View Independent Recognition of Human-Vehicle Interactions using 3-D Models

Jong T. Lee[1], M. S. Ryoo[1,2], and J. K. Aggarwal[1]
[1]Computer & Vision Research Center / Department of ECE, The University of Texas at Austin, U.S.A.
[2]Robot Research Department, Electronics and Telecommunications Research Institute, Korea
jongtaeklee@mail.utexas.edu, mryoo@etri.re.kr, aggarwaljk@mail.utexas.edu

Abstract

Recognition of human-vehicle interactions is a challenging problem. The occlusion by vehicles and motion of humans contribute to the difficulty. In this paper, we present a novel approach for the view independent recognition of human-vehicle interactions. The shape based matching of synthetic 3-D vehicle models is used for accurate localization of vehicles and for the specification of regions-of-interest (e.g. *doors). In the proposed method, the system transforms the optical flow field based on the position of doors and the direction of a vehicle. This enables the system to extract view-independent features. Histogram of oriented optical flow (HOOF) and histogram of oriented gradient (HOG) characterize the optical flow and gradient, respectively. A support vector machine (SVM) classifier is trained for these view-independent features. Consequently, the system recognizes the interactions of a person entering a vehicle and getting out of a vehicle. Our method is applied to a dataset of human-vehicle interactions taken from 8 different viewpoints, composed of 120 video clips. The experimental results show that the system recognizes sequences of complex human-vehicle interactions with a high recognition rate of 86 %.*

1. Introduction

Over the last decade, considerable effort has been devoted to recognize human activities. However, it still remains a challenging problem in computer vision due to errors in the low-level processing, scene changes by the camera viewpoints, and the complexity of semantic representations [22]. In addition to simple human (*e.g.* single person) activity recognition, researchers have proposed methodologies for recognition of human-human (person-to-person) interactions [14, 17], human-object interactions [12], and group activities [16]. Human-vehicle interactions may be categorized as human-object interactions.

The problem of the recognition of human-vehicle interactions has not received the same level of attention as other interactions. In the study of most human-object interaction recognition, objects are smaller than humans (*e.g.* books, cups, phone, and so on [12]). People carry objects or stand near objects. On the other hand, most vehicles are larger than humans in general, and humans can easily be occluded by vehicles. The occlusion varies significantly as the viewpoint of a camera changes. Furthermore, a person may change the appearance (i.e. shape) of a vehicle by opening and/or closing its doors, making the problem more difficult.

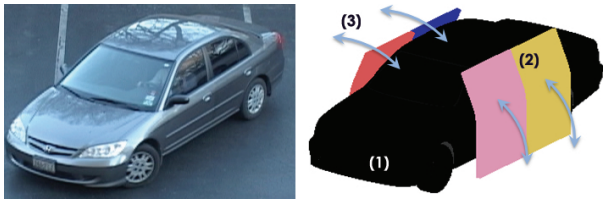

Figure 1. A raw image of a vehicle (left) and its matching synthetic 3-D vehicle model (right). The 3-D model is used for 1) the shape based matching by a vehicle-only model (black color), 2) extraction of regions-of-interest (ROIs) by four door regions (rectangular shape), and 3) transformation of motion features by the direction of door opening/closing on each door (double arrows).

In this paper, we propose a methodology for the recognition of human-vehicle interactions. The appearance of vehicles/humans, occlusion caused by vehicles, and motion of humans are view-dependent in the human-vehicle interactions. To solve these difficulties, our approach uses synthetic 3-D vehicle models for various purposes: 1) localization of vehicles by the shape based matching, 2) regions-of-interest (*e.g.* doors) specification, and 3) transformation of the optical flow field (see Fig. 1). The transformation is accomplished by measuring the direction of door opening or closing that fits the optical flow field. As a result, our system is able to extract view-independent features. We train an SVM classifier with the view-independent features for the classification of interactions.

Our contribution is a view-independent system that recognizes complex human-vehicle interactions using 3-D vehicle models. The system processes a dataset taken from

various viewpoints. The proposed approach has several benefits over previous approaches. First, our approach is able to extract view-independent features from human motion. Consequently, the system requires fewer training data from various viewpoints to achieve the same performance as previous systems which use view-dependent features. Second, our approach is able to reduce computation time and to recognize multiple occurrences of interactions. This advance is possible because we specify ROIs based on localization of vehicles and their fitted 3-D models, while most of the previous approaches specified ROIs based on detection of humans.

The paper is structured as follows. Section 2 describes previous work related to our paper. Section 3 presents a system overview. In Section 4, 5, and 6, we present our methodology for the recognition of human-vehicle interactions. In particular, Section 4 describes low-level processing (*e.g.* vehicle detection) of our system. Section 5 discusses the selection of features, and provides a methodology to transform the features for view-independent feature extraction. Section 6 illustrates a classifier for atomic interactions on each frame and representations for the recognition of composite interactions. Our experimental results are presented in Section 7. The paper concludes with Section 8.

