Final Project for 6.881

Interaction Detection in Far-Field Visual Surveillance

Xiaogang Wang and Chaowei Niu

Abstract

In this paper, we propose a framework to detect interactions between two objects in far-field visual surveillance. A set of simple interactions are first defined using trajectory analysis, which allows the description of long-range interactions based on the knowledge on the history of object movements. Complicated interactions are composed of simple interactions using the interval temporal logic. Two approaches are proposed to detect abnormal and intentional interactions. In the first approach, trajectories with attributes are extracted as features to describe interactions. Using a novel trajectory similarity measure, the similarity between two interactions is defined. Abnormal interactions are detected as outlier samples of the interaction set. In the second approach, the activity of each object is modeled as a Hidden Markov Model (HMM). We detect intentional interactions, by computing the mutual information between the two HMMs.

1. Introduction

In far-field visual surveillance, people often have interests in the automatic interpretation of object interactions, especially some abnormal interactions. In far-field visual surveillance, because of the low resolution of the acquired imagery, there is no enough information on the internal configuration of objects and the interaction description is mainly based on the object movement. Consider two scenarios as examples: (A) "A car was intentionally following another car"; (B) "A car dropped off a person. The person didn't enter the building, but entered another car." Usually, there are two approaches to detect interesting interactions. In the first approach, all kinds of interactions are clustered and abnormal interactions are detected as outlier samples [1][8]. The disadvantage is that it cannot give semantic interpretation to the detected abnormal interactions. The second approach builds a model for each kind of interaction to be detected [3][4][5]. It has a semantic interpretation to the detected interactions. However, it is not a general framework and can only be applied to specific applications. Furthermore, people are often interested in unexpected abnormal interactions. It is difficult to collect their training samples, or to give the explicit logical description.

We propose a framework with two levels of interaction description. In the level of general description, a set of basic interaction types using trajectory analysis and interval temporal logic [6] has been defined. They cover most of the interactions in far-field visual surveillance. In a more detailed description, for each type of interactions, we further compare the similarity between interactions, and detect abnormal or intentional interactions. For example, in scenarios (A) and (B) mentioned above, we first recognize the interactions as *follow* and *drop-off*, and then detect them as intentional *follow* and abnormal *drop-off*.

Most of the previous approaches [3][7] define object interactions in far-field visual surveillance based on the velocity of objects and relative distance between objects. However, these properties are not enough for the detection of some long-range interactions, which need the knowledge on the history of object movement. For example, the *follow* interaction can be detected based on the positions and moving directions of objects in Figure 1(a). However, in Figure 1(b), in order to decide whether one car is following the other, their movement history has to be considered. The later case is often of interest. If one car is intentionally following another car, it will try to keep some distance to its aim avoiding to be noticed. We define this kind of long-range interactions by analyzing the trajectories of the two objects, which encode the historical information of object movement.

(a) (b)

Figure 1. Examples of the *follow* interactions. The *follow* in (a) can be detected based on the moving directions of the two cars. In order to detect *follow* in (b), history of object movement has to be considered.

We develop two approaches to detect abnormal and intentional interactions for any type of interactions. In the first approach, we define the similarity between interactions and detect abnormal interactions as outlier samples. Since interactions have been classified into different types, we can extract proper features for each type of interactions differently instead of using some uniform features as in [1]. We extract trajectories with attributes as features to represent interactions. Using a novel trajectory similarity measure, the interaction similarity is computed. For example, most of the *follow* interactions between cars happen just because they are on the same road. They have no intention at all. However, we may suspect an intentional *follow*, if one car still follows another car even after it has changed several roads. The trajectory of this *follow* is very different with those of normal *follow*. As another example, we represent *drop-off* as the trajectory of the car before stop and the trajectory of the people after getting out of the car. In scenario (B), since most of the people enter the building after getting out of the car, while this person entered another car, his trajectory is different from others.

In the second approach, we model the activity of each object as a Hidden Markov Model (HMM), and compute the mutual information between two HMMs. If the interaction between two objects is intentional, the hidden state of one object will affect the state transition of the other object, and the mutual information between two HMMs is high. Consider the *follow* example again. If one object changes its speed according to the speed of the other object during the *follow* interaction, this is an intentional *follow*.

The contribution of this paper can be summarized as fourfold. First, a two-level interaction detection framework is proposed. Second, we detect the long-range interactions by trajectory analysis. Third, we extract a proper set of trajectories as features to represent each type of interactions. Using a novel trajectory similarity measure, the similarity between interactions is defined. Abnormal interactions are detected as outlier samples by comparing interactions. Fourth, the intentional interaction is detected by computing the mutual information between HMMs. The paper is organized as following. Some related work is reviewed in Section 2. In Section 3, a set of simple interactions are defined and they are extended to complicated interactions using interval temporal logic in Section 4. Abnormal and intentional interactions detection by computing the interaction similarity and mutual information between HMMs are described in Section 5 and 6. Section 7 is conclusion and discussion.

2. Related Work

One possible approach to learn the activity is to cluster all kinds of interactions and then abnormal interactions can be detected as outlier samples [1][8][10].

A. R. Chowdhury and R. Chellappa [8] have represented the activity by the deformations of the point configuration in a shape space. Basis shapes have been learned for each activity based on the 2-D trajectories, and then unknown activity can be projected onto these basis shapes to be recognized as a normal activity or an abnormal activity.

