Recognizing Human-Vehicle Interactions from Aerial Video without Training

Jong Taek Lee*, Chia-Chih Chen*, and J. K. Aggarwal
Computer & Vision Research Center / Department of ECE
The University of Texas at Austin
{jongtaeklee, ccchen, aggarwaljk}@mail.utexas.edu

Abstract

We propose a novel framework to recognize human-vehicle interactions from aerial video. In this scenario, the object resolution is low, the visual cues are vague, and the detection and tracking of objects are less reliable as a consequence. Any methods that require the accurate tracking of objects or the exact matching of event definition are better avoided. To address these issues, we present a temporal logic based approach which does not require training from event examples. At the low-level, we employ dynamic programming to perform fast model fitting between the tracked vehicle and the rendered 3-D vehicle models. At the semantic-level, given the localized event region of interest (ROI), we verify the time series of human-vehicle relationships with the pre-specified event definitions in a piecewise fashion. With special interest in recognizing a person getting into and out of a vehicle, we have tested our method on a subset of the VIRAT Aerial Video dataset [11] and achieved superior results. Our framework can be easily extended to recognize other types of human-vehicle interactions.

1. Introduction

Recognizing human-vehicle interactions is a challenging problem in computer vision. It is of interest in security, automated surveillance, and aerial video analysis. For example, the detection of a person getting into a vehicle may provide the first level alert of abnormal events. The discovery of frequent human-vehicle interactions from aerial video may help pinpoint a warehouse or signify the migration of a group of people. As shown in Fig. 1, due to limited image resolution, air turbulence, cloud coverage, objects temporarily out of field of view, and the constantly moving aerial vehicle, the recognition of human-vehicle interactions from aerial view is a much more challenging task than those in normal scenarios. In this work, we propose a general framework to recognize human-vehicle interactions from an aerial video. More specifically, we illustrate our framework using the cases of recognizing a person getting into and out of a vehicle.

Figure 1. (a) The aerial image of a person approaching the front door of a vehicle. The bounding box of the person is magnified to illustrate this challenging scenario. (b) The snapshots of a vehicle taken from an UAV in every 5 seconds.

With careful and sometimes repeated inspections, a human observer can recognize human-vehicle interactions from aerial video without seeing any examples from the same setup. This is because humans are capable of constantly tracking objects in low quality imagery and are proficient at reasoning about the underlying event without seeing it in its entirety. However, there are two major difficulties for machine vision to perform the same task as well. First, most machine learning algorithms require a sufficient number of training samples to perform reliable recognition; however, the cost is high for taking aerial videos and annotating example sequences. Second, the key moments of human-vehicle interactions always happen when persons are in close proximity of the vehicle; as a result, a human tracker is easily subject to drift due to overlapped object structures in blurry low-resolution imagery.

Our method is a temporal logic based approach which does not require the tracking of human objects nor event-level training examples. Our system starts with process-

* These two authors contributed equally to the paper.

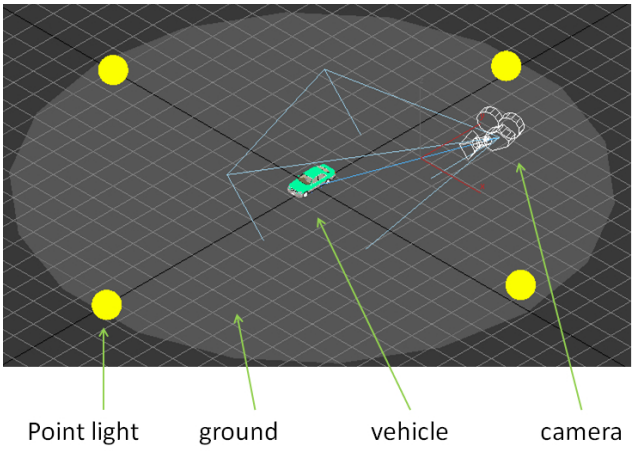

Figure 2. A ray tracer with 3-D scene including a vehicle.

ing the bounding box sequences of the tracked vehicles. To estimate the location and the orientation of a vehicle, we train Support Vector Machines (SVM) [3] classifiers with samples rendered from 3-D vehicle models and ray tracing. Then we search for the optimal solution of vehicle states in a sequence of frames using dynamic programming under a Markovian assumption. Given the aligned 3-D vehicle models, we use the localized door (or trunk) regions together with local human detection results to reason about their interactions over time. We define the temporal flow of a human-vehicle interaction based on the sub-events of particular changes in their spatial relationships. Weights are manually assigned to the interaction associated sub-events according to their relative importance to the composition of the interaction. The likelihood of individual interactions is computed by matching an observation sequence with the formal event representations and binning the weighted votes of matched sub-events. To the best of our knowledge, our work is the first paper which explicitly tackles the problem of recognizing human-vehicle interactions in aerial video.

