Occlusion Sequence Mining for Complex Multi-Agent Activity Discovery

Prithwijit Guha
Dept. of Electrical Engg.
IIT Kanpur, India
Email: pguha@iitk.ac.in

Arindam Biswas
Dept. of Computer Sc. & Engg.
IIT Kanpur, India
Email: arindam@cse.iitk.ac.in

Amitabha Mukerjee
Dept. of Computer Sc. & Engg.
IIT Kanpur, India
Email: amit@cse.iitk.ac.in

K.S. Venkatesh
Dept. of Electrical Engg.
IIT Kanpur, India
Email: venkats@iitk.ac.in

Abstract

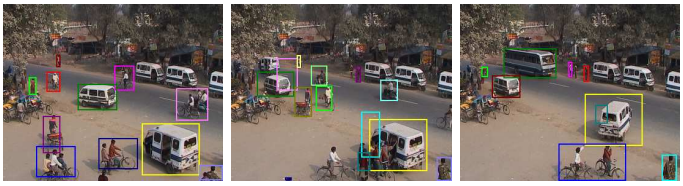
Complex multi-agent interactions result in occlusion sequences which are a visual signature for the event. In this work, multi-agent interactions are tracked using a set of qualitative occlusion primitives derived on the basis of the Persistence Hypothesis - objects continue to exist even when hidden from view. Variable length temporal sequences of occlusion primitives are shown to be well-correlated with many classes of semantically significant events. In surveillance applications, determining occlusion primitives is based on foreground blob tracking, and requires no prior knowledge of the domain or camera calibration. New foreground blobs are identified as putative agents which may undergo occlusions, split into multiple agents, merge back again, etc. Significant activities are identified through temporal sequence mining, and these bear high correlation with semantic categories (e.g. disembarking from a vehicle involves a series of splits). Thus semantically significant event categories can be recognized without assuming camera calibration or any environment/agent/action model priors.

I. Introduction

Object interactions in 3D space often leave their imprint in image space in terms of occlusions. Instead of treating occlusions as a problem, we show that temporal sequences of occlusion phenomena constitute a qualitative signature for large classes of events. In particular, events involving actual contact (push, embark, hit) necessarily involve overlap in image space for part of the event history. However, many classes of non-contact situations also result in occlusions, and the sequence of such occlusions can (depending on viewpoint) lead to characterization of specific events (e.g. overtaking, crossing). We claim that occlusions between agents, other agents, and scene objects constitute an inexpensive and cognitively important cue to reasoning about interactions in space.

Here we explore the limits of what can be learned based on occlusion phenomena. The primary advantage of such an approach is that unlike quantitative approaches using supervised priors for object behavior recognition, (e.g. [1], [2], [3]), occlusion signatures do not require any priors for either agents or events. From a cognitive perspective, categorizing events by combining occlusion with other low-level features such as trajectory and segmentation may constitute a key part of the process leading to formation of image schema [4]. Working together with pre-attentive cues such as image flow and motion, temporal learning in sequences of occlusion phenomena may constitute a pre-linguistic model for concept formation for both activities and agents. These links to cognitive processes also reflect computational efficiencies to be gained by focusing attention and avoiding more expensive 3D computations as called for in [5].

In order to learn events from sequences of occlusion states, we construct variable length temporal sequences of occlusion primitives, and generate signatures for a wide class of actions. For example, a group of people hugging each other, a person coming on a bicycle, getting off and going into a building, a crowd of people embarking a tempo (Figure 1), are all events which have stable signatures in terms of occlusion primitives or *O-primitives*. The feature set for sequence mining constitutes of the O-primitives together with quantified motion features.



(a) (b) (c)

Fig. 1. Embarking a tempo (a short distance public tranport on Indian roads). (a) A crowd approaching a tempo, (b) crowd embarking tempo, (c) tempo moving away

Occlusion mining is especially relevant for activities where agent trajectories result in occlusion at least for part of its scene presence. Such activities may include:

- agents interacting without contact (e.g. two cars crossing each other).
- limited contact situations (e.g. two persons shaking hands)
- agents completely enclosed within other objects (e.g. people entering a bus)
- agents emerging from other objects (e.g. disembarking a vehicle), etc.

Occlusion signatures constitute an extremely general feature set - clearly finer discrimination calls for more object specific features, e.g. in distinguishing actions such as hugging from handshaking.

Once events are discovered from sequence mining, an important discovery constitutes the number of agents involved in the activity. These are relevant in conceptual and linguistic models for the action. Single-agent actions correspond to

intransitive verbs in language, while multi-agent interactions corresponding to transitive verbs. These are distinguished by the number of objects whose features participate in the event discovery process. Single-Agent actions like "turning left onto the road" are based on a monadic set of features (based on a single agent's motion vectors) while agent-object interactions mostly involve some degree of occlusion in actions such as *overtake, embark*, etc.

The next section presents the underlying work on foreground extraction, and section III develops the algorithm for multi-agent tracking and subsequent O-primitive identification. The proposed approach for action/interaction learning through incremental sequence mining is detailed in Section IV. The experimental results of activity discovery are presented in Section V and conclusions in section VI.

II. Foreground Extraction

Agents are identified as foreground regions based on one of two kinds of evidence: first, as regions of change with respect to a learned background model; and second, as regions exhibiting motion. Learning the background model in presence of agents is a challenging problem in itself. Several approaches have been proposed to incrementally learn the background scene model in the presence of agents. The most commonly adopted algorithms include the computation of median [2] or fitting (temporally evolving) Gaussian mixture models [6], [7] on the temporal pixel color histogram of the image sequence. These approaches continuously learn the multi-modal mixture models with the assumption that the moving objects appear at a certain pixel only temporarily and the *true* background remains accessible to the system more frequently leading to higher weight of the corresponding mode. However, such an approach is prone to transient errors persisting over a number of frames (depending on the learning rate), resulting in two types of errors. First, if agents learned as part of the background suddenly start moving, ghosts and holes appear in the foreground segmentation. Second, when a moving agent comes to stasis, it is eventually learned as a part of the background, which may not be desirable in itself, and also in the transition period, objects interacting with it would not be identified. Both these problems are averted in the present approach by combining background-model and motion evidence, and updating based on tracking / previous motion-history feedback.