H. Zhong and J. Shi [1] divided the video into equal length segments and used motion feature to extract prototypes, then computed the prototype-segment co-occurrence matrix and found the correspondence relationship between prototypes and segments through co-embedding. Then abnormal activities are defined as the video segments that have correspondences to distinctive important features.

F. Porikli and T. Haga [10] used time-wise and object-wise features to apply principal component analysis on the feature-wise affinity matrices to obtain object clusters and then to detect abnormal activities. As mentioned before, these kinds of methods cannot give semantic interpretation to the detected abnormal activities.

Another possible method is to extract the trajectories followed by a supervised learning. W. Grimson et al [9] estimated a hierarchy of similar distributions of activities based on the co-occurrence feature clustering. Starner [11] used a Hidden Markov Model (HMM) to represent a simple event and recognize this event by computing the probability that the model produce the visual observation sequence. Town [12] proposed to use a Bayesian Networks for event recognition. Kohler[7] and Hongeng [3] detect interactions based on the velocity and relative distance between objects. As mentioned in Introduction, they cannot detect the “follow” interaction in Figure 1(b). Fernyhough et al [13] constructed qualitative event models to detect *follow* and *overtake*. They could detect interactions only when two objects are close in space and required to learn the path regions from long-term observation in advance. Weghe et al [14] tried to detect the long-range interactions using “Qualitative Trajectory Calculus along a road Network” (QTCN). However, they also needed a road map which is often unavailable in visual surveillance. Since both of the two approaches relied on the knowledge of scene structure, they will fail if objects goes out of roads or regular paths.

Our method is also a supervised learning method. However, we use a two-level framework for interaction description which allows to define the long range interaction by trajectory analysis, then abnormal interactions can be detected by comparing interactions using the proposed trajectory similarity measure. In this paper, we use the Stauffer-Grimson tracker [15] to detect and track moving objects in the scene. Scene clutter is filtered in a preprocessing step.

3. Simple Interaction

Neumann [16] developed a list of motion verbs to provide natural language description of activities. We select a small subset of terms to define the basic types of interactions between two objects in far-field visual surveillance as shown in Table 1. This set can be further extended in practical applications. However, in this paper, we just use it as example to explain how this framework works. In this section, we first define simple interactions and extend them to complicated interactions using interval temporal logic in the next section.

Table 1. Interactions between two objects.

	Independent of history of movement	Depend on history of movement
Simple interactions	approach meet leave pass	follow followed-by follow-approach go-alongside
Complicated interactions	return drop-off pick-up park-leave drive-leave lurk	catch-up over-take cross

As mentioned in the Introduction, some interactions can be defined only based on the velocities, positions, and relative distance of objects. For example, if we use $d(A,B,t)$ to represent the relative distance between two objects *A* and *B* at time *t*, the *approach* interaction is defined as,

$$approach(A,B,t) \Rightarrow \partial d(A,B,t)/\partial t < 0\,. \tag{1}$$

Since there have been a lot of work [3][7] on this kind of interaction definitions, we will not discuss much about it in this paper. We will focus on the long-range interactions, whose definition needs the knowledge on the history of object movement. We solve this problem by analyzing object trajectories which encode the information of movement history.

3.1. Define interaction by trajectory analysis

The trajectory of an object *A* is represented by a sequence of observations, $traj(A) = \{\vec{a}_s \,|\, s = 1, \ldots, N\}$. *N* is the number of observations sampled along the trajectory. $\vec{a}_s = < x_s^a, y_s^a, \theta_s^a, v_s^a, t_s^a, \ldots >$ is the feature vector of the observation, including the spatial coordinates (x_s^a, y_s^a), moving direction θ_s^a, speed v_s^a, and time record t_s^a. More features can be added, such as object's appearance in the image, depending on different applications.

First consider the *follow* interaction. We define "*A* follows *B*" as: *A* passes through the same position where *B* passed some time before, and *A* has the same moving direction as *B* on that position. This definition can be easily formulated by analyzing the trajectories of *A* and *B*. As shown in Figure 2, for an observation $\vec{a}_s$ on trajectory *A*, we find its nearest observation $\vec{b}_{\psi(s)}$ in space on trajectory *B*, where

$$\psi(s) = \arg\min_{j \in B} \left\| \left(x_s^a - x_j^b, y_s^a - y_j^b \right) \right\|. \tag{2}$$

The distance between $\vec{a}_s$ and $\vec{b}_{\psi(s)}$ is

$$d_s^{(a,b)} = \left\| \left(x_s^a - x_{\psi(s)}^b, y_s^a - y_{\psi(s)}^b \right) \right\|. \tag{3}$$

The time delay between the two objects when passing through the same position is computed as,

$$\Delta T_s^{(a,b)} = t_a^s - t_{\psi(s)}^b\,. \tag{4}$$

The *follow* interaction is formulated as,

$$follow(A,B,s) \Rightarrow \left(d_s^{(a,b)} < \eta\right) \wedge \left(\left|\theta_s^a - \theta_{\psi(s)}^b\right| < \alpha\right) \wedge \Delta T_s^{(a,b)} > 0\,, \tag{5}$$

where η and α are the threshold parameters.