This paper is arranged as follows. We discuss the previous work in Section 2. Section 3 introduces the technical details of our dynamic programming based 3-D vehicle alignment. Our temporal logic based interaction recognition scheme is presented in Section 4. We demonstrate the experimental results in Section 5 and conclude in Section 6.

2. Previous Work

There has been an emerging interest in recognizing human activities from aerial view in the past few years. The pioneer work by Efros *et al.* [7] characterizes human actions at a distance by using an optical flow based descriptor. They use the rectified optical flow components to describe the motion patterns between pairs of figure-centric bounding boxes. On the same subject, Chen and Aggarwal [4] present a joint feature action descriptor, which combines features selected from human poses and motion in a supervised manner. Later in their work [5], they propose a novel representation called *action spectrogram*, which characterizes human activities by both local video content and occurrence likelihood spectra of body parts' movements. Their method has been shown to further the recognition accuracy on two low-resolution human activity datasets [12, 11].

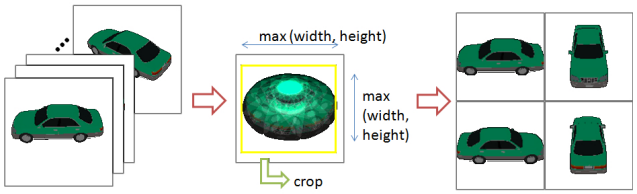

Figure 3. Positive vehicle training sample generation.

Ivanov and Bobick [8] use stochastic context-free grammars on human-vehicle interaction recognition, Joo and Chellappa [9] apply attribute grammars, and Tran and Davis [15] adopt Markov logic networks to recognize human-vehicle interactions. Their methodologies focus on the high-level understanding of human-vehicle interactions. Lee *et al.* [10] propose an view-independent approach for the recognition of human-vehicle interactions. They perform vehicle detection and localization through the use of 3-D vehicle models with chamfer matching. Ryoo *et al.* [13] also recognize person-vehicle interactions in the presence of occlusions using event context under a Bayesian formulation. However, all the mentioned approaches are not applicable to our scenario, where the interactions are filmed from a moving platform and the accurate characterization of object contour and motion is not possible. For the evaluation of human activity and human-object interaction recognition algorithms, the newly published VIRAT Video Dataset [11] includes videos collected from stationary ground cameras as well as unmanned aerial vehicles (UAV). This large-scale benchmark dataset features 6 types of human-vehicle interactions in both camera settings.

3. Alignment of 3-D Vehicle Model

The robust alignment of a 3-D vehicle model is essential for the system to extract event ROI and to estimate the human-vehicle spatial relationship. In this section, we propose a novel and generic approach for the optimal search of vehicles states by the alignment of 3-D vehicle models. In the following subsections, we explain the details of our methodology from 1) 3-D model rendering, 2) localization of a vehicle centroid, 3) estimation of vehicle orientation, and 4) the optimal search of vehicle states using dynamic programming.

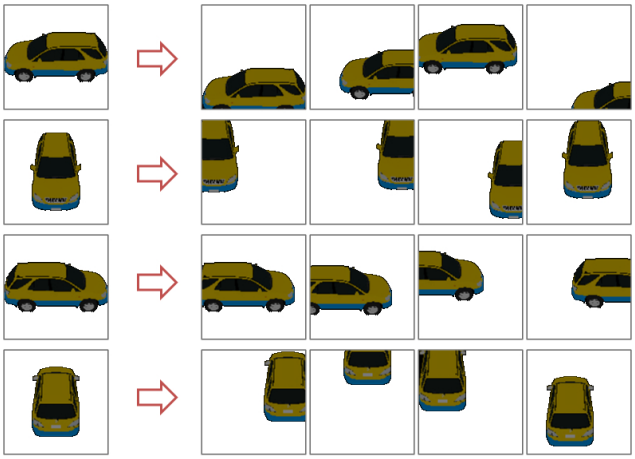

Figure 4. Negative vehicle training samples.