2. Previous works

The problem of vehicle detection and tracking is a critical issue in vehicle related research. The problem has been studied in a variety of environments. The cameras are placed in various places; poles, buildings, traffic lights, and inside a vehicle. The scenes are also taken from different types of roads: straight roads, crossroads, and highways.

Sun *et al*. [20] presented a review of the problem of vehicle detection and integrating detection with tracking. Jun *et al*. [7] and Tamersoy and Aggarwal [21] proposed systems that detect vehicles in order to count the number of vehicles in highway traffic. Lee *et al*. [8] presented a novel system to detect illegally parked vehicles using 1-D transformation and compared the system with state-of-art systems.

Joo and Chellapa [6] recognized activities in a parking lot, such as parking, dropping off, and picking up. The activities of "dropping off" and "picking up" are similar to "getting out of a car" and "getting into a car." For the recognition of such activities, they used attribute grammars to represent the activities. Their contribution was on the representation of specific activities using the attribute grammars, not on the accurate detection of objects or motions. Therefore, their system was neither fully automatic nor view-independent.

Several researchers have tried to use 3-D models to recognize more complex activities from arbitrary viewpoints. The usage of 3-D models is focused on human representations for human recognition [10] or object representations

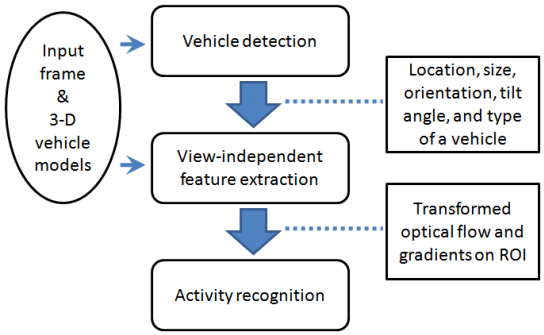

Figure 2. An overview of the system

for object detection [19]. Song and Nevatia [19] proposed a methodology to detect and track vehicles using 3-D models from various viewpoints. They extracted 2-D shape templates from 3-D models and matched the templates with observed foreground. They formed a hypothesis for vehicle information and refined it by a data driven Markov Chain Monte Carlo (MCMC) process.

The descriptors based on histogram of oriented gradient (HOG) and histogram of oriented optical flow (HOOF) have popularly been used for object recognition and action classification [3, 4, 11]. Dalal and Triggs [4] used a dense grid of HOG to detect humans. Chaudhry *et al*. [3] recognized 10 basic human actions including running, side walking, waving, and jumping by classifying a HOOF time-series. Training sets of the both systems [3, 4] are, however, taken from the limited viewpoints (front-and-back views or side views). Marszalek *et al*. [11] recognized 12 complex human actions from various viewpoints by the mixture of the descriptors. The labeled actions consist of "getting out of a car," "driving a car," "hand shaking," and "hugging person.". The descriptors include HOG, HOOF, SIFT, and 3-D/2-D Harris detector. The problem of their system was that the recognition rate of "getting out of car" is about 15 %, which is lower than the recognition rates of other actions. They identified scene classes and combined them with the descriptors in order to improve performance. The precision of the recognition of "getting out of car" did not improve.

3. System overview

This section provides an overview of our system, which is summarized in Fig. 2. For vehicle detection, we obtain foreground masks by using an adaptive background mixture model. Foreground blobs are extracted from the foreground masks using morphological operations. The system tracks the extracted blobs and classifies them as vehicle / non-vehicle blobs. For vehicle blobs, we extract geometrical parameters (*e.g.* position, size, orientation, and tilt angle) of vehicles by matching the vehicle blobs with 2-D projection templates of 3-D vehicle models. We further identify ROIs

of the vehicle blobs by the 3-D vehicle models with the estimated parameters. Optical flow and gradient are computed on the extracted ROIs. The system transforms the optical flow field based on the extracted vehicle pose. The fields become view-independent after transformation. Thus, HOOF and HOG of the same actions in different viewpoints can have similar distributions after transformation. We train an SVM classifier for the view-independent features to classify atomic interactions. Our system recognizes composite human interactions with the vehicles by representations of the temporal structure of the atomic interactions.

4. Vehicle localization

Accurate localization of vehicles is an important first step for the system. For the accurate vehicle localization, we first detect and track foreground blobs, then classify the blobs as vehicle / non-vehicle blobs. For vehicle blobs, we match silhouettes of them to silhouettes of synthetic 3-D vehicle models. Therefore, we can extract geometrical parameters and a type of the vehicles. We will describe each of these processes below.