Generally, the background model $\mathcal{B}_t$ at the t^{th} instant is selectively updated based on the classification results of the t^{th} frame Ω_t. Classification based on $\mathcal{B}_{t-1}$ first results in a set of foreground pixels $\mathbf{F}_b(t) \subset \Omega_t$. Next, an inter-frame motion estimation [8] is performed between Ω_t and Ω_{t+1} to delineate the set of moving foreground pixels $\mathbf{F}_m(t) \subset \Omega_t$. This results in single-frame latency that helps us in identifying the regions that suddenly start moving or come to a stop.

Pixels identified as both foreground and moving are clearly identified as agent pixels. Among the mismatched pixels, moving pixels not identified as foreground, are denoted as

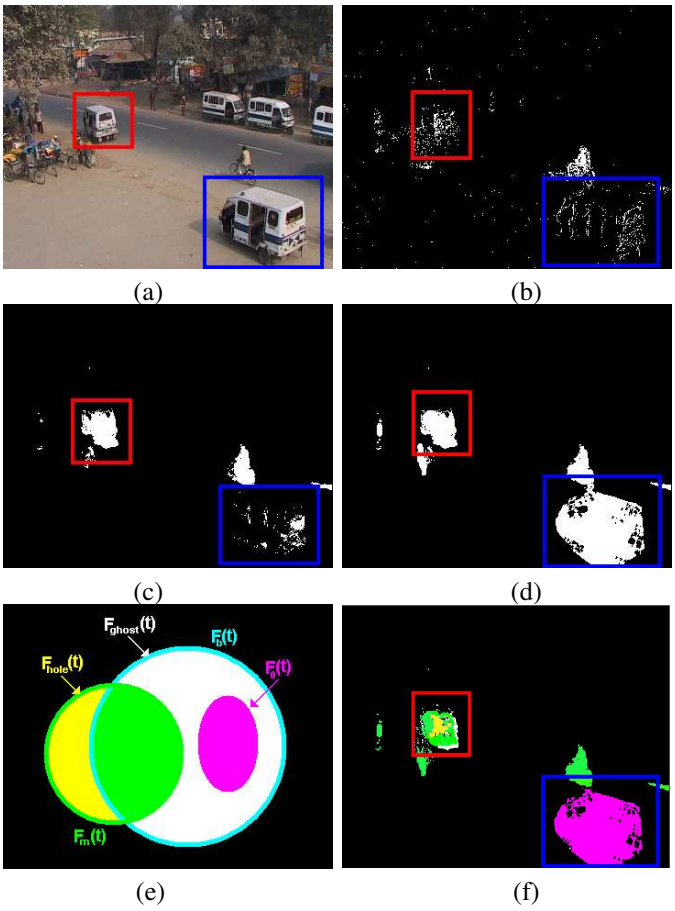


Fig. 2. Results of foreground detection. (a) The *tempo* highlighted by red bounding box suddenly starts moving and the *tempo* highlighted by blue bounding box comes to rest (Frame 1628); Foreground extraction results (b) using only per pixel Gaussian mixture model; (c) with moving pixel detection only and (d) after applying both moving pixel detection and tracking feedback; (e) A Venn diagram indicating the sets $\mathbf{F}_b(t)$, $\mathbf{F}_m(t)$, $\mathbf{F}_{hole}(t)$, $\mathbf{F}_{ghost}(t)$ and $\mathbf{F}_0(t)$; (f) Results of classifying the pixels corresponding to moving regions (yellow and green), holes (yellow), ghosts (white) and agents coming to rest (pink).

$\mathbf{F}_{hole}(t) = \mathbf{F}_m(t) - \mathbf{F}_b(t)$. On the other hand, the set of non-moving pixels in $\mathbf{F}_b(t)$, is given by $\mathbf{F}_{ghost}(t) = \mathbf{F}_b(t) - \mathbf{F}_m$ and is identified as possible background candidate. However, these non-moving ghost pixels may contain actual agent regions which have not shown up in the optical flow, or where an agent has actually come to a stasis. Using information from the motion history and tracking (discussed in Section III) we delineate the set of agents pixels that have come to rest, $\mathbf{F}_0(t) \subset \mathbf{F}_{ghost}(t)$. The set of agent pixels that emerge from this analysis is defined as $\mathbf{F}(t) = (\mathbf{F}_b(t) - \mathbf{F}_{ghost}(t)) \cup \mathbf{F}_{hole}(t) \cup \mathbf{F}_0(t)$. Now, the complement of $\mathbf{F}(t)$ is used to update the background model to $\mathcal{B}_t$. The set of detected foreground pixels $\mathbf{F}(t)$ is further subjected to shadow removal (based on the criteria of equality among sub-unity intensity modulations in the 3 color chanels), neighbourhood voting, followed by connected component analysis to obtain the set of disjoint foreground blobs $\mathcal{F}(t) = F_i(t)_{i=1}^{n_t}$. These blobs constitute the basic units (putative agents) that are tracked over

the entire sequence, and it is their participation in occlusion that results in O-primtive identification, and eventually in the activity discovery.

III. Multi-Agent Tracking

Here we adopt the multi-agent tracking algorithm proposed in [9], which works by comparing the foreground blobs at time t, $F_i(t)$ with the predictions based on their previous positions and shape. The same foreground (agent) pixel being claimed by more than one agent (foreground blob) is one of the primary indicators of occlusion.

We define several elementary occlusion behaviors according to the **Persistence Hypothesis**: Objects continue to exist even when hidden from view. The agent-blob association is performed over an *active* set $\mathcal{S}_A(t)$ containing agents tracked till the t^{th} instant and also a set $\mathcal{S}_{lost}(t-1)$ of agents which have disappeared within the viewing window. The system initializes itself with empty sets and the agents are added (removed) as they (dis)appear in the field of view. The proposed approach is a two stage process. Initially, the agents in $\mathcal{S}_A(t-1)$ are localized in the current frame Ω_t. This is followed by the identification of O-primitives by the process of agent-blob association with selective updates of agent features. This process is detailed next.

A. Agent Representation and Localization

All moving objects are considered as agents, and are detected based on extracted foreground blobs and are initialized with features computed from the blobs. Agents maintain their identity as they are successfully tracked across frames, and even when they are re-identified upon re-appearance. The j^{th} agent $\mathcal{A}_j(t)$ is characterized by its supporting region (the set of pixels it occupies, $a_j(t)$), color (weighted color distribution $h_j(t)$) and motion (position history of minimum bounding rectangle of $a_j(t)$, as its previous τ centers $c_j(t)$.