3.1. 3-D Vehicle Model

Collecting training samples for vehicle detection is a tedious task, and it is impractical to collect them in all possible view points. Therefore, we use ray tracing with 3-D vehicle models to generate controlled training images with detailed annotations. In order for our ray tracer to generate synthetic training samples, we create the scene of vehicles using the following descriptions: we place a vehicle model in the center of a 3-D space and a ground plane model below the vehicle model. Then, four point light sources are placed on the front, rear, left, and right of the vehicle model, respectively. Finally, a scene camera is added and controlled by the system as shown in Fig. 2. By adjusting the position and direction of the camera, our ray tracer can generate the projected images of a 3-D vehicle in different orientations.

Without loss of generality, our ray tracer disables reflection and refraction. It is not possible for the system to simulate the detailed characteristics of the texture of vehicles and the ground from most aerial video data due to low resolution scenes and compression errors.

3.2. Vehicle Location Detection

In this subsection, we explain the probabilistic approach to localize the centroid of the vehicle. Here, we assume that a vehicle is completely visible in the scene. We train an SVM classifier with the Histogram of Oriented Gradient (HOG) [6] features extracted from positive and negative vehicle sample images from 3-D vehicle models. The positive sample images have a vehicle at the center of the image and the negative sample images either have a vehicle near the boundary of the image or do not have a vehicle. Therefore, the trained binary SVM classifier can estimate the probability of the vehicle located at the center of a testing image.

The positive sample set has 720 images from 360 degree orientations and 2 vehicle types. The size of the projected image of a vehicle varies with respect to the camera views.

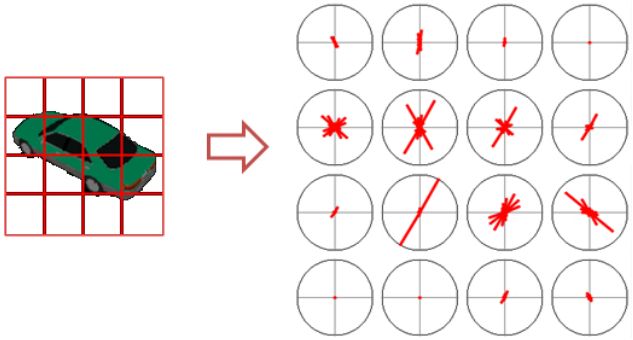

Figure 5. The configuration of our HOG descriptors.

These training samples are uniformly resized with a minimal margin as shown in Fig. 3. In this process, we measure the maximum length of the height and width of a vehicle for all orientations, crop the margin, and resize the cropped image.

The negative sample set is generated from the positive sample set. For every positive set sample, we generate four negative samples by relocating the vehicle image of the positive sample. For the generality of the negative sample set, the relocation is processed randomly in x and y direction. The system chooses the sample if the center of the relocated vehicle is far enough from the center of the image. Fig. 4 shows negative vehicle training samples.

Extracting reliable features from the generated training samples is as important as generating robust training samples. The HOG descriptor has shown its excellence in detecting humans and vehicles. Here, we compute HOG descriptors from square image patches using 4x4 cell rectangular blocks, 9 orientation bins, and an unsigned gradient as shown in Fig. 5.

We train an SVM classifier with the HOG descriptor of generated positive and negative sample images. The classifier has two classes: 1) *positive*, a vehicle is located in the center of an image and 2) *negative*, a vehicle is not located at the center of an image [14].

In order to correct vehicle location in the given image with a tracked vehicle presence, we scan the image by sliding a window to extract the HOG and calculating the probability of vehicle existence in the center of the window by the SVM classifier. The center of a window with the highest probability of vehicle existence ideally indicates the centroid of the vehicle in the given image.

3.3. Vehicle Orientation Estimation

Accurate vehicle orientation estimation enables the extraction of regions-of-interest (ROI) such as door regions after the vehicle location detection. This subsection explains the method to estimate 360 degree vehicle orientation in the order of 10 degree. The method of vehicle orientation estimation is similar to the method of vehicle location detection

Figure 6. Vehicle orientation estimation results.

in that both methods use generated images from a ray tracer with 3-D vehicle models and extract the HOG descriptor from the synthetic images.