Followed-by, *go-alongside* and *follow-approach* are defined in the similar way,

$$followed-by(A,B,s) \Rightarrow \left(d_s^{(a,b)} < \eta\right) \wedge \left(\left|\theta_s^a - \theta_{\psi(s)}^b\right| < \alpha\right) \wedge \Delta T_s^{(a,b)} < 0\,, \tag{6}$$

$$go-alongside(A,B,s) \Rightarrow \left(d_s^{(a,b)} < \eta\right) \wedge \left(\left|\theta_s^a - \theta_{\psi(s)}^b\right| < \alpha\right) \wedge \Delta T_s^{(a,b)} = 0\,. \tag{7}$$

$$follow-approach(A,B,s) \Rightarrow follow(A,B,s-1) \wedge follow(A,B,s) \wedge \left(\Delta T_s^{(a,b)} < \Delta T_{s-1}^{(a,b)}\right). \tag{8}$$

In summary, two objects have interaction when their trajectories are close enough, and the interaction type is recognized by comparing the time delay when they pass through the same position. In Figure 3, object *A* is following object *B* at a higher speed on the road. However, *A* and *B* are moving in opposite direction and their spatial distance is increasing at that moment. Only using spatial distance and velocity, *follow* and *follow-*

approach cannot be detected. They can be detected from the fact that the time delay is positive and decreasing. The scene structure information, which is implicitly encoded in the trajectories of object movement, is utilized in our interaction recognition. However, our method does not require the map of the scene.

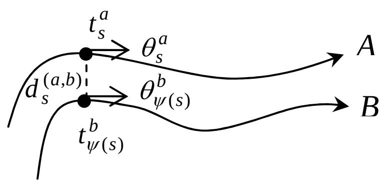


Figure 2. Trajectory analysis in interaction detection. For an observation $\bar{a}_s$ on trajectory *A*, find its nearest observation $\vec{b}_{\psi(s)}$ in space on trajectory *B*. Interaction is detected by comparing the spatial distance $d_s^{(a,b)}$ between the two observations, their moving directions (θ_s^a and $\theta_{\psi(s)}^b$) and the time records (θ_s^a and $\theta_{\psi(s)}^b$)

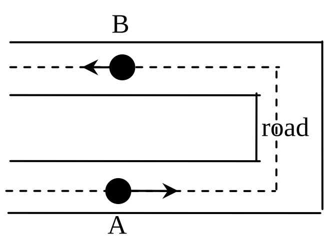


Figure 3. Object *A* is following object *B* at a higher speed on the road, but they are moving in opposite directions and their spatial distance is increasing. The detection of this *follow* interaction needs the knowledge on the road structure, which is implicitly encoded in the trajectories of the two objects.

3.2. Experiment

Figure 4 (a)-(e) are some examples of the detected *follow* interactions in two scenes. Object *A*, marked by the red window, is following object *B* marked by the cyan window. The trajectories of *A* and *B* are marked by red and cyan color. The blue color indicates the part of trajectory *A* where *follow* happens. Figure 4 (a)(b) are two normal short-range *follow* interactions, which can also be detected by the methods in [3][7]. Figure 4(c)(d) show two long-range *follow* interactions, in which the two objects are in large distance and move in different directions. They cannot be detected only using the velocities and positions of the objects. In Figure 4(c), *B* drops a bag on the ground and keeps on walking. *A* follows *B* for a while, picks up the bag, and proceeds in a different direction. In this example, *follow* only happens on some part of the trajectory. In Figure 4(d), both of the two cars come from another road, and make a turn when coming to the main road in the scene. In Figure 4(e)(f), we plot the trajectories of *follow* interactions happening in two video sequences of the two scenes. In Scene 1, most of the *follow* interactions happen when cars make u-turn to enter the parking lot, people cross the parking lot, and pedestrians walk on the path on the top right of the scene. In Scene 2, *follow* interactions happens when cars drive on the road.

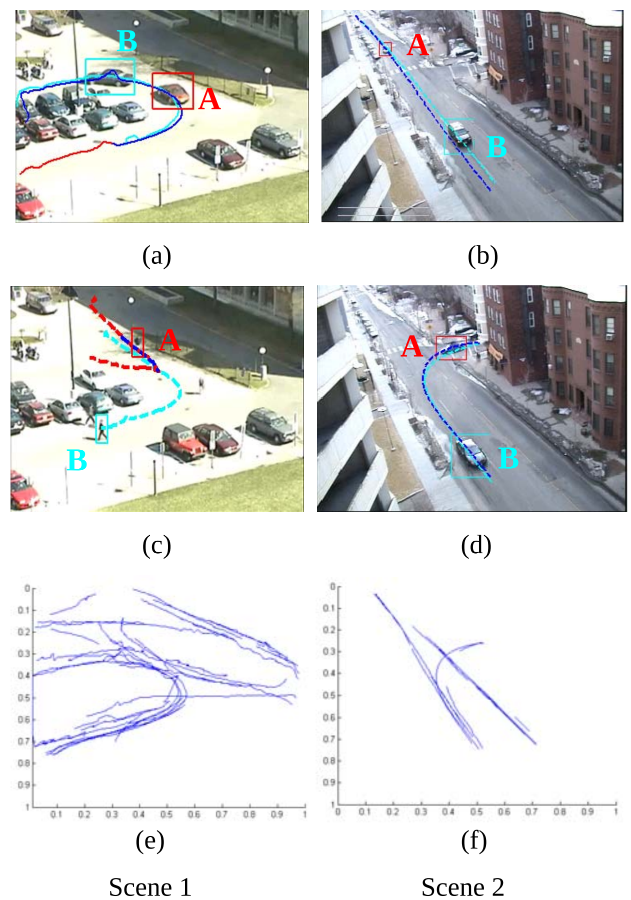


Figure 4. Experiment of detecting follow interactions in two scenes. (a)-(d) are examples of the detected follow interactions in the two scenes. Object *A*, marked by the red window, is following object *B* marked by the cyan window. The trajectories of *A* and *B* are marked by red and cyan color. The blue color indicates the part of trajectory *A* where follow happens. (e) and (f) plot the trajectories of *follow* interactions happening in two video sequences of the two scenes.