The pixel set $a_j(t)$ and weighted color distribution $h_j(t)$ are initially learned from the foreground blob extracted at the first appearance of the agent and are then updated throughout the sequence whenever it is in isolation. The color distribution $h_j(t)$ is computed from the b-bin color histogram of the region $a_j(t)$ (in Ω_t) weighted by the Epanechnikov kernel [10] supported over the minimum bounding ellipse of $a_j(t)$ (centered at $c_j(t)$) and is given by,

$$h_j[l](t) = \frac{1}{C_E} \sum_{X \in a_j(t)} \mathbf{K_E}(\|X - c_j(t)\|^2)\delta(l - B_f(X)) \quad (1)$$

$$C_E = \sum_{X \in a_j(t)} \mathbf{K_E}(\|X - c_j(t)\|^2) \quad (2)$$

Where C_E is the normalizing constant computed from the Epanechnikov kernel $\mathbf{K_E}$ and the function B_f maps the pixel location $X \equiv (x, y)$ to its corresponding color bin derived from the pixel $\Omega_t(x, y)$.

The agents in the t^{th} frame are localized by their trajectory information and color distribution obtained till the $(t-1)^{th}$ instant. An estimate $c_j^{(0)}(t)$ is obtained by extrapolating from the trajectory $\{c_j(t-1), \ldots, c_j(t-\tau)\}$. The mean-shift iterations [10], initialized at an elliptic region centered at $c_j^{(0)}(t)$ further localize the agent region at $a_j(t) \in \Omega_t$.

B. Identifying Occlusion Primitives

The extent of association between a predicted agent region $a_j(t)$ for an agent in $\mathcal{S}_A(t-1) = \{\mathcal{A}_j(t-1)\}_{j=1}^{m_{t-1}}$ and the foreground blob $F_i(t) \in \mathcal{F}(t)$ is estimated by constructing a thresholded *localization confidence matrix* $\Theta_{AF}(t)$ and the *attribution confidence matrix* $\Psi_{FA}(t)$. These confidences are computed by a fractional overlap measure $\gamma(\omega_1, \omega_2) = \frac{|\omega_1 \cap \omega_2|}{|\omega_1|}$ signifying the fraction of the region ω_1 overlapped with ω_2.

$$\Theta_{AF}[j,i](t) = \begin{cases} 1; & \gamma(a_j(t), F_i(t)) \geq \eta_A \\ 0; & \text{Otherwise} \end{cases} \quad (3)$$

$$\Psi_{FA}[i,j](t) = \begin{cases} 1; & \gamma(F_i(t), a_j(t)) \geq \eta_F \\ 0; & \text{Otherwise} \end{cases} \quad (4)$$

Where the thresholds η_A and η_F signify the extent of allowable localization and attribution confidences. The number of foreground regions attributed to the j^{th} agent ($\Theta_A[j](t) = \sum_{i=1}^{n_t} \Theta_{AF}[j,i](t)$ and $\Psi_A[j](t) = \sum_{i=1}^{n_t} \Psi_{FA}[i,j](t)$) and agents localized in $F_i(t)$ ($\Theta_F[i](t) = \sum_{j=1}^{m_{t-1}} \Theta_{AF}[j,i](t)$ and $\Psi_F[i](t) = \sum_{j=1}^{m_{t-1}} \Psi_{FA}[i,j](t)$) are further computed from these matrices to identify the occlusion primitives.

(a) (b)

Fig. 3. Cases of occlusions. (a) Partial occlusion: agent occluded by tree is visible as two fragmented blobs; in this state, the agent is recognized, but its visual characteristics are not updated. (b) Crowding: multiple agents merge to form a single blob.

The j^{th} agent in $\mathcal{S}_A(t-1)$ is **isolated** or unoccluded ($\mathcal{O}(I)[j,t]$), if the localization confidence is significantly high and the associated foreground blob is not overlapped with other agents. However, when the agent **disappears** ($\mathcal{O}(D)[j,t]$) both localization and attribution confidences fall below η_A and η_F signifying very poor or no association of the agent to any foreground blob. In case of **partial occlusions** ($\mathcal{O}(P)[j,t]$), the attribution confidence of one or more foreground blobs to the j^{th} agent remains high, although the localization confidence falls significantly. On the other hand, while in a **crowd** ($\mathcal{O}(C)[j,t]$), the localization confidence of the j^{th} agent in the crowded blob (overlapped with more than one agent) remains high although the attribution confidence of that blob to the agent remains low. Thus the four Boolean

predicates for these occlusion primitives can be constructed as follows.

$$\mathcal{O}(I)[j,t] = \exists i[\Theta_{AF}[j,i](t)=1] \wedge [\Theta_F[i](t)=1] \quad (5)$$

$$\mathcal{O}(D)[j,t] = [\Theta_A[j](t)=0] \wedge [\Psi_A[j](t)=0] \quad (6)$$

$$\mathcal{O}(P)[j,t] = \forall i[\Psi_{FA}[i,j](t)=1] \wedge [\Theta_F[i](t)=1] \wedge [\Psi_A[j](t)\geq 1] \quad (7)$$

$$\mathcal{O}(C)[j,t] = \exists i[\Theta_{AF}[j,i]=1] \wedge [\Theta_F[i](t)>1] \quad (8)$$

To obtain the current active set $\mathcal{S}_A(t)$, updates are applied to all of color, shape and trajectory of individual agents under $\mathcal{O}(I)$, but only to the trajectory of agents under $\mathcal{O}(P)$ and $\mathcal{O}(C)$. Agents under $\mathcal{O}(D)$ are moved from the active set to the putative set. This enables the system to remain updated with agent features while keeping track of them.

The **entry/reappearance** of an agent is attributed to the existence of a foreground blob $F_i(t)$ in the scene having no association with any agent from $\mathcal{S}_A(t-1)$ and is thus detected as $\mathcal{O}(N)_i(t) = [\Theta_F[i](t)=0] \wedge [\Psi_F[i](t)=0]$. The features of the new blob $F_i(t)$ are matched against those in $\mathcal{S}_{lost}(t-1)$ to search for the reappearance of agents. If a match is found, the agent is moved from $\mathcal{S}_{lost}(t-1)$ to $\mathcal{S}_A(t)$ and a *reappearance* ($\mathcal{O}(R)[j,t]$) is noted. Otherwise, a new agent is added to $\mathcal{S}_A(t)$ and the system detects an *entrance* ($\mathcal{O}(E)[j,t]$). Similarly, an agent is declared to **exit** the scene ($\mathcal{O}(X)[j,t]$), if its motion predicted region lies outside the image region and is thus removed from the active set.

