Pattern Recognition Letters 30 (2009) 180–186

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Semantic event representation and recognition using syntactic attribute graph grammar

Liang Lin [a,b,*], Haifeng Gong [b], Li Li [c], Liang Wang [d]

[a] School of Information Science and Technology, Beijing Institute of Technology, Beijing 100081, China
[b] Lotus Hill Research Institute for Computer Vision and Information Science, Ezhou 436000, China
[c] Ocean University of China, Qingdao 266071, China
[d] Institute of Computing Technology of CAS, Beijing 100190, China

ARTICLE INFO

Article history:
Available online 18 March 2008

Keywords:
Visual surveillance
Event representation
Event recognition
Attribute graph grammar

ABSTRACT

The representation and recognition of complex semantic events (e.g. illegal parking, stealing objects) is a challenging task for high-level understanding of video sequence. To solve this problem, an attribute graph grammar for events modeling is studied in this paper. This grammar models the variability of semantic events by a set of meaningful "event components" with the spatio-temporal constraints. The event components are defined manually according to their semantic meaning, and further decomposed into atomic event primitives. These event primitives are learned on a object-trajectory table that describes mobile object attributes (location, velocity, and visibility) in a video sequence. A dictionary of temporal and spatial relations are defined to constrain the event primitives. With this representation, one observed event can be parsed into an "event parse graph", and all possible variability of one event can be modeled into an "event And–Or graph", in a syntactic way. The probability model of an "event And–Or graph" can be learned on a set of annotated event instances, and given a learned event And–Or graph, a Gibbs sampling scheme is utilized for inference on a testing video. In the experiments, we test events recognition performance of the proposed on both real indoor and outdoor videos and show quantitative recognition rate on the public LHI dataset.

1. Introduction

Video understanding is a hot research topic in recent years, with many applications, such as visual surveillance, video browsing and content-based video indexing. And its key task is to monitor/recognize events on-line or off-line. To achieve these tasks, a number of key issues, such as background modeling (Stauffer and Grimson, 1999), object tracking (Li et al., 2007), object detection/classification (Lin et al., 2007; Zhu and Mumford, 2007; Lin et al., 2007), illumination/occlusion problems (Haritaoglu et al., 2000), are well studied in computer vision research. However, a good event representation is still required for high-level meaningful event understanding, fully taking advantage of those object tracked and classified results.

In this paper, we aim to define a probabilistic attribute graph grammar that allows syntactic representation of complex spatio-temporal events common in real visual surveillance. This grammar model decomposes a semantic event into a composition of meaningful actions, called "event components", with a dictionary of spatio-temporal relations. Each event component is further divided into a number of atomic activities, called "event primitives". An specific semantic event is thus a configuration of event primitives with tempo-spatial relations, and it can be described by an "event parse graph". In order to incorporate semantic meaning of events, the "event components" are always labeled manually, while the "event primitives" and the corresponding tempo-spatial relations can be learned in a supervised way.

As shown in Fig. 1, one event " a waiting car is picking up a coming man" is divided into three event components, "car waiting", "picking up", and "moving away", with temporal constraints (sequential order). And the "picking up" component is further decomposed into two components, "man is approaching to the car" and "man is entering the car", with sequential order as well. Finally these event components can be explained by event primitives in the lowest level, such as "stop", "moving", "stay", and "death". These primitives can be computed via related tracked objects features (visibility, location, velocity), as shown in the bottom of Fig. 1. Besides, all possible configurations for a variable semantic event can be modeled into an "event And–Or graph", with the proposed attribute grammar representation (Fig. 2). The attribute

* Corresponding author. Address: School of Information Science and Technology, Beijing Institute of Technology, Beijing 100081, China. Tel.: +86 10 68912565; fax: +86 10 68911272.
E-mail addresses: linliang@ieee.org (L. Lin), hfgong@lotushill.org (H. Gong), lli@lotushill.com (L. Li), wangliang@jdl.ac.cn (L. Wang).