We train an SVM vehicle orientation classifier with the 720 images and their HOG descriptors from positive samples of vehicle location detection. The classifier has 36 classes for every 10 degrees so that each class has 20 training images.

Then, the SVM classifier estimates the probabilities of vehicle orientations in the testing images. Our SVM classifier can perform correctly if the vehicle is located in the center of testing images (Fig. 6 (a)). If a vehicle is not correctly located (Fig. 6 (b)), or does not exist in the testing images (Fig. 6 (c)), the estimation of our classifier cannot be valid. Therefore, we need to combine the results of vehicle location detection and vehicle orientation estimation for the valid estimation of a vehicle states.

3.4. Dynamic Programming for the Optimal Search

In this subsection, we explain the method for the optimal search of vehicle states (location and orientation) in a sequence of frames using dynamic programming. For the event ROI extraction in Section 4, searching both the correct location and orientation of a vehicle is required. We first formulate the joint probability of vehicle location and orientation in a single frame under the assumption that vehicle location and orientation are conditionally independent. Then, we formulate the transition probability of vehicle states in two consecutive frames. With the formulated probability model and our dynamic programming solution, we are able to effciently search the optimal vehicle states in every frame.

The joint probability of vehicle location (l) and orientation (o) given an image (I), $P(l, o|I)$, is represented as a product of the probability of vehicle location, $P(l|I)$, and vehicle orientation given vehicle location, $P(o|l, I)$ as shown in Eq. 1. The estimation of $P(l|I)$ and $P(o|l, I)$ are explained in Subsection 3.2 and 3.3.

$$P(l, o|I) = \frac{P(l, o, I)}{P(I)} = \frac{P(l, I)}{P(I)} \cdot \frac{P(l, o, I)}{P(l, I)} = P(l|I) \cdot P(o|l, I) \quad (1)$$

We formulate the joint probability model of a sequence of the vehicle states given a sequence of corresponding images, $P(l_{\{1,t\}}, o_{\{1,t\}}|I_{\{1,t\}})$, under the Markovian assumption. Subscripts in equations indicate frame number(s) of variables. Let $S = \{l, o\}$, which indicates a vehicle state composed of l and o. Then, $P(l_{\{1,t\}}, o_{\{1,t\}}|I_{\{1,t\}})$ can be simplified as $P(S_{\{1,t\}}|I_{\{1,t\}})$. $P(S_{\{1,t\}}|I_{\{1,t\}})$ is expanded by using Bayes' Theorem as shown in Eq. 2

$$P(S_{\{1,t\}}|I_{\{1,t\}}) = \frac{P(S_{\{1,t\}}, I_{\{1,t\}})}{P(I_{\{1,t\}})} = \frac{P(S_t|S_{\{1,t-1\}}, I_{\{1,t\}})P(S_{\{1,t-1\}}, I_{\{1,t\}})}{P(I_{\{1,t\}})} \quad (2)$$

In Eq. 2, the term $P(S_{\{1,t-1\}}, I_{\{1,t\}})$ can be expanded as $P(S_{\{1,t-1\}}, I_{\{1,t-1\}}) \cdot P(I_t)$, and the term $P(S_t|S_{\{1,t-1\}}, I_{\{1,t\}})$ can be simplified as $P(S_t|S_{t-1}, I_t)$ by the Markovian assumption. Also, $P(I_t)$ and $P(I\{1, t\})$ are counted as constants given a sequence of images. Therefore,

$$P(S_{\{1,t\}}|I_{\{1,t\}}) \propto P(S_t|S_{t-1}, I_t)P(S_{\{1,t-1\}}, I_{\{1,t-1\}}) \quad (3)$$

In Eq. 3, the left term can be expanded as the following by using the Bayes' Theorem:

$$P(S_t|S_{t-1}, I_t) = P(S_t|S_{t-1})P(S_t|I_t)\frac{P(S_t)}{P(I_t)P(S_{t-1})} \quad (4)$$

The right term can also be expanded as the following by using the Bayes' Theorem:

$$P(S_{\{1,t-1\}}, I_{\{1,t-1\}}) = P(S_{\{1,t-1\}}|I_{\{1,t-1\}})P(I_{\{1,t-1\}}) \quad (5)$$

Under the assumption of the uniform prior probability distribution for S, Eq. 3 can be represented as in Eq. 6 by Eq. 4 and Eq. 5.

