surveillance video. Our approach to tracking is based on constrained SURF interest point matching and motion masking. We then used a canonical mapping of tracking output to a simple road network to analyze various traffic flow patterns using velocity signatures. Detected patterns include traffic flow direction, uni vs. bidirectional roads, acceleration/deceleration zones, and bidirectional stops. We demonstrated our approach with quantitative experiments on synthetic data and qualitative analysis with the Greene 07 data set. The results demonstrate a good first step in understanding scene activity present in wide-area aerial surveillance video.

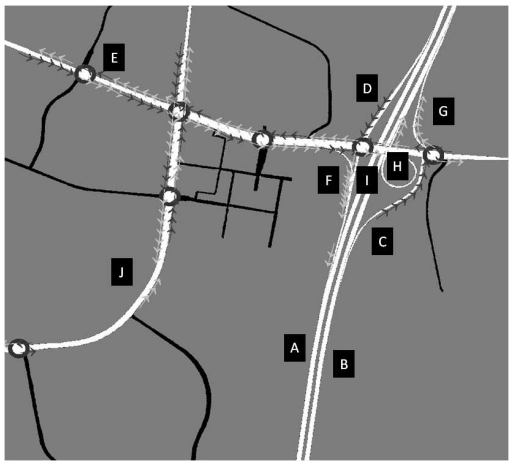

Figure 10. White roads were the subject of analysis, black roads are only provided for reference, circles indicate bidirectional stop detections, darker arrows are decelerations, and lighter arrows are accelerations. Letters correspond to roads in Fig. 9.

8. Acknowledgments

This research was supported in part by AFRL under contract No. FA8650-07-D-1220.

References

[1] Air Force Research Laboratory (WPAFB, Dayton). Greene 2007 data collection. Sensor Data Management System. https://www.sdms.afrl.af.mil/index.php?collection=greene07.

[2] M.-F. Auclair-Fortier, D. Ziou, C. Armenakis, and S. Wang. Survey of work on road extraction in aerial and satellite images. Technical report, Département de Mathématiques et d'Informatique, Université de Sherbrooke, 1999.

[3] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-up robust features (SURF). *CVIU*, 110, 2008.

[4] E. Christophe and J. Inglada. Robust road extraction for high resolution satellite images. In *ICIP*, volume 5, 2007.

[5] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Comm. ACM*, 24:381 – 395, 1981.

[6] Y.-K. Jung and Y.-S. Ho. Traffic parameter extraction using video-based vehicle tracking. In *Int. Conf. Intel. Trans. Sys.*, 1999.

[7] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *IJCAI*, volume 2, 1981.

[8] J. Melo, A. Naftel, A. Bernardino, and J. Santos-Victor. Detection and classification of highway lanes using vehicle motion trajectories. *Trans. Intel. Trans. Sys.*, 7(2):188 – 200, 2006.

[9] V. Reilly, H. Idrees, and M. Shah. Detection and tracking of large number of targets in wide area surveillance. In *ECCV*, 2010.

[10] K. Streib and J. Davis. Extracting pathlets from weak tracking data. In *Int. Conf. Adv. Vid. Sign. Bas. Surv.*, 2010.

[11] J. Xiao, H. Cheng, H. Sawhney, and F. Han. Vehicle detection and tracking in wide field-of-view aerial video. In *CVPR*, 2010.