0167-8655/$ - see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.patrec.2008.02.023

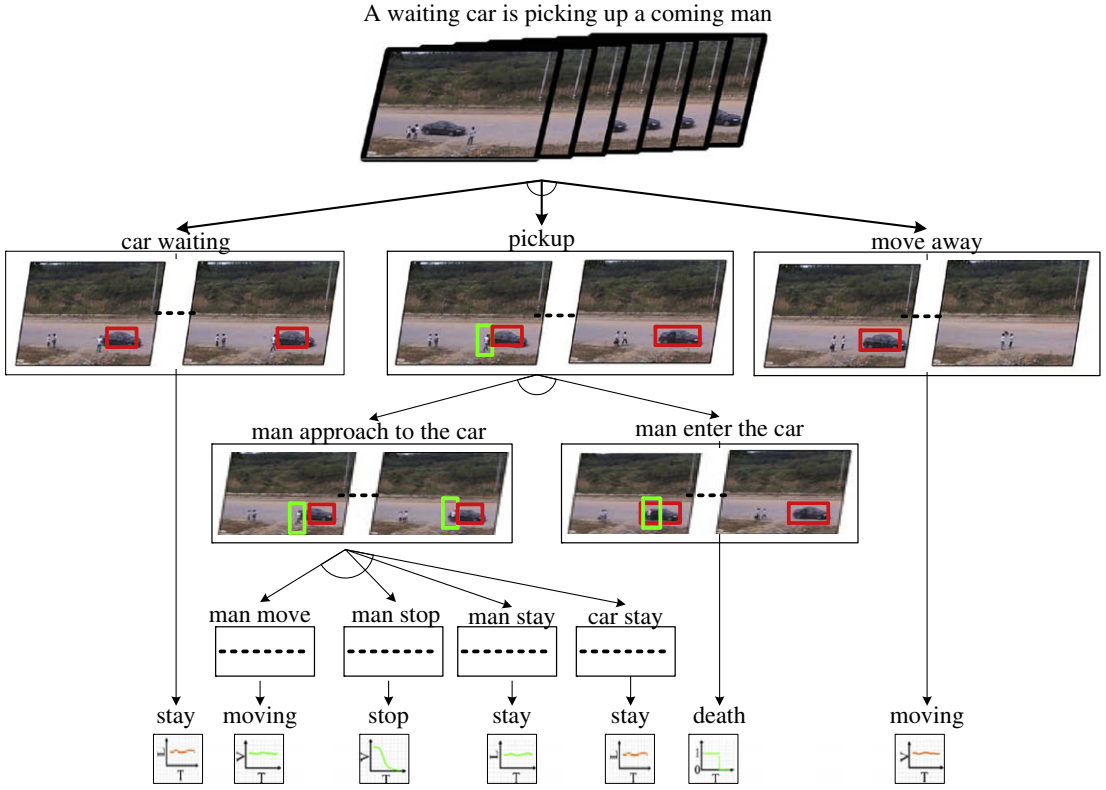

Fig. 1. Event parse graph to represent one specific semantic event. An event is divided into three event components, with temporal constraints (sequential order). And one of these components is further decomposed into two sub-components with sequential order as well. Finally these event components can be explained by event primitives in the lowest level, which can be computed via related tracked objects features (visibility, location, velocity), as shown in the bottom of this figure.

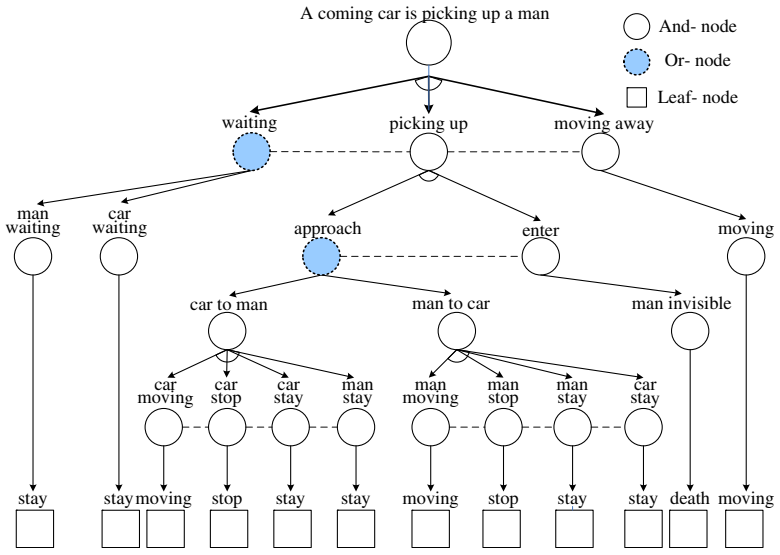

Fig. 2. Event And–Or graph to represent all possible variety of a semantic event. In this figure, an event "a coming car is picking up a man" is modeled with an event And–Or graph, in which the And-nodes and Or-nodes denote meaningful event components. Each And-node is compositional and Or-node is a choice. The And-nodes and Or-nodes (event components) can be decomposed into a set of Leaf-nodes (atomic event primitives) with temporal-spatial constraints.

grammar and event And–Or graph will be further studied in Section 3. Intuitively, one event parse graph is once instance derived from event And–Or graph, as well as one specific event from all possible variety.