4. Complicated Interactions

4.1. Interval temporal logic

In [6], Allen introduces 13 relations between two temporal intervals *I* and *J*. Seven of them are shown in Figure 5. The remaining six converse relations, *After*,

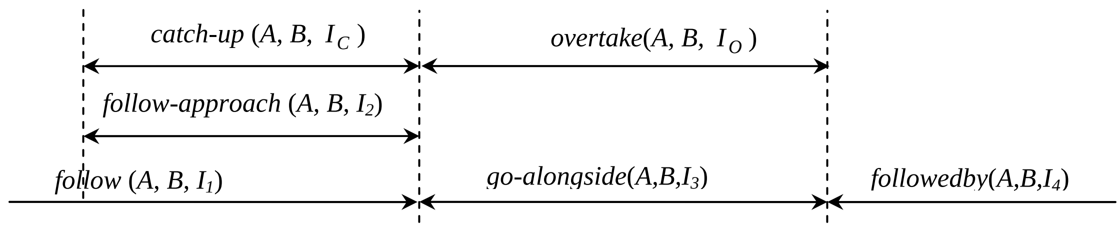


Figure 6. Representations of *catch-up* and *overtake* using interval temporal logic

Metby, *OverlappedBy*, *StartedBy*, *Contains* and *FinishedBy*, are given by exchanging *I* and *J*.

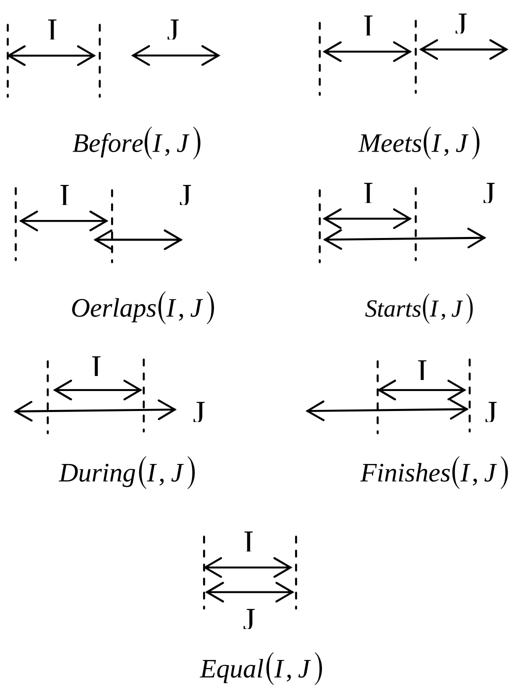


Figure 5. Temporal relations for intervals of time.

4.2. Represent complicated interactions

We represent complicated interactions from simple interactions using the interval temporal logic. Figure 5 shows the interval temporal logic of *catch-up* and *overtake*. When object *A* catches up object *B* in time interval I_C, *A* first follows and approaches *B*, and finally goes alongside with *B*. *Overtake* is described as *A* first follows *B*, then goes alongside with *B*, and is finally followed by *B*. One advantage of using interval temporal logic is that the time interval when *catch-up* or *overtake* happen can be clearly decided. *Catch-up* and *overtake* can be written as following.

$$catch-up(A,B,I_C) \Rightarrow \exists I_1, I_2, I_3 . follow(A,B,I_1) \wedge$$
$$follow-approach(A,B,I_2) \wedge go-alongside(A,B,I_3) \wedge$$

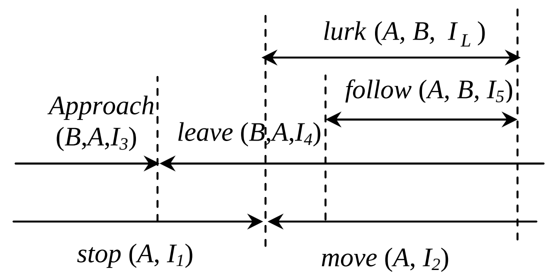


Figure 7. Representation of *lurk* using interval temporal logic

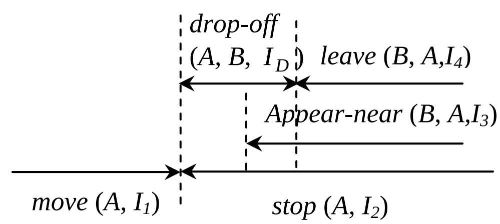


Figure 8. Representations of drop-off using interval temporal logic

$$Equal(I_C, I_2) \wedge Fihishes(I_C, I_1) \wedge Meets(I_C, I_3) \,. \qquad (9)$$

$$overtake(A,B,I_O) \Rightarrow \exists I_1, I_2, I_4 . follow(A,B,I_1) \wedge$$
$$go-alongside(A,B,I_3) \wedge \ followed-by(A,B,I_4) \wedge$$
$$Equal(I_O, I_4) \wedge Meets(I_1, I_O) \wedge Meets(I_O, I_4) \,. \qquad (10)$$

We use *lurk* and *drop-off* as more examples. Their interval temporal logic representations are shown in Figure 7 and 8. In the *lurk* interaction, object *A* first stops and object *B* is approaching *A*. When *B* passes and leaves *A*, *A* begins to move. After some time, *A* follows *B*. *Drop-off* is described as: *A* first moves, and then stops; after some time, a new object *B* appear near *A* and leaves *A*.