4.1. Blob detection / tracking

Our system uses an adaptive Gaussian mixture model for background subtraction based on Zivkovic's work [23]. Their methodology updates parameters and the number of components of the mixture model after an efficient scene adaptation. Connected component analysis and erosion/dilation are applied to the detected foreground pixels to detect foreground blobs. The system removes noisy blobs whose size is smaller than an assigned threshold.

In order to track blobs, we extend the W4 tracking algorithm (Haritaoglu *et al*. [5]). The W4 system tracks objects for five different cases. The five cases are 'one-to-one matching,' 'objects splitting into multiple regions,' 'several objects merging into one,' 'new object appearing,' and 'object disappearing.' They computed the correspondence between foreground blobs by measuring distance of blobs with a dynamic shape model for tracking objects.

Our approach is to use SIFT feature points to measure similarity of blobs [9] as shown in Eq. 1. The similarity of two blobs ($blob_2$(*t*+1), $blob_2(t)$) are measured by the number of feature points on blob *1* in frame *t* ($blob_1(t)$) that are matched to feature points on blob *2* in frame *t+1* ($blob_2$(*t+1*)). The distance of two SIFT feature points (SF_1 and SF_2) can be computed as shown in Eq. 2.

$$\begin{aligned} &\mathit{Similarity}\,(\mathit{blob}_1(t), \mathit{blob}_2(t+1)) = \\ &\quad \mid \{(SF_i, SF_j), SF_i \in \mathit{blob}_1(t), SF_j \in \mathit{blob}_2(t+1) \\ &\qquad \mid (i,j) = \underset{(a,b)}{\operatorname{argmax}}\, \mathit{Distance}(SF_a, SF_b)\} \mid \end{aligned} \quad (1)$$

where,

$$\begin{aligned} &\mathit{Distance}\,(SF_1, SF_2) = \sum_{\widehat{\delta}} \left(SF_1(\widehat{\delta}) - SF_2(\widehat{\delta})\right) + \\ &\quad (SF_1(p) - SF_2(p)) \cdot w_p + (SF_1(S) - SF_2(S)) \cdot w_S \end{aligned} \quad (2)$$

Here, $\mid A \mid$ denotes the number of elements in a set A. $\widehat{\delta}$ is a 128 dimension vector of descriptors, p is position, and S is size. Weights w_p and w_S correspond to features p and S, respectively. The match of SIFT points is quickly found by a non-iterative greedy algorithm. Then, we make a matrix to represent the similarity of blobs in frame *t* (*blob* (*t*)) to blobs in frame *t+1* (*blob* (*t+1*)). The matrix is analyzed to track blobs for the occurrences of the five cases that the W4 system categorized.

4.2. Blob classification

The shape based matching requires expensive computation. The matching on all the blobs is not efficient for the system. The system classifies blobs as vehicle / non-vehicle blobs to exclude non-vehicle blobs from the shaped based matching. The difficulty for classifying objects is on the classification of multi class objects. By tracking blobs, the system knows whether the blobs are merged or not. Thus, the system collects separated blobs (single class object) and classifies them.

We use shape features (*e.g*. size and compactness) and histogram of SIFT feature points (bag of visual-words [18]) of the single class object blobs to train a classifier. To represent the histogram of SIFT feature points, we set up 10 bins for SIFT feature points (extracted in Section 4.1) by K-means clustering. Since SIFT feature points are scale-invariant, additional features such as size and compactness can be useful to obtain shape-based information. The size of the blob is equal to the number of pixels in the blob. The compactness is equal to the value of the square of the number of pixels on the boundary over the size. We build a 12 dimension feature vector by a mixture of shape features (2 dim.) and histogram of SIFT feature points (10 dim.). The K-nearest neighbor classifier is trained with the feature vector to classify vehicle / non-vehicle blobs.

4.3. Shape based matching of 3-D vehicle models

After a blob is classified as a vehicle, we extract its geometric parameters and type by the shape based matching of 3-D vehicle models. We build synthetic 3-D vehicle models of a sedan and an SUV (sports utility vehicle), then extract 2-D templates from the 3-D models. We adopt Song and Nevatia's approach [19]. They extracted 2-D images for 72 bins from 360° orientation and 19 bins from 90° tilt angle for optimal processing. Each 3-D vehicle model has 1368 extracted 2-D templates. Sample images of 2-D templates from our 3-D vehicle models are shown in Fig. 3.