The event representation we studied is embedded in an intelligent visual surveillance system, which comprises a motion detection module, a target tracking module, and an object classification module. In order to highlight the event representation and recognition, we assume that the good object trajectory and object type can be provided by the surveillance system. To extend application of event recognition, we can also describe events with "virtual objects", a region or a line specified by user in the scene. For example, to recognize the event "a man is turning over a wall", we can label a forbidden virtual region for the wall.

2. Related work

In the computer vision literature, there has been a significant amount of event understanding research in various application domains (Buxton and Gong, 1995; Haritaoglu et al., 2000; Medioni et al., 2001; Ivanov and Bobick, 2000; Bobick and Wilson, 1997; Collins et al., 2000; Xu and Chang, 2007).

The early research for event analysis started on model postures (e.g. "standing close to a car") or simple events (e.g. "sitting") from the visual evidence gathered during a short video sequence (Bobick and Wilson, 1997). Bayesian network and its variants are widely used for these approaches (Binder et al., 1997). Their main limitation is that they are not suitable for encoding the dynamic of long-term activities, due to discarding the temporal relations.

To represent temporal trajectories, Hidden Markov Models and their variants are adopted as the state-based representations, inspired by the applications in speech recognition. These approaches automatically learn the states and transition probabilities from event samples. For example, parameterized-HMMs (Wilson and Bobick, 1999) and coupled-HMMs (Oliver et al., 2000) were introduced to recognize a more complex event such as an interaction of two mobile objects. However, most of these approaches lack a multi-layer probabilistic model to combine each state in a syntactic way to represent meaningful events in high-level.

It is worth mentioning one remarkable work by Ivanov and Bobick (2000), which our approach is related to. This method aims at higher-level behavior and employs a two-layer event abstraction. At the lowest level, simple events similar to our defined event primitives are modeled by HMM, and a stochastic context free grammar (SCFG) is constructed for the problem domain with the simple events as terminals. However, this method and ours are different in many aspects. First, they define simple events using only trajectory of tracked object, while our event primitives are defined on more detailed attributes, such as velocity, visibility, and location. Second, they use temporal constraints to connect simple events without addressing spatial relations and they did not provide the parameterized definition of constraints for learning, resulting in most essential information are omitted. In contrast, we define the events probability model with spatial and temporal relations, respectively, which are learned from a set of annotated videos. Third, their applications are limited to simple event with single-agent, while our approach is able to solve more complex interactive events, such as "a car stopped and is picking up a waiting pedestrian".

In addition, the attribute graph grammar was first presented by Han and Zhu (2005), Chen et al. (2006), and Zhu and Mumford (2007) further discussed it as a large scale knowledge representation. We also show its success in object category recognition (Lin et al., 2007). In this work, we extensively study it on event representation and recognition, and show its applications in visual surveillance system.

The remainder of this paper is arranged as follows. We first present the event representation with attribute graph grammar in Section 3, including event primitive definition (Section 3.3), and spatio-temporal relations learning (Section 3.4). We then follow with a description of the inference method on a learned event representation in Section 4. The experiment results are shown in Section 5 and the paper is concluded in Section 6 with a summary.

3. Event representation with attribute graph grammar

An event representation needs to be able to represent a wide variety of events flexibly. In this section, we introduce the attribute graph grammar with the probabilistic models to represent semantic events with event components, event primitives, and corresponding constraints.

3.1. Attribute graph grammar

An attribute graph grammar is augmented from the stochastic context free grammar (SCFG) by incorporating attributes and constraints on the nodes. The attributes and constraints can be modeled by a markov random field (MRF).

An attribute graph grammar $\mathscr{G}$ is specified by a five-tuple

$$\mathscr{G} = (S, V_N, V_T, R, P) \tag{1}$$

where S is the root node and denotes one semantic compositional event including a number of semantic event components.

The non-terminal nodes $V_N = \{V_N^{\mathrm{AND}}, V_N^{\mathrm{OR}}\}$ contain a set of And-nodes and Or-nodes. Each Or-node V_i^{OR} has a distribution $p(\omega_i)$ over which of its $\omega = \{1, 2, \ldots, N(\omega_i)\}$ children it is expanded into. $V_T = \{t_1, t_2, \ldots, t_{T_n}\}$ represents the set of terminal nodes. In our model, the non-terminal nodes (And-nodes and Or-nodes) are meaningful event components, and the terminal nodes are atomic event primitives, which are defined on event attributes $\phi(t_i)$.