4.3. Experiments

Some experimental results of *catch-up*, *overtake*, *lurk* and *drop-off* detection are shown in Figure 9, 10, 11. We call the object marked by the red window *A*, and the object marked by the blue window *B*. In Figure 9, *A* follows *B* in frame 14901, and is following and approaching B in frame 15002. A catches up and goes alongside with B in frame 15064. In frame 15276, A has overtaken B, and is followed by B. Figure 10 is an example of *lurk* detection. In frame 3749, A stops, while B is approaching A. In frame 3782, when B comes close to A and passes by A, A still stops there. When B has left A some distance, A begins to move in frame 3823. After some time, A begins to follow B in frame 3876. Figure 11 shows an example of *drop-off* detection. In frame 1317536, A is moving towards the parking lot, and stops in frame 1317600. B appears and leaves A in frame 1317833 and 1318016.

Frame 14901 Frame 15002

Frame 15064 Frame 15276

Figure 9. Example of *catch-up* and *overtake* detection.

Frame 3749 Frame 3782

Frame 3823 Frame 3876

Figure 10. Example of *lurk* detection.

Frame 1317536 Frame 1317600

Frame 1317833 Frame 1318016

Figure 11. Example of *drop-off* detection.

5. Abnormal Interaction Detection Based on Trajectory Analysis

In far-field visual surveillance, same interaction pattern repeats every day every minute. For example, in a restrict parking lot, at the morning rush hour, people following each other drive into the parking lot, park, leave the car and finally disappear at some building's entrance. For this specific scenario, a lot of *follow* and *drop-off* happens. Among those usual interactions, we are more interested in the abnormal interactions. Given the above scenario, one possible abnormal interaction would be: After people park their cars, they are expected to go to the building entrance and then disappear. So if one person gets out his car, instead of approaching to the building, he walks to another vehicle, gets in and drives away. This kind of activity will arouse our attention, and need to be reported for further examinations.

5.1 Trajectory Analysis

What are "Abnormal Interactions"? What differ them from "Usual Interactions"? "Abnormal Interactions" are rare, difficult to describe, hard to predict and can be subtle [1]. However, given a large number of observations, it is still possible to detect the abnormal interactions. The trajectories of abnormal interactions must be somehow different from the normal interactions, which suggest detecting the abnormal interactions by modeling the interactions based on the trajectories.

As in Part 3.1 "A *follows* B" has been defined as:A passes through the same position where B passed some time before, and A has the same moving direction as B on that position. Based on the nature of *follow*, it can be characterized into two categories: unintentional *follow* (i.e. Normal interaction) and intentional *follow* (i.e. abnormal interaction). Unintentional *follow*, which happens a lot, is just that people or vehicle passes the same location in the order due to the nature of traffic. The follower only "follows" the followee for a short length, and then divides for its own destination. On the other hand, intentional *follow* is that the follower will follow the followee all the way down, which interests us and need to be detected. As shown in Fig. 12. There are three trajectories, A (in blue), B (in green), C (in red) respectively. It is already been detected that "B follows A" and "C follows A". Apparently "B follows A" is more intentional than "C follows A".

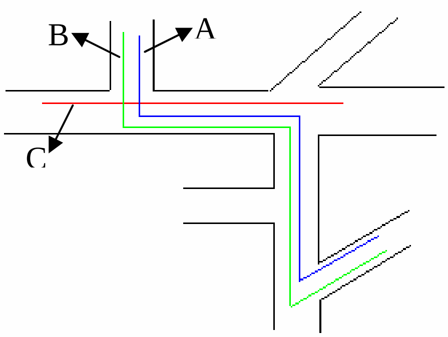


Figure 12 Examples of *follow*

In order to distinguish the abnormal interactions from normal interactions, we must define a similarity criterion which measures the degree of similarity between any two trajectories.

As described in Part 3.1, the trajectory of an object *A* is represented by a sequence of observations, $traj(A)=\{\vec{a}_s \mid s=1,\ldots,N\}$. N is the number of observations sampled along the trajectory. $\vec{a}_s =< x_s^a, y_s^a, \theta_s^a, v_s^a, t_s^a, \ldots >$ is the feature vector of the observation, including the spatial coordinates (x_s^a, y_s^a). For two trajectories $traj(A)=\{\vec{a}_s\}$ and $traj(B)=\{\vec{b}_s\}$, we define their spatial distance, similar to the modified Hausdorff distance [17]. For observation $\vec{a}_s$ on A, its nearest observation in space on B is:

$$\psi(s)=\arg\min_{j\in B}\left\|\left(x_s^a - x_j^b, y_s^a - y_j^b\right)\right\|.$$

The directed spatial distance of trajectories is

$$h(A,B)=\frac{1}{N_A}\sum_{\vec{a}_s\in A}\left\|\left(x_s^a - x_{\psi(s)}^b, y_s^a - y_{\psi(s)}^b\right)\right\|. \qquad (11)$$