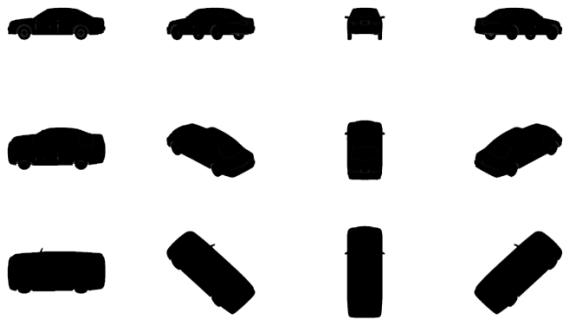

Figure 3. Extracted 2-D templates from a 3-D vehicle model (sedan)

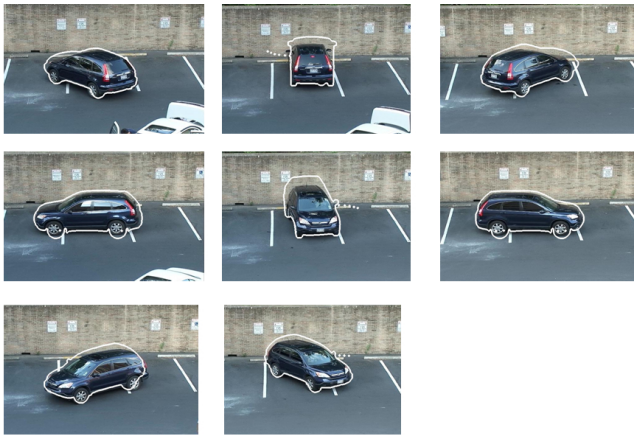

Figure 4. A vehicle from various viewpoints is detected, and its silhouettes are marked by white color lines. The silhouettes are generated by a 3-D vehicle model (SUV).

For the shape based matching, the system scales the 2-D vehicle templates to have a similar size as the foreground blob. The system calculates an area matching score and a contour matching score. The area matching score is the number of overlapped pixels of blobs and 2-D templates. The contour matching score is obtained by chamfer matching [2] on edges of blobs and contours of 2-D templates. The system calculates the final matching score by the multiplication of the two matching scores. The geometrical parameters and type of a vehicle can be extracted from the 2-D template which has the maximum value of the final matching score. The detection of an SUV from eight different orientations is shown in Fig. 4.

5. View-independent feature extraction

The extraction of appropriate features is an important step that enables the system to operate fast and robustly. We extract view-independent features after a vehicle is correctly localized. By using 3-D vehicle models, ROIs are specified. On each ROI, we extract optical flow and gradient. The optical flow field is not view-independent. To make it view-independent, the system transforms it using 3-D vehicle models. The transformation is accomplished by measuring the direction of door opening / closing that fits the optical flow field. We illustrate these processes below.

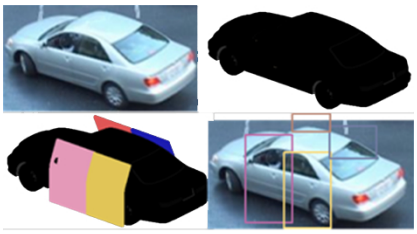

Figure 5. ROI extraction. The regions of four doors are extracted separately.

5.1. Regions-of-interest extraction

Careful specification of regions is an important step in ROI analysis. In the human-vehicle interactions of "a person getting into/out of a vehicle," a vehicle is parked so it does not change the location and orientation. Therefore, the system specifies ROIs only once in a human-vehicle interaction. People can get in or out of a vehicle through specific regions (door regions). Thus, the extraction of features on the ROIs can be enough for the recognition of interactions. By maintaining multiple ROIs, the system is also able to recognize several interactions simultaneously (*e.g.* a driver and a passenger getting out of a vehicle at the same time).

We can specify ROIs of a vehicle using 3-D vehicle models with movable doors (see Fig. 1) after accurate localization of a vehicle. The ROIs are correctly sized and located on the vehicle by the 3-D vehicle models as shown in Fig. 5.

5.2. Transformation of the optical flow field

To recognize human-vehicle interactions, it is critical to understand motion of humans. We use optical flow to detect and analyze motion. Accurate optical flow calculation is important for motion analysis. Ogale and Aloimonos [13] proposed an advanced optical flow detection algorithm and presented its implementation. We apply their implementation to extract optical flow accurately. However, relying on the raw extracted optical flows may cause problems because the optical flow field appears different as viewpoint changes. Particularly, the raw optical flow field cannot distinguish whether human opens or close a door.