$R = \{r_1, r_2, \ldots, r_{N(R)}\}$ in the formulation represents the set of pairwise relations defined as functions over pairs of nodes $(v_i, v_j) \in V_T \cup V_N$, $r = \psi(v_i, v_j)$. Each relation is a temporal or spatial function between pair of nodes, for example the distance and angle between the centers of the two nodes. These relations are defined at all levels of the tree, and will be described in Section 3.3. The probability model P is defined explicitly and will be introduced in Section 3.2.

With this grammar representation, a class of semantic events, "a car picks up a man and leaves", can be represented as shown in Fig. 2. This event can be first decomposed into three event components "waiting", "picking up", and "moving away", with the temporal constraints. The "waiting" component is an Or-node, due to two possible cases, "one car waiting" and "one man waiting". The "picking up" component is an And-node and is composition of the "approach" component and the "enter" component, with the temporal and spatial constraints ("approach" occurs before "enter", and "enter" occurs close to "approach" in spatial). Finally, all event components can be explained by a set of parameterized event primitives, such as "stay", "moving" and "stop". In sum, this event And–Or graph models all possible variability of an event in a syntactic way.

3.2. Probability model learning

The probability model of an event And–Or graph contains the frequencies at the Or-nodes, the relations constraints at the And-nodes, and the attributes constraints at the Leaf-nodes (event primitives).

As discussed in (Zhu and Mumford, 2007), the And–Or graph probability model can be transformed to a tree structure with embedded constraints, following the composition of SCFG and MRF model, and the probability model of one semantic event can be learned from a set of event instances, labeled event parse graphs. In this sense, an parse graph is a valid traversal of an And–Or graph. Therefore, each event parse graph consists of the set of non-terminal nodes, $V = \{v_1, v_2, \ldots, v_{N(v)}\} \in V_N$, a set of resulting terminal nodes $T = \{t_1, t_2, \ldots, t_{N(t)}\} \in V_T$, and a set of relations observed between graph nodes, $R \in \mathscr{R}$.

The structural components of And–Or graph are in forms of a parse tree, and its prior model follows the product of all the switch variables ω_i at the Or-nodes visited, that is, ω_i denotes the index of the child node selected by Or-node V_i^{OR}. Following (Zhu and Mumford, 2007) and the SCFG model (Chi and Geman, 1998), we define the probability of the parse tree as

$$p(T) = \prod_{i \in V^{OR}} p_i(\omega_i) \tag{2}$$

Let $p(\omega_i)$ be the probability distribution over the switch variable ω_i at node V_i^{OR}. Suppose $p(\omega_{ij})$ is the probability that ω_i takes value j, and n_{ij} is the number of times we observe this production, $p(\omega_i)$ at node V_i^{OR} can be calculated as

$$p(\omega_i) = \prod_{j=1}^{N(\omega_i)} p(\omega_{ij})^{n_{ij}} \tag{3}$$

And according to the derivation in (Porway et al., 2007), $p(\omega_{ij})$ can be directly learned from a number of observations (the corresponding node in labeled event parse graphs), and Eq. (2) can be reformulated as

$$p(T) = \prod_{i \in V^{OR}} \prod_{j=1}^{N(\omega_i)} p(\omega_{ij})^{n_{ij}} \tag{4}$$

The MRF is defined as a probability on the configurations of the resulting parts of the parse tree. It can be written in terms of the pairwise energies (relations) between nodes, and singleton energies (event primitive attributes)

$$p(C) = \frac{1}{Z} \exp^{-\sum_{i \in T} \phi(t_i) - \sum_{\langle i,j \rangle \in V} \psi(v_i, v_j)} \tag{5}$$

where $\phi(t_i)$ denotes the singleton function corresponding to a singleton primitive attribute and $\psi(v_i, v_j)$ denotes the pairwise constraint corresponding to a pairwise relation. The constraint between non-terminal graph nodes will be computed on corresponding primitives finally, and thus $\psi(v_i, v_j) = \psi(t_i, t_j)$.