IV. Activity Discovery

Activities can be broadly classified into two different categories:

- *Single agent actions* or events with a single participant, the agent. Such actions have no object on which the action is being performed, and correspond in natural language syntax to the intransitive verb category ("John runs").
- *Agent-object interactions*, or events with two or more participants, the agent, as well as an object on which the action is taking place, (e.g. "John rides a bike") which corresponds to the transitive verb category in syntax.

Models of single-agent behaviors are characterized by the agent and some characterization of the temporal character of the action - e.g. the class of trajectories the agent may take (e.g. the path taken by a vehicle in a traffic scenario or the pose sequence exhibited by a dancer). Activities in this class include "Cars drive towards the left", "the motorcycle joins the road", "the man hides behind the tree", etc.

Agent-object interactions exhibit several different modes - the actions may involve actual contact (e.g. riding a bike, boarding or disembarking a vehicle, grouping etc.) or may involve interactions at a distance (e.g. following, chasing, overtaking, etc.). In terms of image space, actual contacts are necessarily reflected in O-primitive structures, but non-contact situations do not necessarily characterized by non-overlap. More so, the agents participating in agent-object interactions may be either homogeneous (e.g. "car1 overtaking car2") or heterogeneous (e.g. "man entering the tempo").

Both single-agent actions and agent-object interactions can be expressed as temporal sequences of agent states (actions) or co-occurrent states of interacting agents. Thus, the domain of activity analysis demands efficient statistical sequence modeling techniques for recognizing significant temporal patterns from the time-series data of action/interaction features. A number of methodologies employing hidden Markov models, time-delay neural networks, recurrent networks etc. have been proposed for modeling and recognition of action/interaction sequences in a supervised learning framework. On the other hand, unsupervised learning of activity patterns have also been proposed by trajectory clustering [11] or variable length Markov model learning [12]. A good overview of such techniques can be found in [13].

Supervised activity modeling techniques are mostly task oriented and hence fail to capture the corpus of events from the time-series data provided to the system. Unsupervised data mining algorithms, on the other hand, discover the modes of spatio-temporal patterns thereby leading to the identification of a larger class of events. The use of VLMMs in the domain of activity analysis was introduced for automatic modeling of the actions in exercise sequences [14] and interactions like handshaking [12] or overtaking of vehicles [15] in a traffic scenario. These approaches propose to perform a vector quantization over the agent feature and trajectory space to generate temporally indexed agent-state sequences from video data. These sequences are parsed further to learn VLMMs leading to the discovery of behavioral models of varying temporal durations.

Motion (pose) primitives derived from agent (state) trajectories are a necessary set of activity descriptors but are not sufficient as they lack the power to describe the interactions involving agent-region contacts in the image space. We, thus augment the activity feature space with the set of occlusion primitives which form a more fundamental notion of interaction signatures. More so, we identify that the occlusion state transition sequence form a more significant interaction description than the occlusion state sequences themselves. In this work, we aim to discover the interactions arising out of agents moving in complex environments undergoing both static and dynamic occlusions with background objects and other agents respectively. In the following subsections we discuss the methodologies adopted for sequence modeling and event primitive representations for interaction modeling.

A. Incremental Transition Sequence Learning

Agent-agent/background object interactions are discovered by incremental learning of temporal sequences of event primitives from $\mathcal{E} = \{\epsilon_r\}_{r=1}^{R}$. It involves the construction of an *activity tree* $\mathcal{T}_\alpha$ whose branches represent the variable length event sequences. An empty (first in first out) buffer β_j of length L and the activity tree $\mathcal{T}_\alpha(j)$ are initialized with a root node ρ_j at the very first appearance of every j^{th} agent in $\mathcal{S}_\mathcal{A} \cup \mathcal{S}_\mathcal{P}$. This ensures the discovery of variable length event

sequences whose length do not exceed L. Each node of $\mathcal{T}_\alpha(j)$ is a two tuple $\mathcal{T}_n \equiv (\epsilon, \pi)$ containing the primitive $\epsilon \in \mathcal{E}$ and a real number $\pi \in (0, 1]$ signifying the probability of occurrence of the path $\{\rho_j, \ldots, \mathcal{T}_n\}$ among the set of all possible paths of the same length.

Let, $\beta_j[l](t)$ be the event primitive at the l^{th} depth of the buffer, the most recent one being logged at $l = 0$ and the last one at $l = L - 1$. The event primitive $\epsilon(j, t) \in \mathcal{E}$ detected for the j^{th} agent at the t^{th} instant is pushed to β_j, iff the current event primitive changes from that of the previous, i.e. $\epsilon(j, t) \neq \epsilon(j, t-1)$. This prevents learning the variable length temporal sequences of similar events as separate activities. Let, $B^{(l)}(j, t) = \{\alpha_u^{(l)}(j, t)\}_{u=1}^{b_l}$ be the set of l-length paths (originating from ρ_j) of $\mathcal{T}_\alpha(j, t)$. More so, if the sequence $\{\beta_j[l-k](t)\}_{k=1}^{l}$ signify the b^{th} path of $B^{(l)}(j, t)$, then the probabilities $\{\pi_u^{(l)}(j, t)\}_{u=1}^{b_l}$ of the nodes of $\mathcal{T}_\alpha(j, t)$ at the l^{th} depth are updated as,

$$\pi_u^{(l)}(j, t) = (1 - \eta_l(t))\pi_u^{(l)}(j, t-1) + \eta_l(t)\delta(u - b) \quad (9)$$

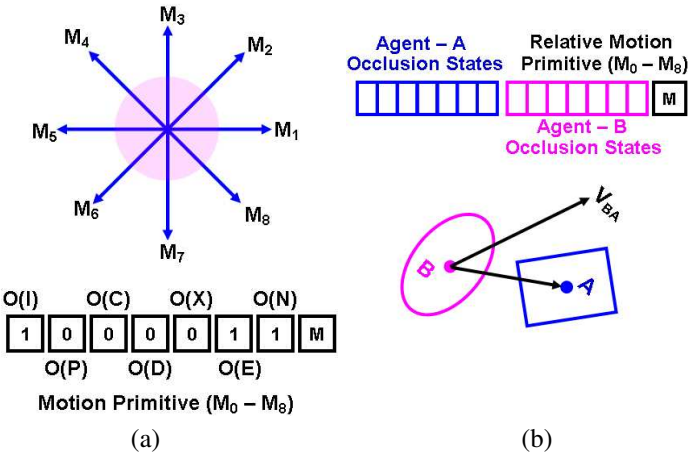
Where, $\eta_l(t)$ is the rate of learning l-length sequences at the t^{th} instant and δ is the Kronecker delta function. However, in the current implementation a fixed learning rate η is employed such that $\eta_l(t) = max\left(\frac{1}{t}, \eta\right) \forall l$. A new event primitive is added to the tree with an initial probability of $\eta_l(t)$ and the self normalizing nature of equation 9 ensures the properties of the probability measure at each depth of $\mathcal{T}_\alpha(j)$.