$$P(S_{\{1,t\}}|I_{\{1,t\}}) \propto P(S_t|S_{t-1})P(S_t|I_t)P(S_{\{1,t-1\}}|I_{\{1,t-1\}}) \quad (6)$$

By induction, Eq. 6 can be the product of a sequence of terms as shown in Eq. 7.

$$P(S_{\{1,t\}}|I_{\{1,t\}}) = P(S_1|I_1)\prod_{k=2}^{k=t}[P(S_k|S_{k-1})P(S_k|I_k)] \quad (7)$$

By replacing back S by l and o, we can derive the following equation:

$$P(l_{\{1,t\}}, o_{\{1,t\}}|I_{\{1,t\}}) = P(l_1, o_1|I_1)\prod_{k=2}^{k=t}[P(l_k, o_k|l_{k-1}, o_{k-1})P(l_k, o_k|I_k)] \quad (8)$$

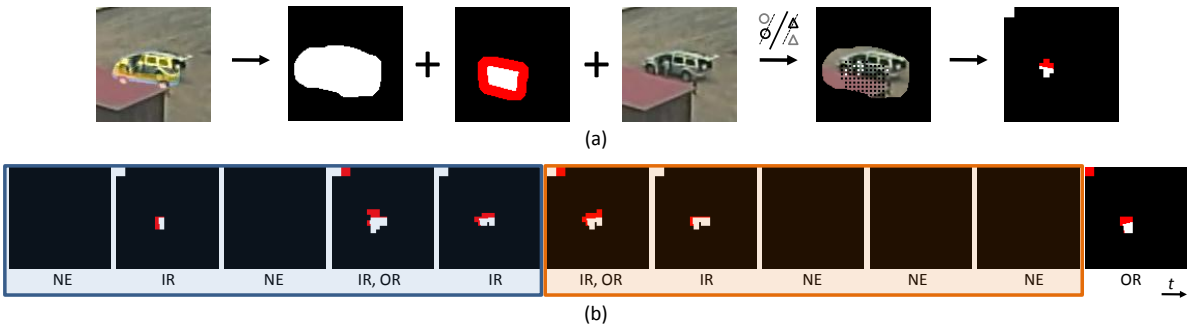

Figure 7. (a) The illustration of our human detection process. (b) Our system extracts interaction associated sub-events from a labeled human-vehicle sequence using a two-sided sliding window. The sliding window detects *Meets*(IR,NE), which contributes a weighted vote to the interaction of a person getting into a vehicle.

$P(l_k, o_k|l_{k-1}, o_{k-1})$ implies the transition probability of vehicle states in two consecutive frames, k and $k-1$. $P(l_k, o_k|I_k)$ is derived from Eq. 1. We assume that the transition probability model has an exponential distribution as follows:

$$P(l_k, o_k|l_{k-1}, o_{k-1}) = \lambda_l \cdot \lambda_o \cdot exp(-\lambda_l \cdot \|l_k, l_{k-1}\| - \lambda_o \cdot \|o_k, o_{k-1}\|) \quad (9)$$

After all, the problem of searching for an optimal sequence of vehicle states can be modeled as a Markov decision process. In order to have a finite set of states, locations are downsampled by every 5 pixels x 5 pixels windows, orientations are downsampled by every 10 degrees, and the original dataset with 30 fps (framesec) is downsampled in time to 2.5 fps.

Finding optimal states can be determined by a value iteration, V as follows:

Initialize $V(S_k)$ *arbitrarily*
loop for frame k
  loop for states at k, $S_k = (l_k, o_k)$
    loop for states at $k-1$, S_{k-1}
      $V(S_k) = max_{S_{k-1}} \{ S_P(l_1, o_1|I_1) \cdot \prod_{k=2}^{k=t}(P(l_k, o_k|l_{k-1}, o_{k-1}) \cdot P(l_k, o_k|I_k)) \}$
    end loop
  end loop
end loop

Through dynamic programming, the optimal search improves with each frame. When real-time processing is required, our system provides the optimal solution in the current frame. Without the time constraints, the optimal vehicle states in previous frames can be updated using a backward search.

4. Temporal Logic for Human-Vehicle Interaction Recognition

In this section, we introduce our temporal logic based approach, which derives the most likely human-vehicle interaction from low-level information. The low-level processing results include the localized event ROI and the locations of detected human objects, which are assigned with object states and parsed with modified temporal logic for interaction analysis.