The symmetric distance which is the similarity between A and B is defined as:

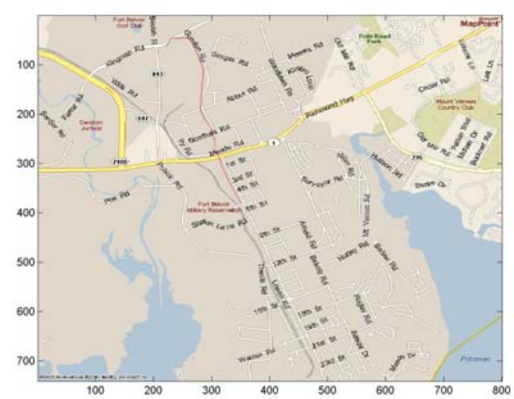
$$H(A,B)=\max\left(h(A,B),h(B,A)\right) \qquad (12)$$

Now we can detect abnormal interactions as the outliers [1] [8] [10]. In specific, given an unknown activity, first based on the criterions, which are described in Part3 and Part 4, to check what kind of interaction it is, and then use the similarity measurement to check if it is a normal interaction or an abnormal interaction. The advantage of this approach is that for an unknown interaction, we can tell if it is an abnormal interaction, as well as the semantic interpretation to it.

5.2 Simulated Experiment

We tested this algorithm on a simulated network. This simulator synthetically generates the traffic flow in a set of city streets, allowing for stop signs, traffic lights, and differences in traffic volume (i.e. morning rush hours and

(a) Detected *Follows*, the width of the trajectories is proportional to the frequency of the detected *follow* at that location.



(b) The detected most intentional *follow* trajectories (in red).

Figure 13 Intentional *follow* detection results.

afternoon rush hours have a higher volume). The network includes 110 cameras which are located at roads intersections (including cross and T intersections). For each camera, there are two observers that look in the opposite directions of the traffic flow (i.e. Observer 1 and

2 belong to camera 1, Observer 3 and 4 belong to camera 2, etc). Tracking data has been simulated 24 hours every day for 7 days.

We have used 2 hour data (8am to 9am) every day for 7 days to detect the intentional *follow*. The criterion to detect *follow* is as same as described as Equ. (5). The length of *follow* is defined by the number of observers the follower and followee have passed by. For Fig.13 (a), the threshold for the length of *follow* is 4. The width of the plotted trajectories is proportional to the frequency of the detected *follows* at that location. From Fig.13 (a), we can see that there are a lot of *follows* have been detected, especially, in the heavy traffic area (i.e. at the lower part of the map). Then, the outlier samples of those *follow*s are detected, as shown in (b), the most intentional *follow* trajectories are actually not in the heavy "*follow*" zone.

6. Abnormal Interaction Detection Using Mutual Information of HMMs

6.1. Mutual information of HMMs

Another approach to detect intentional interactions is to model the activities of two objects as HMMs and compute the mutual information between HMMs. If the two objects have intentional interaction, the state of one object will affect the state change of the other object, the mutual information between the two HMMs should be high, otherwise the two objects will change their states independently and their mutual information is low.

We model the joint probability distribution of two objects *A*, *B* as coupled HMM $p_{CHMM}(O^A, O^B \mid \theta_C)$, where O^A and O^B are two observation sequences of *A* and *B*, and θ_C is the parameters of the model. A graphical model representation of coupled HMM is shown in Figure 13. There are two chains in coupled HMM. The conditional probability of the hidden state in time *t* is decided by the hidden states on both of the two chains at the previous time step *t-1*. The two data streams can also be modeled as two independent HMMs, with probability distribution $p_{HMM}(O^A \mid \theta_A)$ and $p_{HMM}(O^B \mid \theta_B)$, where θ_A and θ_B are the parameters of HMM models for *A* and *B*. We estimate the mutual information between the two HMMs as

$$I(A,B) = \int P_{CHMM}\left(O^A, O^B \mid \theta_C\right)\ldots$$

$$\log \frac{P_{CHMM}\left(O^A, O^B \mid \theta_C\right)}{P_{HMM}\left(O^A \mid \theta_A\right) P_{HMM}\left(O^B \mid \theta_B\right)} dO^A dO^B \qquad (13)$$

One can randomly and independently generate a set of sequences (O_1^A, O_1^B), …, (O_M^A, O_M^B) based on the distribution of $P_{CHMM}\left(O^A, O^B \mid \theta_C\right)$, and estimate $I(A,B)$ as

$$I(A,B) \approx \frac{1}{M} \sum_{i=1}^{M} \log \frac{P_{CHMM}(O_i^A, O_i^B \mid \theta_C)}{P_{HMM}(O_i^A \mid \theta_A) P_{HMM}(O_i^B \mid \theta_B)} \quad (14)$$

However, it is time consuming. From two observed sequences O^A and O^B, we learn parameters using the maximum likelihood,

$$\hat{\theta}_A = \arg\max_{\theta_A}(P_{HMM}(O_A \mid \theta_A)),$$

$$\hat{\theta}_B = \arg\max_{\theta_B}(P_{HMM}(O_A \mid \theta_B)),$$

$$\hat{\theta}_C = \arg\max_{\theta_C}(P_{CHMM}(O_A, O_B \mid \theta_C)),$$

θ_B, and θ_C, the directly estimate $I(A,B)$ by comparing the maximum likelihoods,