B. Unsupervised Interaction Learning

We construct event primitives for agents by combining their occlusion states and motion primitives. The occlusion states of *isolation* ($\mathcal{O}(I)$), *partial occlusion* ($\mathcal{O}(P)$), *crowded* ($\mathcal{O}(C)$), *disappeared* ($\mathcal{O}(D)$), *exit* ($\mathcal{O}(X)$), *entry* ($\mathcal{O}(E)$) and *entrance of new agent in neighborhood* ($\mathcal{O}(N)$) to form a 7-bit occlusion driven interaction descriptor. The direction of (relative) motion of the agent is quantized to assign one of the eight motion primitives $\mathcal{M}_1$ to $\mathcal{M}_8$ signifying the directions of *East*, *North-East*, *toSouth-East* (going anti-clockwise) respectively. Besides, a motion primitive $\mathcal{M}_0$ is used to signify the state of stasis of the agent. The final event descriptor for a single agent is formed by augmenting the occlusion and motion primitives as shown in figure 4(a).

Consider a short video sequence where a person walks across a tree from left to right in the image space from which we sample 18 frames to illustrate the process of agent-background object interaction discovery. Key frames from this sequence are shown in figure 5(a)-(e). Incremental transition sequence learning is performed with a maximum depth of $L = 10$ and a learning rate η inversely proportional to the frame number. The growth of the activity tree is shown in figure 5(f).

Semantic labels can be assigned to the sequences in the occlusion-primitive space to denote different activities, and subsequences may constitute sub-activities. For example, consider the longest path $\{(\mathcal{O}(I), \mathcal{M}_1), (\mathcal{O}(P), \mathcal{M}_1), (\mathcal{O}(D), \mathcal{M}_0), (\mathcal{O}(P), \mathcal{M}_1), (\mathcal{O}(I), \mathcal{M}_1)\}$



(a) (b)

Fig. 4. Event primitive descriptors. (a) Combining occlusion and motion states form the agent-background object interaction primitives. (b) Agent B is considered to be interacting with A, if the center of the former lies within an attentional window of the later. Combining the co-occurrent occlusion states of A and B with the relative motion primitives $\mathcal{M}_i$ $(i = 0, \ldots, 8)$ form the agent-agent interaction primitive. Temporally ordered sequence of these primitives are parsed further to discover significant and meaningful activity descriptors.

learned in the activity tree from the aforementioned video that correspond to the activity of **walking across a tree from left to right**. Subsequences of this path viz. $\{(\mathcal{O}(I), \mathcal{M}_1), (\mathcal{O}(P), \mathcal{M}_1), (\mathcal{O}(D), \mathcal{M}_0)\}$ and $\{(\mathcal{O}(D), \mathcal{M}_0), (\mathcal{O}(P), \mathcal{M}_1), (\mathcal{O}(I), \mathcal{M}_1)\}$ also correspond to the visually significant events of **going to hide from left to right** and **reappearing and moving to the right**.

We consider the agent B to be interacting with A, if the center of the minimum bounding box of the former lies within an attentional window of the later [15]. The interaction primitives are formed by combining the co-occurrent occlusion states of the interacting agents (taken two at a time) along with the motion primitive obtained from the relative velocity between the agents (figure 4(b)). The relative motion primitive is computed by quantizing the angle measured from the vector $\overline{BA}$ to the relative velocity (of B with respect to A) vector $\vec{V}_{BA}$ in an anti-clockwise direction. Figure 6 shows the results of discovering the interaction sequences of **overtaking** and **crossing** from a traffic video.

V. Results

Experiments are performed on a traffic surveillance video of 30 minutes duration consisting of a wide variety of vehicles like bikes, rickshaw, cars, heavy vehicles etc along with men and animals. The background modeling is performed by learning pixel-wise mixture of Gaussians over the RGB color space with a learning rate of $\alpha = 0.01$ and a diagonal covariance matrix $\Sigma_{init} = \{4.0\}$. The foreground extraction is performed with inter-frame motion information and selective model update with higher layer agent position feedback. Comparative results of foreground extraction are shown in figure 2, SectionII.