4.1. Human Detection

After the process of 3-D vehicle model alignment, we perform human detection on the event ROI. As shown in Fig. 7 (a), for the recognition of a person getting into and out of a vehicle, our 3-D vehicle alignment provides the binary masks of the vehicle and its door regions. We dilate both types of masks and apply the vehicle mask to the bounding box so that arbitrary image content around the vehicle will contribute less to the human detector. The door mask after dilation is marked with a different color to indicate the peripheral of the ROI, which is used to capture a person's approach of ROI.

We use HOG to characterize human objects in low-resolution imagery. Our SVM based human detector is trained with HOG features extracted from manually cropped figure-centric bounding boxes and negative samples from patches around the figures. To save computation, the SVM window classifier only performs detection on grid locations of the event ROI. We train linear SVM to compute calibrated likelihood values [16], which are thresholded to indicate the likely grid locations of human presence. However, the detection accuracy inevitably suffers from the blurry low-resolution imagery as in Fig. 7 (a). Therefore, instead of taking the risk of missing true detections, a low threshold (< 0.5) is used to allow a certain amount of false positives. We perform connected component analysis on the detection grid coordinate to label the detected persons and remove unlikely blobs by area.

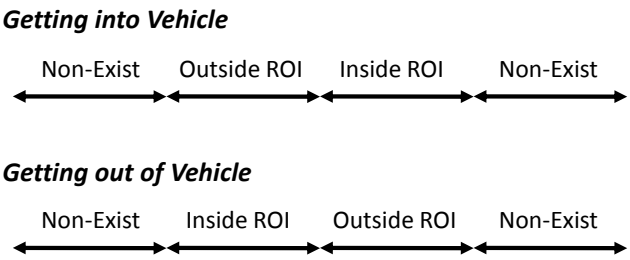

Figure 8. The formal event representation of a person getting into and out of vehicle.

To identify the human-vehicle spatial relationship in each bounding box, the dilated mask of event ROI is applied to the mask of human blobs. Based on the overlapped mask, our system estimates whether the person is inside the ROI (IR), outside the ROI (OR), or does not exist (NE) in the image patch. The specific permutations of these three event states are defined as the constituent sub-events of interactions.

4.2. Piecewise Temporal Logic

In Allen and Ferguson's classic temporal interval representation of events [2], an event is defined as having occurred if and only if the sequence of observations matches the formal event representation and satisfies the pre-defined temporal constraints. Temporal logic based approaches have been successfully applied for the recognition of human activities, human-human interactions, human-object interactions, and group activities [1]. Most importantly, instead of learning events from training examples, temporal logic allows the direct encoding of human knowledge. However, the recognition of interaction related sub-events from aerial video is far less accurate than that in regular scenarios. Therefore, capturing human-vehicle interactions by matching them against their complete event representation is rarely a success in our experiments.

We adopt a modified temporal logic approach to mine the pieces of event evidence embedded in a human-vehicle sequence. We name our method piecewise temporal logic (PTL), which is different from the classic temporal logic in two major aspects. First, our interaction representation is defined based on event states, from which the higher level interaction associated sub-events are derived. Second, our method recognizes interactions by comparing the weighted sums of detected sub-events, the temporal relationships among which are *not* taken into account.

We found that in a human-vehicle sequence, the moments of interaction related primitive actions are not always observable and cannot be reliably recognized. Therefore, we define human-vehicle interactions in terms of the event states that lead to them. Fig. 8 shows the formal event representation of a person getting into and out of a vehicle.

Interaction	Sub-event	Weight
Getting into vehicle	*Meets*(IR,NE)	2
	Meets(OR,IR)	1
	Meets(OR,NE)	0.5
	Finishes(IR)	-2
Getting out of vehicle	*Meets*(NE,IR)	2
	Meets(IR,OR)	1
	Meets(NE,OR)	0.5
	Starts(IR)	-2

Table 1. Interaction associated sub-events and their corresponding weights. IR, OR, and NE are shorts for human inside the ROI, outside the ROI, and does not exist (NE) in the image bounding box, respectively. *Meets*, *Starts*, and *Finishes* are the temporal predicates used to define their relationships.