We propose an approach to transform the optical flow field, so the system is able to extract view-independent features. Using the transformation, the system makes the direction of optical flow vectors extracted from the same interac-

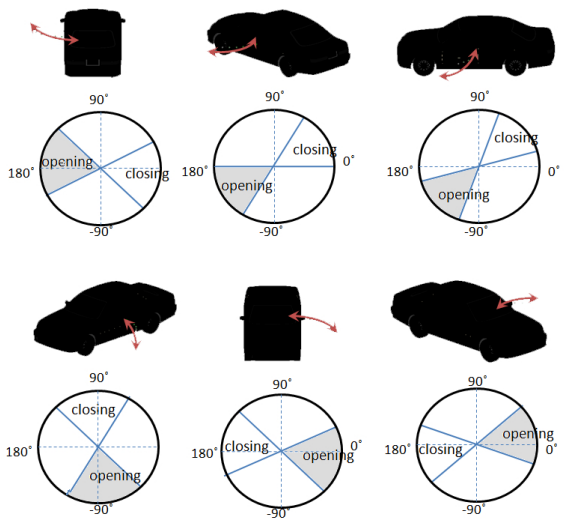

Figure 6. 2-D templates from various view points including door opening/closing direction and their graphs representing the range of direction. Optical flow is remapped by the direction of a vehicle. When a driver side door is opened (closed), optical flow vectors on the ROI are transformed so their angles are ranged from 0° to 90° (from 180° to 270°).

tion occurrences be similar, regardless of its viewpoint. The direction of optical flow is ranged from 0 to 2π. By transformation, we want to make the range of direction of optical flow on opening (or closing) door be from 0 to $\pi/2$ (or from π to $\pi \cdot 3/2$). In order to do that, we estimate the direction of door opening (or closing) using 3-D vehicle models. In the 3-D vehicle models, we draw a curved line for each door to represent the direction. As we change orientation of the vehicle models, the shape of the curved line changes also as shown in Fig. 6. We can estimate the range of direction for door opening (or closing) from the selected 2-D templates (from Section 4.3) of the 3D models. Let $[\theta_1, \theta_2]$ (or $[\theta_3, \theta_4]$) be the range of direction for door opening from an assigned viewpoint. We can now transform the direction of optical flow vectors by the following equation:

$$\Theta' = \frac{\pi}{2} \cdot \left(\frac{mod(\Theta - \theta_i,\ 2\pi)}{mod(\theta_{i+1} - \theta_i,\ 2\pi)} + (i-1) \right) \qquad (3)$$

$$if\ \{\theta_i \le \Theta \le \theta_{i+1}\}\ or\ \{\theta_{i+1} \le \theta_i\ \&\ (\Theta \le \theta_{i+1} \| \theta_i \le \Theta)\}$$
$$for\ \ i = 1,\ 2,\ 3,\ 4\ (\theta_5 = \theta_1)$$

As a result, the system obtains a set of optical flow vectors whose direction is transformed to be view-independent. Using our 3-D vehicle model, the system adaptively transforms the vectors depending on the viewpoint.

5.3. Histogram of transformed & oriented optical flow and histogram of oriented gradient

We build two histograms to reduce the dimension of features. One histogram is of the optical flow field for analyzing motion and the other histogram is of the gradient field for analyzing shape. Both transformed optical flow and raw image gradient are used to construct the histograms: motion is significantly dependent on viewpoints, while shape of humans does not.

We classify interactions (R) on an image (I). We extract parameters (geometric parameters and a type), optical flow field (*opfl*), and gradient field (*grad*). After specification of ROIs and transformation of the optical flow field, we can obtain transformed optical flow field (*T-opfl*) and gradient field on ROI. For dimensionality reduction, we build histogram of transformed & oriented optical flow (T-HOOF) and histogram of oriented gradient (HOG).

$$\begin{aligned} \mathbf{P}(R \mid I) &\approx \mathbf{P}(R \mid param,\ I) \\ &\approx \mathbf{P}(R \mid param,\ opfl(I),\ grad(I)) \\ &\approx \mathbf{P}(R \mid T\text{-}opfl(ROI),\ grad(ROI)) \\ &\approx \mathbf{P}(R \mid \texttt{T-HOOF}(ROI),\ \texttt{HOG}(ROI)) \end{aligned} \qquad (4)$$

Here, R refers to interactions and *param* is geometric parameters and a type of a vehicle. *opfl* (I) and *grad* (I) are optical flow field and gradient field on image I, respectively.

To build histogram of transformed & oriented optical flow (T-HOOF), we create 9 bins for each direction (opening, closing, and the two others) so that we have 36 bins in 360° for the histogram. Each optical flow vector is weighted by its magnitude and is smoothed by Gaussian filter. To make T-HOOF scale-invariant, each bin is divided by the area of the ROI. Examples of T-HOOF and HOOF representations are shown in Fig. 7

The second feature is HOG on ROIs. T-HOOF is a strong feature to detect motion. However, the system may not distinguish the interaction of "a person opening a door" from the interaction of "a person appearing from a vehicle." To overcome this difficulty, we calculate gradient field on pixels where the magnitude of optical flow vectors is non-zero. Because the shape of humans is more complex than the shape of doors (more edges), the magnitude of the gradient on humans is generally higher than the magnitude of gradient on doors. We use the same number of bins for HOG as the ones from the calculation of T-HOOF.