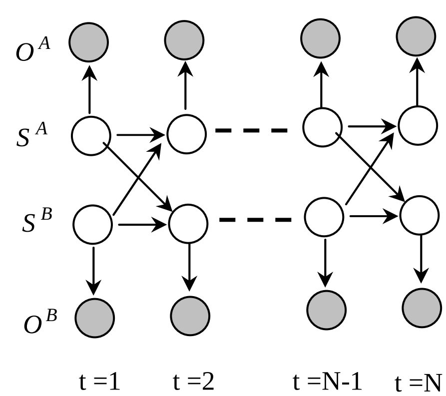
$$\hat{I}(A,B) = \log \frac{P_{CHMM}(O^A, O^B \mid \hat{\theta}_C)}{P_{HMM}(O^A \mid \hat{\theta}_A) P_{HMM}(O^B \mid \hat{\theta}_B)}. \quad (15)$$

Since when there is on intentional interaction between two objects, the conditional probability of the hidden state of one chain is independent of the hidden state of the other chain, we can also estimate $I(A,B)$ by observing the transition probability matrix,

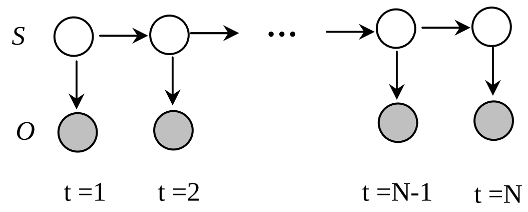
$$\hat{I}(A,B) = \sum_{i,j=1}^{2} \left\| p\left(s_t^A = i \middle| s_{t-1}^A = j, s_{t-1}^B = 1\right) - p\left(s_t^A = i \middle| s_{t-1}^A = j, s_{t-1}^B = 2\right) \right\|$$

$$+ \sum_{i,j=1}^{2} \left\| p\left(s_t^B = i \middle| s_{t-1}^B = j, s_{t-1}^A = 1\right) - p\left(s_t^B = i \middle| s_{t-1}^B = j, s_{t-1}^A = 2\right) \right\|. \quad (16)$$

Here, we assume that there only two sates for each chain.



Coupled HMM



HMM

Figure 14. Graphical models of coupled HMM and HMM. S^A, S^B and S are the hidden variables. O^A, O^B and O are the observation variables.

6.2. Simulation experiment

In the normal cases, when one object follows another object without intention, the two objects will change their speed independently without effect on each other. However, if *A* is intentionally following *B*, *A* will change its speed according to the speed variation of *B*. If B speeds up, *A* will accelerate after a short time in order to keep the track of *B*. If *B* starts to move at a low speed, *A* will also gradually slow down avoiding being noticed by *B*. This process is simulated in Figure 15. Using the speed variations along time as observation sequences, the data streams of the two objects are modeled as coupled HMMs. The transition probabilities of the coupled HMMs are shown in Table 2. We can observe that: (1) the conditional probability of the hidden state of *B* at time *t* is almost completely decided by the hidden state of *B* at the previous time *t-1*, and it is little affected by the hidden state of *A* at time *t-1*, so *B* freely changes its speed independent of *A*. (2) the conditional probability of the hidden state of *A* depends on not only its own hidden state at *t-1*, but also the hidden state of *B* at *t-1*. So there is intentional interaction between *A* and *B*. The dependences of the two data streams computed using Eq. (15) and Eq. (16) are 19.7 and 0.9520 respectively.

For comparison, we simulate two independent speed variations as shown in Figure 16, and also build the coupled HMMs. The transition probabilities of the coupled HMMs are shown in Table 3. For each object, the conditional probability of the hidden state at time *t* only depends on its own hidden state at *t-1*, while is independent of the hidden state of the other object at *t-1*. So there is no intentional interaction between the two objects. The dependences of the two data streams computed using Eq. (15) and Eq. (16) are 1.6956 and 0.172 respectively, much smaller than those compute from the data in Figure 15.

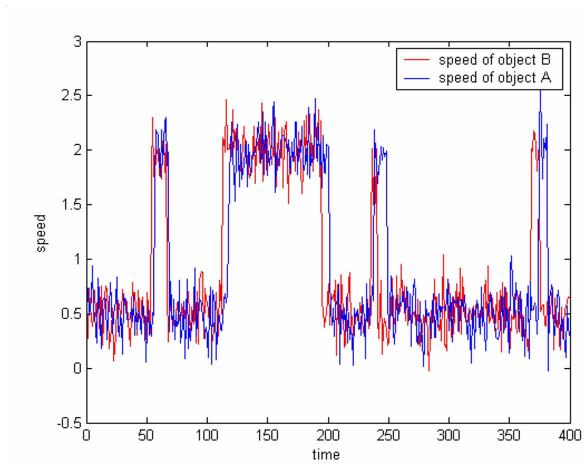


Figure 15. Speed variations of two objects in a simulated intentional *follow*. Object *B* freely change its speed. Object A is intentionally following *B*. *A* changes its speed according to the speed variation of *B*.

Table 2. Transition probabilities of coupled HMMs modeling the data shown in Figure 13. *A* is intentionally following *B*.