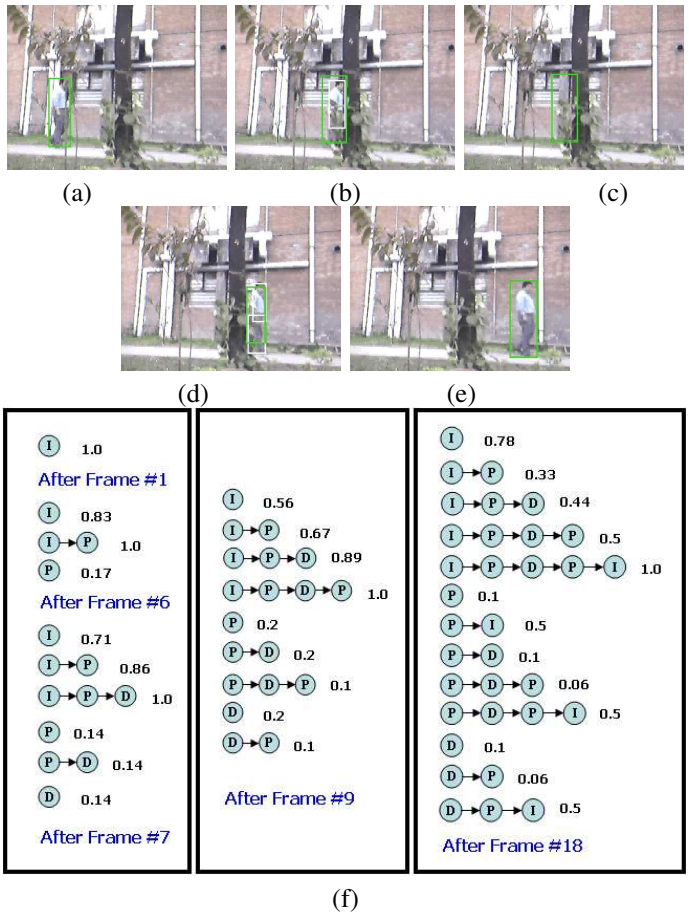


Fig. 5. The example video sequence. Agent (a) *isolated* (frame $1-5$), (b) *partially occluded* (frame 6), (c) *disappeared* (frame $7-8$), (d) *partially occluded* (frame 9), (e) *isolated* (frame $10-18$) and moving from left to right ($\mathcal{M}_1$). (f) The growth of the activity tree formed by incremental transition sequence learning after processing the frames 1, 6, 7, 9 and 18 shows the different variable length event primitive sequences (along with their relative frequencies of occurence) mined as its branches. As for example, consider the 3-length sequence $\{(I-P-D), 0.89\}$ in the activity tree learned upto frame 9. This implies that the event primitive sequence $\{(\mathcal{O}(I), \mathcal{M}_1) \rightarrow (\mathcal{O}(P), \mathcal{M}_1) \rightarrow (\mathcal{O}(D), \mathcal{M}_0)\}$ corresponding to the activity of *hiding while moving from left to right* occurs with a relative frequency of 89% among all observed 3-length sequences.

Multiple agents in the traffic video are tracked with O-primitive identification. The tracking performance of the j^{th} agent at the t^{th} instant is evaluated by the fraction of the ground-truth region of the same ($G_j(t)$) overlapped with the region $a_j(t)$, localized by the proposed algorithm and is thus given by the quantity $\gamma(G_j(t), a_j(t))$. Hence, if there are $m_g(t)$ number of agents present in the ground-truth marked images at the t^{th}, instant, then the overall performance $\mathcal{P}$ for a video of T frames is given by,

$$\mathcal{P} = \frac{1}{T}\sum_{t=1}^{T}\frac{1}{m_g(t)}\sum_{j=1}^{m_g(t)}\gamma(G_j(t), a_j(t)) \qquad (10)$$

The above measure of overall performance $\mathcal{P}$ signifies the average fraction of the actual agent regions (or ground-truth regions) localized by the tracking algorithm in a certain video

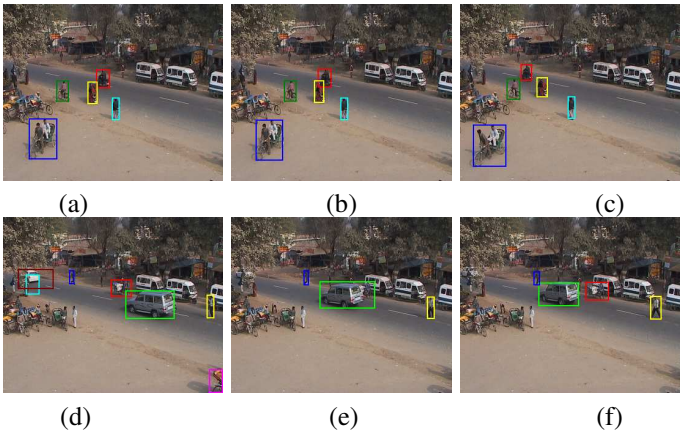


Fig. 6. Overtaking sequence (frame $853-868$). (a)-(c) A *man on bike* (Agent B, marked by red bounding box) overtaking another *man on bike* (Agent A, marked by yellow bounding box) generating a sequence $\{(\mathcal{O}_A(I), \mathcal{O}_B(I), \mathcal{M}(4)) \rightarrow (\mathcal{O}_A(C), \mathcal{O}_B(C), \mathcal{M}(3)) \rightarrow (\mathcal{O}_A(I), \mathcal{O}_B(I), \mathcal{M}(2))\}$. Crossing sequence (frame $124-138$). (d)-(f) A *car* (Agent A, marked by green bounding box) crossing a *rickshaw* (Agent B, marked by red bounding box) generating a sequence $\{(\mathcal{O}_A(I), \mathcal{O}_B(I), \mathcal{M}(1)) \rightarrow (\mathcal{O}_A(C), \mathcal{O}_B(C), \mathcal{M}(1)) \rightarrow (\mathcal{O}_A(I), \mathcal{O}_B(I), \mathcal{M}(1))\}$.

sequence. The overall performance varies, as the thresholds η_A and η_F are changed. It is evident from equations 3 and 4 that, as the thresholds η_A and η_F are increased, the detection rates of correspondences between predicted agent regions and foreground blobs reduce and thus the rate of track loss increases. On the other hand, too low values of these thresholds would increase the number of false detections of the O-primitives. Thus, to achieve optimal performances, we have chosen $\eta_A = \eta_F = 0.6$ and an overall tracking performance of approximately 68% was observed.

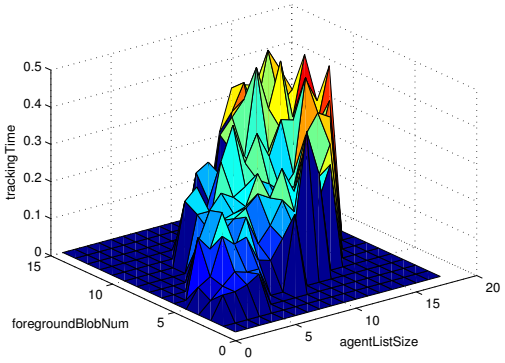


Fig. 7. Surface plot of tracking time (in seconds) with respect to number of active agents and foreground blobs

The tracking time largely depends on the number of agents in the active and putative set along with the number of foreground blobs. The variation of the tracking time with respect to these two factors is shown in figure 7. It is worth noting, that an estimate of the algorithm execution time with respect to crowding can also be obtained from this graph. The

results of tracking in the traffic surveillance video are shown in figure 8.