Given the temporal flows of event states, interaction associated sub-events are defined in terms of the alternations of specific states. The set of predicates we used to describe the temporal relationships of event states include *Meets*, *Starts*, and *Finishes*. These sub-events are manually assigned with weights based on their relative importance to the actual occurrence of the interaction. For example, in Fig. 7 (b), the alternation of event states from IR to NE is more informative than the change from NE to OR for the detection of a person getting into a vehicle. Table 1 shows the interaction associated sub-events and their corresponding weights. Note that the exact values of sub-event weights cause much less effect on the system performance than their relative values.

It is a difficult task to extract instances of sub-events from a noisy event state sequence such as Fig. 7 (b). We propose to use a two-sided sliding window to detect interaction associated sub-events. As shown in Fig. 7 (b), the sub-event *Meets*(IR,NE) extracted from rear and front sliding windows is compared with the human encoded list in Table 1. The matched sub-event contributes a weighted vote to the corresponding bin of an event histogram. We use the sum of absolute sub-event weights in an event histogram to determine if any of the two interactions have ever occurred. The normalized event histogram indicates the occurrence likelihood of interactions.

5. Experimental Results

We test our methodology with the challenging VIRAT Aerial Video dataset [11]. The videos were taken in 30 frames per second with the resolution of 720 by 480 pixels. As shown in Fig. 9, the challenges posed by this dataset include low image resolution, vague object appearance and motion (due to air turbulence and video compression artifacts), time-varying views, changing weather conditions, salient shadow, and cluttered backgrounds.

Figure 9. The snapshots of four true positive (TP), two true negative (TN), one false negative (FN), and one false positive (FP) sequence are shown. We treat the subject human-vehicle interactions (getting into vehicle, getting out of vehicle) as the positive class and all other events (others) as the negative class.

There are a number of human-vehicle sequences in this dataset. However, we can only find 7 instances of a person getting into and out of a vehicle. We manually select 20 other types of human-vehicle interaction sequences, in which a person may be passing by or (un)loading the vehicle. Therefore, in our evaluation set, there are 4 sequences of a person getting into a vehicle, 3 sequences of a person getting out of a vehicle, and 20 other types of human-vehicle sequences. We use the same set of parameters for vehicle alignment and interaction analysis without any event-level training. Fig. 9 shows the snapshots of our testing sequences. Despite the differences in the types of vehicles, viewpoints, and interactions, our system is able to correctly detect the subject human-vehicle interactions from sequences such as the TP examples in Fig. 9. The FP and FN examples in Fig. 9 show the cases when our method fails. In the sequence of *"Getting into vehicle, FN"*, the approach of the person from the left was partially occluded by the building, and in the sequence of *"Others, FP"* the departure of the person from the ROI misled the system.

Our system demonstrates superior results on the search of the optimal vehicle states. In 20 sequences out of 27 testing sequences (74.1%), both the orientation and location of vehicles are correctly estimated. In the 6 isntances out of 7 incorrect sequences (22.2%), the locations of the vehicles are correctly detected but the vehicle orientations are 180° reversed. In spite of that, the ROI in those sequences were correctly located because of the symmetry of vehicle shape. In the other 1 instance (3.7%), the estimation of the vehicle orientations is incorrect. For interaction recognition, we analyze sub-events in every 4-second long two-sided sliding window. The system classifies a sequence as the subject human-vehicle interactions if its sum of absolute sub-event weights exceeds 1 and there is no tie in the event histogram. A sequence is recognized as other events if the sum of absolute sub-event weights is less than 1 or there is a tie in its event histogram. Fig. 10 shows the confusion matrix. By treating the subject human-vehicle interactions as the positive class and all other events as the negative class, the accuracy of our method on this evaluation set is 77.78% ((TP + TN) / (TP + TN + FP + FN)), the precision is 53.85% (TP / (TP + FP)), and the recall is 100.0% (TP / (TP + FN)).