6. Recognition of interactions

Our system classifies atomic interactions on doors of the vehicle using view-independent features in each frame. Then, the system recognizes complex human-vehicle interactions of person getting into and out of a vehicle using atomic interaction classification results.

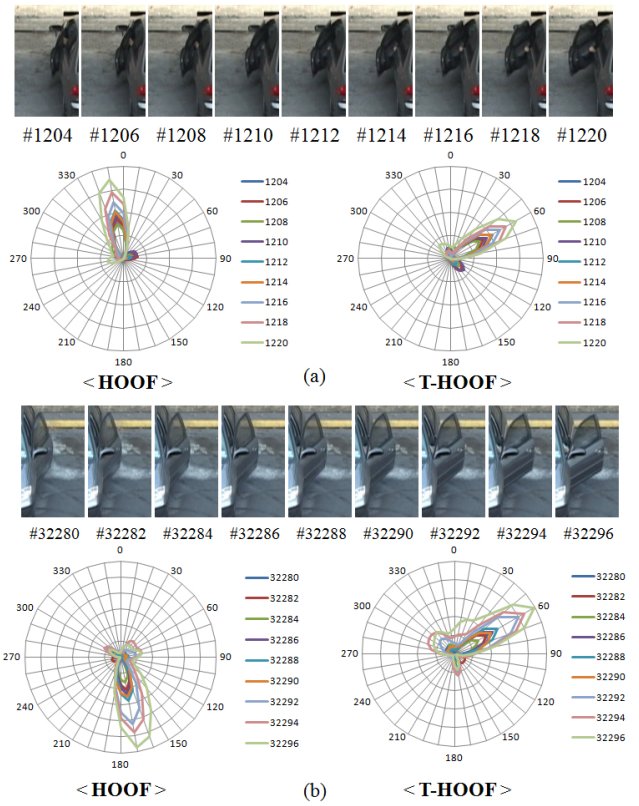

Figure 7. Representations of HOOF and T-HOOF. (a) and (b) represent the same atomic interaction, "a person opens a door," but they are taken from different viewpoints.

6.1. Atomic interaction classification

When a person gets into or out of a vehicle, the person performs several distinguishable actions. In order to get in a vehicle, a person gets close to the vehicle, opens a door, disappears into the vehicle, and closes the door. Similarly, in order to get out of a vehicle, a person opens a door, appears from the vehicle, closes the door, and gets away from the vehicle. To represent these sub-events, we define six classes of atomic interactions as follows: "person appearing into / disappearing from a vehicle," "person opening / closing a door," "person walking around a vehicle," and "no movements."

We classify these six interactions (R) on image (I) by an SVM classifier. Several researchers [3, 11] used an SVM classifier with HOOF and/or HOG features, and they showed that an SVM classifier performs well with these features. Instead of training $\mathbf{P}(param,\ opfl(I),\ grad(I) \mid R)$, we train $\mathbf{P}(\texttt{T-HOOF}(ROI),\ \texttt{HOG}(ROI) \mid R)$ after extracting T-HOOF and HOG features. We use an SVM classifier with a RBF kernel to train the features simultaneously. We classify the six classes of atomic interactions on ROIs robustly using those two features in various viewpoints. More details on classification results of these actions are presented in Section 7.

6.2. Composite interaction recognition

Once our system classifies six atomic interactions on ROIs in all frames, our system recognizes composite interactions based on the classification of atomic interactions. Temporal filtering is performed for improving initial interaction classification performance and for clustering video frames which are classified as a same interaction. Thus, we can have a series of atomic interactions which are composed of consecutive frames.

Ryoo and Aggarwal [17] proposed a general methodology for complex human activity recognition using Allen's event presentation [1]. Compared with their system, our system does not require the recognition of general human activities to solve the problem. All the interactions are represented by one interval of temporal logic, "before." The representations of interactions that a person gets into or out of a vehicle are as follows, where p denotes a person and v denotes a vehicle.