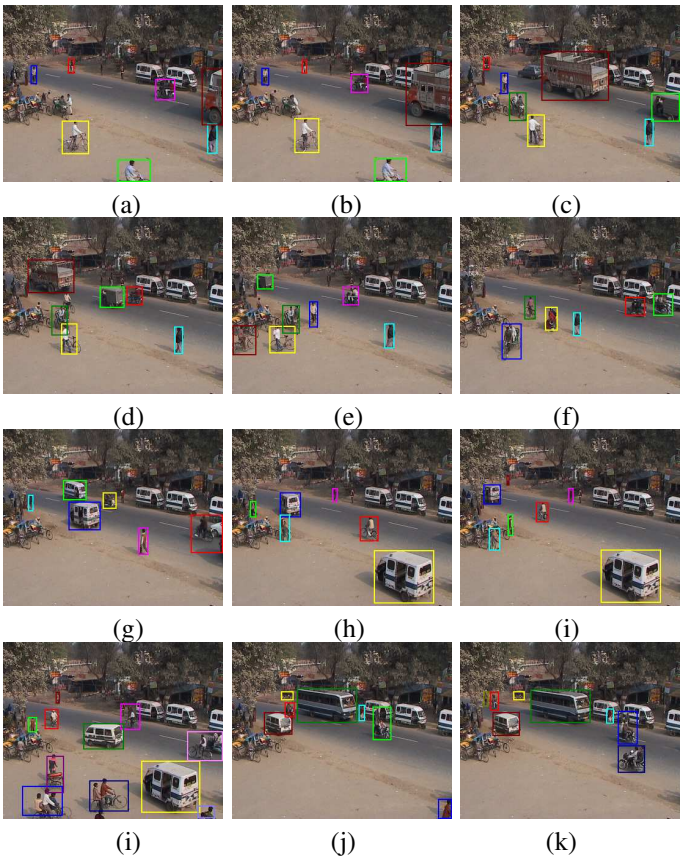


Fig. 8. (a)-(k) Results of tracking in the traffic surveillance video

The results of multi-agent tracking are logged into a database, where each agent is stored with its various appearances (learned only when isolated), image space trajectory and occlusion state sequence for its scene presence in the surveillance video. These constitute the surveillance logs from which the agent information can be retrieved with simple SQL queries. We assume the availability of object recognition modules that can categorize the agents based on their appearance features. A few samples from the surveillance logs (as seen in the HTML front-end) are shown in figure 9.

The activities are learned with a maximum depth of $L = 10$ and a learning rate of $\eta_t = max\left(\frac{1}{t}, 0.01\right)$ at the t^{th} instant. Activities are discovered for a particular query agent by mining its monadic and dyadic occlusion and motion primitive sequences. We have empirically chosen an attentional window of size 1.5 times of the minimum bounding box of the agent for all our experiments. In addition to overtaking and crossing, we have discovered the activities of **(dis)embarking** vehicles in the traffic video. The results of these interactions are shown in figure 10.

VI. Conclusion

This paper demonstrates the power of capturing temporal events in terms of occlusion features or O-primitives along

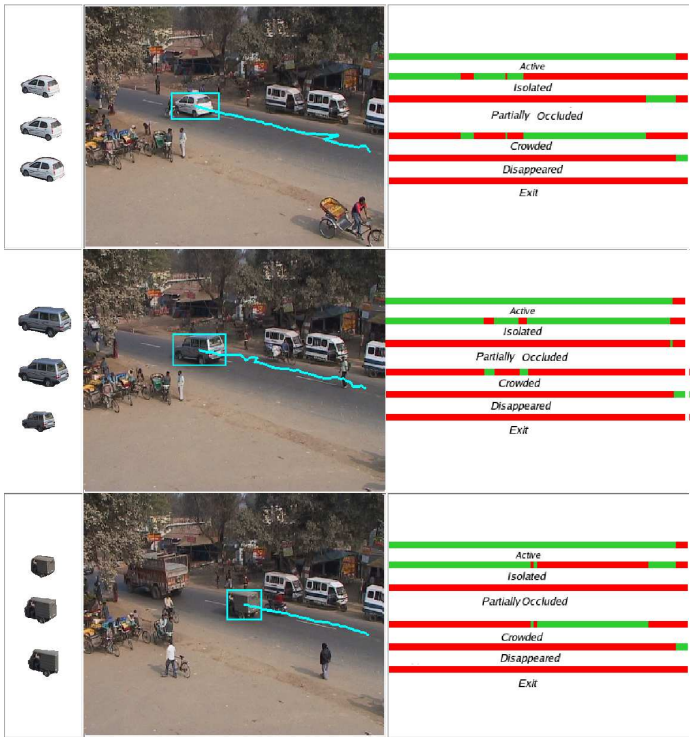


Fig. 9. Sample surveillance logs showing the agent appearances in the left, trajectories in the middle and occlusion primitive timelines in the right-most column for two cars and a tempo. The occurrence of different occlusion primitives are drawn in green in their respective timelines.

with the image plane motion. The temporal sequences of O-primitives are posited as a powerful tool for identifying agent-background object / multi-agent interactions. The system performs robust foreground extraction by using online background learning, inter-frame motion evidence and multi-agent tracking information. The occlusion states (O-primitives) of the agents are identified by analysing the fractional overlaps between the supporting regions of the agents and the foreground blobs. The O-primitves along with quantized (relative) motion primitves are used to form the atomic events. Temporal sequences of these event primitives are mined to discover several activities like embarking, disembarking, crossing, overatking, hiding, re-appearing etc.

In future work, we plan to explore other low-level tools available for activity recognition. With the mildest constraints on camera calibration (that it is nearly horizontal) one may add further motion characterizations such as *translate left/right/towards/away*, *rotate*, *speeding up*, *halting* etc. which can by themselves be informative for many actions.