6. Conclusions

We propose a general framework for the recognition of human-vehicle interactions from aerial view. Our method offers three major advantages to better resolve the challenges posed in this scenario. First, we adopt a temporal logic based approach to avoid the cost of manually collecting and labeling the training examples. Second, we employ a dynamic programming based 3-D vehicle model alignment technique, which accurately locates event ROI with the consideration of the previous alignment results. Third, based on classic temporal logic, we introduce the concept of PTL, which significantly improves the recognition per-

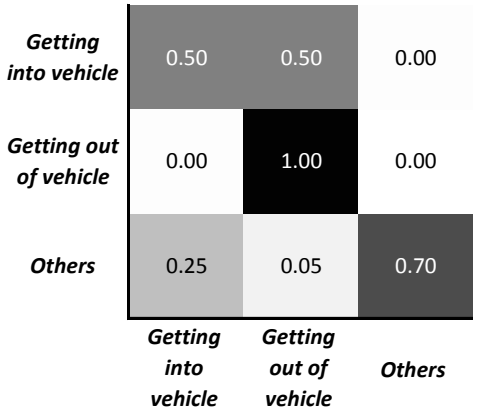

Figure 10. The confusion matrix of our method on a subset of the VIRAT Aerial Video dataset.

formance in our problem. PTL detects interaction sub-events by checking the temporal relationships between the event states. However, at the semantic-level, the temporal logics among the sub-events are not verified to induce the robustness against sequences of noisy sub-events. Furthermore, the proposed method can be generalized to recognize any kinds of human-vehicle interactions with the proper encoding and weighting of the temporal logics between event states. Most importantly, our method demonstrates high recognition accuracy on the challenging VIRAT Aerial Video dataset.

7. Acknowledgement

This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. HR0011-08-C-0135. Any opinions, findings, conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of DARPA.

References

[1] J. K. Aggarwal and M. S. Ryoo. Human activity analysis: A review. In *ACM Computing Surveys*, 2011. 6

[2] J. F. Allen and G. Ferguson. Actions and events in interval temporal logic. In *Journal of Logic and Computation*, volume 4, 1994. 6

[3] C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm. 2

[4] C.-C. Chen and J. K. Aggarwal. Recognizing human action from a far field of view. In *IEEE Workshop on Motion and Video Computing (WMVC)*, 2009. 2

[5] C.-C. Chen and J. K. Aggarwal. Modeling human activities as speech. *In IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011. 2

[6] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005. 3

[7] A. A. Efros, A. C. Berg, G. Mori, and J. Malik. Recognizing action at a distance. In *International Conference on Computer Vision*, 2003. 2

[8] Y. Ivanov, C. Stauffer, A. Bobick, and W. Grimson. Video surveillance of interactions. *Visual Surveillance, IEEE Workshop on*, 0:82, 1999. 2

[9] S. W. Joo and R. Chellappa. Attribute grammar-based event recognition and anomaly detection. In *IEEE Conference on Computer Vision and Pattern Recognition Workshop (CVPR-W)*, 2006. 2

[10] J. T. Lee, M. S. Ryoo, and J. K. Aggarwal. View independent recognition of human-vehicle interactions using 3-d models. In *IEEE Workshop on Motion and Video Computing (WMVC)*, 2009. 2

[11] S. Oh, A. Hoogs, A. Perera, N. Cuntoor, C.-C. Chen, J. T. Lee, S. Mukherjee, J. K. Aggarwal, H. Lee, L. Davis, E. Swears, X. Wang, Q. Ji, K. Reddy, M. Shah, C. Vondrick, H. Pirsiavash, D. Ramanan, J. Yuen, A. Torralba, B. Song, A. Roy-Chowdhury, and M. Desai. A large-scale benchmark dataset for event recognition in surveillance video. *In IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011. 1, 2, 6

[12] M. S. Ryoo, C.-C. Chen, J. K. Aggarwal, and A. Roy-Chowdhury. An overview of contest on semantic description of human activities 2010. *In International Conference on Pattern Recognition Contests (ICPR)*, 2010. 2

[13] M. S. Ryoo, J. T. Lee, and J. K. Aggarwal. Video scene analysis of interactions between humans and vehicles using event context. In *Proceedings of the ACM International Conference on Image and Video Retrieval*, CIVR '10, pages 462–469, 2010. 2

[14] B. Tamersoy and J. K. Aggarwal. Robust vehicle detection for tracking in highway surveillance videos using unsupervised learning. In *IEEE Conference on Advanced Video and Signal Based Surveillance (AVSS)*, 2009. 3

[15] S. D. Tran and L. S. Davis. Event modeling and recognition using markov logic networks. In *ECCV '08: Proceedings of the 10th European Conference on Computer Vision*, pages 610–623, 2008. 2

[16] T.-F. Wu, C.-J. Lin, and R. C. Weng. Probability estimates for multi-class classification by pairwise coupling. In *Journal of Machine Learning Research*, volume 5, pages 975–1005, 2004. 5