$$
\begin{aligned}
&\textbf{person\_getting\_into\_vehicle}(p,\ v) = (\\
&\quad list(\ def(w,\ \textbf{walk\_around\_vehicle}(p,\ v)),\\
&\qquad list(\ def(o,\ \textbf{open\_door}(p,\ v)),\\
&\qquad\quad list(\ def(d,\ \textbf{disappear\_into\_vehicle}(p,\ v)),\\
&\qquad\qquad list(\ def(c,\ \textbf{close\_door}(p,\ v))))))),\\
&\quad and(\ and(before(w,\ o),\ before(o,\ d)),\\
&\qquad before(d,\ c)\ )\\
&);\\
&\textbf{person\_getting\_out\_of\_vehicle}(p,\ v) = (\\
&\quad list(\ def(o,\ \textbf{open\_door}(p,\ v)),\\
&\qquad list(\ def(a,\ \textbf{appear\_from\_vehicle}(p,\ v)),\\
&\qquad\quad list(\ def(c,\ \textbf{close\_door}(p,\ v)),\\
&\qquad\qquad list(\ def(w,\ \textbf{walk\_around\_vehicle}(p,\ v))))))),\\
&\quad and(\ and(before(o,\ a),\ before(a,\ c)),\\
&\qquad before(c,\ w)\ )\\
&);
\end{aligned}
$$

Our system recognizes the activity if all its sub-events are recognized. To reduce the false negatives of activity recognition, our system allows missing or wrong presentation of one interval temporal logic of three interval temporal logic. The idea of allowing missing or wrong presentation has been suggested by Pinhanez and Bobick as well [15].

7. Experimental results

We test the implementation of the system to recognize composite interactions such as "a person getting into a vehicle" and "a person getting out of a vehicle." We generated two video datasets for our experiments. Each dataset

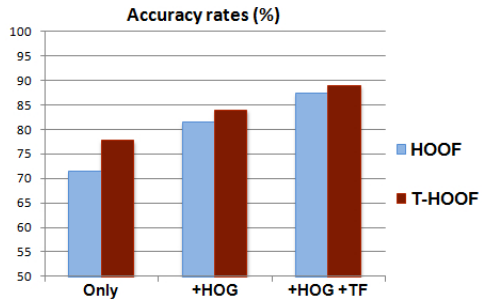

Figure 8. Overall accuracy rates for the classification of atomic interactions to compare T-HOOF with HOOF. 'only,' '+HOG,' and '+HOG +TF' denote that HOOF/T-HOOF is used *without additional features or processing, with HOG feature,* and *with HOG feature followed by temporal filtering*, respectively.

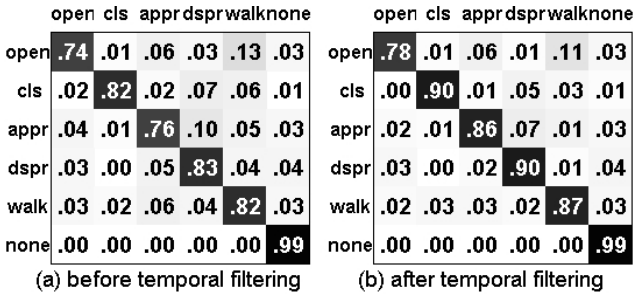

Table 1. Atomic interaction classification results (a) before temporal filtering and (b) after temporal filtering. 'open,' 'cls,' 'appr,' 'dspr,' 'walk,' and 'none' denote "opening a door," "closing a door," "appearing from a vehicle," "disappearing into a vehicle," "walking around a vehicle," and "no movements," respectively. The numbers of frames processed per interaction is 1122, 682, 1148, 1043, 1209, and 1673, respectively.

includes four executions of two composite interactions performed by a driver from eight different views. Thus, each dataset has 64 composite interactions. We use 32 interactions for training and the other 96 interactions for testing. Vehicles used in the dataset are a sedan and an SUV. The videos were taken in 12.5 frames per second in the resolution of 720 * 480.

Table 1 shows the confusion matrix of the classification rates of atomic interactions with a sedan. The recognition rates of classification are 74 % at the worst and 84 % at the average before temporal filtering, and 78 % at the worst and 89 % at the average after temporal filtering. We also present accuracy rates for the classification of atomic interactions to compare T-HOOF with HOOF as shown in Fig. 8. T-HOOF performed superior to HOOF in all provided conditions. The performance of HOOF and T-HOOF is improved by adding a feature, HOG and by processing temporal filtering.

Table 2 shows the interaction recognition results on the dataset. We recognize two complex interactions: "a person getting out of a vehicle" and "a person getting into a vehicle." The representations of the interactions are presented in Section 6.2. 83 of 96 composite interactions are recognized correctly, while 13 interactions are not detected and 1 interaction is detected erroneously. Because of the accurate representation on composite actions, the system is superior to the previous systems. The results are, moreover, obtained from consecutive sequences of interactions, and humans and vehicles are not manually segmented. The system recognizes sequences of composite human-vehicle interactions with a high degree of accuracy.

Fig. 9 shows example sequences of human-vehicle interactions which our system recognized correctly. Four human-vehicle interactions are presented horizontally on each row. Interactions on the first and third row are "person getting out of a vehicle," and interactions on the second and fourth row are "person getting into a vehicle."