	$s_t^A=1$	$s_t^A=2$	$s_t^B=1$	$s_t^B=2$
$s_{t-1}^A=1, s_{t-1}^B=1$	1	0	0.985	0.015
$s_{t-1}^A=1, s_{t-1}^B=2$	0.765	0.235	0	1
$s_{t-1}^A=2, s_{t-1}^B=1$	0.182	0.818	1	0
$s_{t-1}^A=2, s_{t-1}^B=2$	0	1	0.044	0.956

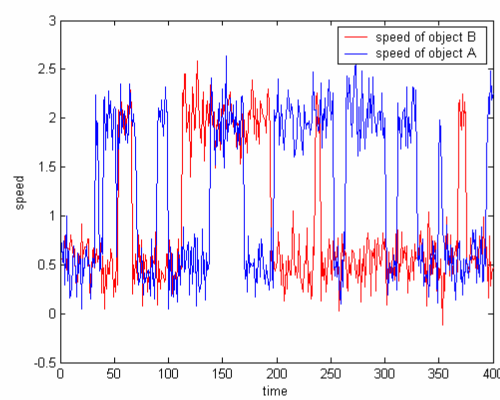


Figure 16. Speed variations of two objects which change their speed independently.

Table 3. Transition probabilities of coupled HMMs modeling the data shown in Figure 14. The two object independently change their speed.

	$s_t^A = 1$	$s_t^A = 2$	$s_t^B = 1$	$s_t^B = 2$
$s_{t-1}^A = 1$, $s_{t-1}^B = 1$	0.960	0.040	0.980	0.020
$s_{t-1}^A = 1$, $s_{t-1}^B = 2$	0.966	0.034	0.017	0.983
$s_{t-1}^A = 2$, $s_{t-1}^B = 1$	0.014	0.986	0.944	0.056
$s_{t-1}^A = 2$, $s_{t-1}^B = 2$	0.014	0.986	0.061	0.939

7. Conclusion

In this paper, we propose a framework to detect interactions in far-field visual surveillance. Long-range interactions are described using trajectory analysis. Trajectories are extracted as features to represent interactions. Using a novel trajectory similarity measure, the similarity between interactions is computed. Abnormal interactions are detected as outlier samples by comparing interactions. Modeling the activities of objects as HMMs, intentional interactions are detected by computing the mutual information between HMMs. Because of the difficulty of data collection, in this work, we use some simulation data for experiments of abnormal and intentional interaction detection. In the future work, we will try to work on some real scenarios.

References

[1] H. Zhong and J. Shi, “Finding (Un)Usual Events in Videos,” Tech. Report, CMU-TR-03-05, Robotics Institute, Carnegie Mellon University, May, 2003.

[2] N.M. Oliver, B. Rosario, and P. Pentland, "A Bayesian Computer Vision System for Modeling Human Interactions," IEEE Trans. on PAMI, Vol. 22, No.8, August 2000.

[3] S. Hongeng, F. Bremond, and R. Nevatia, "Representation and Optimal Recognition of Human Activities," in Proceedings of CVPR 2000.

[4] S. Hongeng and R. Nevatia, "Multi-Agent Event Recognition," in Proceedings of ICCV 2001.

[5] N. Ghanem, D. DeMenthon, D. Doermann, and L. Davis, "Representation and Recognition of Events in Surveillance Video Using Petri Nets," 2004 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'04) Volume 7, June 27 - July 02, 2004, Washington, D.C., USA.

[6] J. F. Allen, “Towards a General Theory of Action and Time,” Artificial Intelligence, 23 (2): pp. 123-154, 1984.

[7] C. Kohler, “Selecting Ghosts and Queues from a Car Trackers Output using a Spatial-Temporal Query Language,” in Proceedings of CVPR, 2004.

[8] A. R. Chowdhury, R. Chellappa, “ A Factorization Approach for Activity Recognition”, in Proceedings of CVPR 2003.

[9] W.E.L. Grimson, L.Lee , R. Romano and C. Stauffer, “Using adaptive tracking to classify and monitor activities in a site”, in Proceedings of CVPR 1998.

[10] F. Porikli and T. Haga, “Event Detection by Eigenvector Decomposition Using Object and Frame Features”, in Proceedings of CVPRW 2004.

[11] T. Starner and A. Pentland, “Visual recognition of American sigh language using hidden markov models”, Proceedings of Int’l Workshop Automatic Face- and Gesture-Recognition, 1995.

[12] C. Town, “Ontology-driven Bayesian Networks for Dynamic Scene Understanding”, Proceedings of CVPRW 2004.

[13] J. Fernyhough, A. G. Cohn, D. C. Hogg, "Constructing qualitative event models automatically from video input," Image and Vision Computing, Vol. 18, pp. 81-103, 2000.

[14] N. V. Weghe, A.G. Cohn, P. Bogaert and P. Maeyer, “Representation of Moving Objects Along a Road Network,” In Proceedings of 12th Int. Conf. on Geoinformatics, 2004.

[15] C. Stauffer and E. Grimson, “Learning Patterns of Activity Using Real-Time Tracking,” IEEE Trans. on PAMI, Vol. 22, No. 8, pp. 747-757, 2000.

[16] B. Neumann, “Natural Language Descriptions of Time-Varying Scenes,” in D.L. Waltz (Ed.), Semantic Structures: Advances in Natural Language Processing, Lawrence, Hillsdale, NJ. 1980, pp. 167-206.

[17] M. P. Dubuisson and A. K. Jain, “A Modified Hausdorff distance for Object Matching,” in Proc. of ICPR, 1994.