Event predicates are characterized by the type-of-activity (modeled as a fine-grained image schema) as well as the ordered set of agents participating in it, as well as adjunct characteristics such as time, place, manner etc. In this work, we have classified agents only by their shape and motion characteristics, but possibly a more important characterization is in terms of actions that an agent participates in (e.g. what objects participate in embark/disembark events?). These are

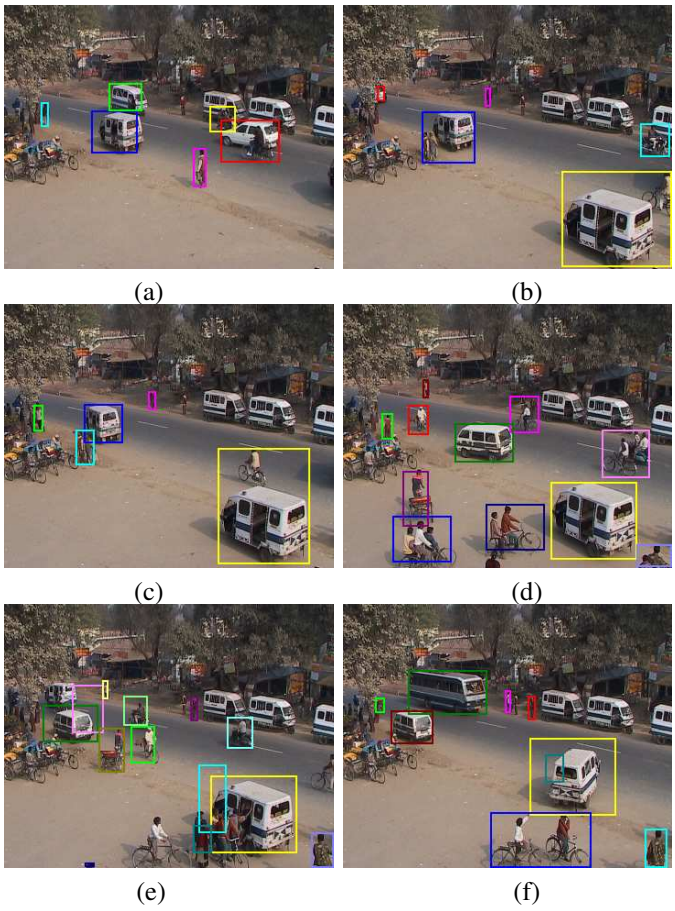


Fig. 10. Results of Activity Discovery. (a-c) Disembarking from Vehicle (marked in blue bounding box). (a) *Tempo* comes to stop (frame 1439); (b) Fragmentation due to people disembarking (frame 1594); (c) New Agents (men) formed in neighborhood of Tempo (frame 1624). (d-f) Embarking on vehicle (marked in yellow bounding box). (a) *men* in attentional window of *Tempo* (frame 1923); (b) men crowded with tempo (frame 2027); (c) men disappear, tempo still tracked (frame 2319)

important areas for further analysis.

Based on these low-level categories, one can build up to higher level constructs based on several sources of additional information:

- *Co-occurring linguistic descriptions*: It would be simple enough to identify the most salient action and its participants which would result in grounded models of the head verb and its noun subcategories. Linguistic text also serves as *anchors* for the categories learned, preventing them from Wittgensteinian drift, and also enabling broader clusters to form from the basic level categories being learned here.
- *Camera Calibration / Ground-Plane Assumption*: By using camera calibration data and making ground-plane assumptions for the agents in a given domain, considerable evidence can be added to the event characterizations.
- *Shape and Scene priors*: Availability of agent shape priors, while valuable, we would like to avoid for some time since it limits the scalability of the approach.

An important consequence of our activity analysis may relate to a long lasting debate in language acquisition - the role of syntax in verb learning. The *Syntactic Bootstrapping* view posits that the syntactic structure of verb usage informs the learner how many arguments to expect, while the *Semantic Bootstrapping* view [16] claims that children are making this inference based on semantic cues, e.g. the prepositions. It would appear plausible from our event-learning here that the distinction between actions involving single agents and actions involving agents and (linguistic) objects may be perceptually determinable in pre-linguistic cognition based on the dimension of the feature space in which the actions are learn-able. Thus, transitive verbs are learned in a feature space with the dimensions of a single-agent motions, while agent-object interactions involve relative-motions, as well as interactions such as occlusion states. Thus, the semantic features of the action may specify the number of participants as a consequence of what is perhaps already categorized in the pre-linguistic system.

REFERENCES

[1] D. Gavrilla, "Visual analysis of human movement: A survey," in *Computer Vision and Image understanding*, vol. 73, 1999, pp. 82–98.
[2] I. Haritaoglu, D. Harwood, and L. Davis, "W4 : Real time surveillance of people and their activities," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 809–830, August 2000.
[3] T. Zhao and R. Nevatia, "Tracking multiple humans in crowded environments," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, July 2004, pp. 406–413.
[4] J. Mandler, "How to build a baby: Ii. conceptual primitives," *Psychological Review*, vol. 99, no. 4, pp. 587–604, 1992.
[5] G. Granlund, "Organization of architectures for cognitive vision systems," in *Proceedings of Workshop on Cognitive Vision*, Schloss Dagstuhl, Germany, October 2003.
[6] C. Stauffer and W. Grimson, "Adaptive background mixture models for real-time tracking," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, June 1999, pp. 246–252.
[7] Z. Zivkovic, "Improved adaptive gaussian mixture model for background subtraction," in *Proceedings of the 17th International Conference on Pattern Recognition*, vol. 2, 2004, pp. 28–31.
[8] M. Proesmans, L. V. Gool, E. Pauwels, and A. Osterlinck, "Determination of optical flow and its discontinuities using non-linear diffusion," in *The 3rd Eurpoean Conference on Computer Vision*, vol. 2, 1994, pp. 295–304.
[9] P. Guha, A. Mukerjee, and K. Venkatesh, "Efficient occlusion handling for multiple agent tracking with surveillance event primitives," in *The Second Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, October 2005.
[10] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Transaction on Pattern Analysis Machine Intelligence*, vol. 25, no. 5, pp. 564–575, 2003.
[11] N. Johnson and D. Hogg, "Learning the distribution of object trajectories for event recognition," in *Proceedings of the 6th British conference on Machine vision (Vol. 2)*. BMVA Press, 1995, pp. 583–592.
[12] A. Galata, N. Johnson, and D. Hogg, "Learning variable-length markov models of behavior," *Computer Vision and Image Understanding*, vol. 81, no. 3, pp. 398–413, 2001.
[13] H. Buxton, "Learning and understanding dynamic scene activity: a review," *Image and Vision Computing*, vol. 21, no. 1, pp. 125–136, 2003.
[14] N. Johnson, A. Galata, and D. Hogg, "The acquisition and use of interaction behavior models," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. IEEE Computer Society, 1998, pp. 866–871.
[15] A. Galata, A. G. Cohn, D. Magee, and D. Hogg, "Modeling interaction using learnt qualitative spatio-temporal relations and variable length markov models," in *Proceedings of European Conference on Artificial Intelligence*, F. van Harmelen, Ed., 2002, pp. 741–745.
[16] S. Pinker, "How could a child use verb syntax to learn verb semantics?" *Lingua*, vol. 92, pp. 377–410, 1994.