Dataset	Sequence	True positive	False positive	False negative
Dataset 1 (Sedan)	getting out	21	0	3
	getting in	23	0	1
Dataset 2 (SUV)	getting out	20	1	4
	getting in	19	0	5
Total	96	83	1	13

Table 2. Composite interaction recognition results

8. Conclusions

We have recognized complex human-vehicle interactions with vehicles with a high degree of accuracy. The proposed methodology classifies atomic interactions from various viewpoints and improves the recognition rates of composite interactions. The contributions of our work are: the extraction of view-independent features using 3-D vehicle models and the recognition of "getting into or out of a vehicle" interactions. We showed that our approach is superior to the previous approaches. The methodology benefits from decreasing the requirement of training data from various viewpoints.

Acknowledgments

This material is based upon work supported partly by Texas Higher Education Coordinating Board award #003658-0140-2007, and partly by the Defense Advanced Research Projects Agency (DARPA) under Contract No. HR0011-08-C-0135. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of DARPA.

Figure 9. An example sequences of human-vehicle interactions. 'open,' 'cls,' 'appr,' 'dspr,' 'walk,' and 'none' are defined in Table. 1. All clips are cropped from input frames by the specified ROIs. They are arranged by time (left to right).

References

[1] J. F. Allen and G. Ferguson. Actions and events in interval temporal logic. *Journal of Logic and Computation*, 4(5):531–579, 1994.

[2] H. G. Barrow, J. M. Tenenbaum, R. C. Bolles, and H. C. Wolf. Parametric correspondence and chamfer matching: Two new techniques for image matching. In *IJCAI*, 1977.

[3] R. Chaudhry, A. Ravichandran, G. Hager, and R. Vidal. Histograms of oriented optical flow and binet-cauchy kernels on nonlinear dynamical systems for the recognition of human actions. In *CVPR*, 2009.

[4] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005.

[5] I. Haritaoglu, D. Harwood, and L. S. Davis. W4: real-time surveillance of people and their activities. *PAMI*, 22(8):809–830, Aug 2000.

[6] S. W. Joo and R. Chellappa. Attribute grammar-based event recognition and anomaly detection. In *CVPRW*, 2006.

[7] G. Jun, J. K. Aggarwal, and M. Gokmen. Tracking and segmentation of highway vehicles in cluttered and crowded scenes. In *WACV*, 2006.

[8] J. T. Lee, M. S. Ryoo, M. Riley, and J. K. Aggarwal. Real-time illegal parking detection in outdoor environments using 1-D transformation. *IEEE T-CSVT*, 19(7):1014–1024, July 2009.

[9] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.

[10] F. Lv and R. Nevatia. Single view human action recognition using key pose matching and viterbi path searching. In *CVPR*, 2007.

[11] M. Marszalek, I. Laptev, and C. Schmid. Actions in context. In *CVPR*, 2009.

[12] D. J. Moore, I. A. Essa, and M. Hayes. Exploiting human actions and object context for recognition tasks. In *ICCV*, 1999.

[13] A. S. Ogale and Y. Aloimonos. A roadmap to the integration of early visual modules. *IJCV: Special Issue on Early Cognitive Vision*, 72(1):9–25, 2007.

[14] N. M. Oliver, B. Rosario, and A. P. Pentland. A bayesian computer vision system for modeling human interactions. *PAMI*, 22(8):831–843, 2000.

[15] C. Pinhanez and A. Bobick. Human action detection using pnf propagation of temporal constraints. In *CVPR*, 1997.

[16] M. S. Ryoo and J. K. Aggarwal. Recognition of high-level group activities based on activities of individual members. In *WMVC*, 2008.

[17] M. S. Ryoo and J. K. Aggarwal. Semantic representation and recognition of continued and recursive human activities. *IJCV*, 82(1):1–24, 2009.

[18] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *ICCV*, 2003.

[19] X. Song and R. Nevatia. A model-based vehicle segmentation method for tracking. In *ICCV*, 2005.

[20] Z. Sun, G. Bebis, and R. Miller. On-road vehicle detection: a review. *PAMI*, 28(5):694–711, 2006.

[21] B. Tamersoy and J. K. Aggarwal. Robust vehicle detection for tracking in highway surveillance videos using unsupervised learning. In *AVSS*, 2009.

[22] P. Turaga, R. Chellappa, V. S. Subrahmanian, and O. Udrea. Machine recognition of human activities: A survey. *IEEE T-CSVT*, 18(11):1473–1488, Nov. 2008.

[23] Z. Zivkovic. Improved adaptive gaussian mixture model for background subtraction. In *ICPR*, 2004.