Following the deriving process of Porway et al. (2007), we obtain the final expression of $P = p(G, \theta)$. Suppose R_N^1 is the number of singleton constraints and R_N^2 is the number of pairwise constraints. Here we assume each event primitive is defined on one attribute (location, velocity, or visibility), and one pair of primitives is constrained by one temporal relation and one spatial relation. Therefore, $R_N^1 = 1$, $R_N^2 = 2$, and $\psi(t_i, t_j) = \{\psi^s(t_i, t_j), \psi^t(t_i, t_j)\}$.

$$p(G) = \frac{1}{Z} \exp\{-E(G)\} \tag{6}$$

$$\begin{aligned} E(G) &= \log(p(T)) + \sum_{i \in T} \sum_{a=1}^{R_N^1} \alpha_i^a \phi_i^a(t_i) + \sum_{\langle i,j \rangle \in V} \sum_{b=1}^{R_N^2} \beta_{ij}^b \psi^b(t_i, t_j) \\ &= \log(p(T)) + \sum_{i \in T} \alpha_i \phi_i(t_i) + \sum_{\langle i,j \rangle \in V} \beta_{ij}^s \psi^s(t_i, t_j) + \beta_{ij}^t \psi^t(t_i, t_j) \end{aligned} \tag{7}$$

where $\Theta = (\alpha, \beta)$ are related parameters of the probability model and can be learned from a few annotated parse graphs, as proved in (Porway et al., 2007). Intuitively, for a variable event, the basic structural components (event components and primitives) and corresponding relations are essential and finite, like the basic words and grammar rules, and thus can be learned from a few typical instances.

3.3. Event primitives

We define atomic event primitive via tracked blob trajectories, as shown in Fig. 3. This table can be the output from the visual surveillance system, and contains tracked objects type and three main attributes (IsVisible A_1, Location A_2, Velocity A_3) in the video sequence. In Fig. 3, each row in the trajectory table includes all objects' type and attributes in one frame, and a few frames (right) are illustrated, including tracking blobs.

Based on the trajectory table, we define 6 atomic event primitives $E_{p_{set}} = \{E_{p_1}, E_{p_2}, \ldots, E_{p_6}\}$, as terminal nodes in graph representation, $t_i \in E_{p_{set}}$. The event primitives with related blob attributes and descriptions are shown in Table 1.

Table 1
Event primitives defined on the trajectory table

Attributes	Primitives function (B)	Description
IsVisible	Death	Tracked blob B becomes invisible
	Birth	Tracked blob B becomes visible
Location	Moving	Tracked blob B is moving
	Stay	Tracked blob B stands by
Velocity	Start	Tracked blob B starts to move
	Stop	Tracked blob B stops

There are three attributes of tracked blobs in trajectory table, which are "IsVisible", "Location", "Velocity" shown in left column. Based on those attributes, the six atomic event primitives are defined in middle column, and their descriptions are shown right column.

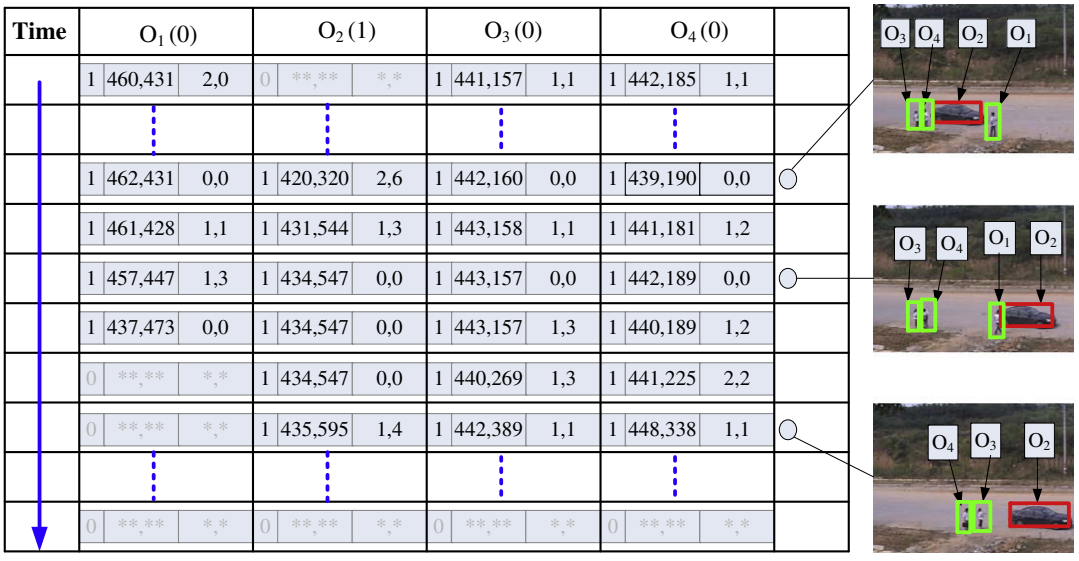

Fig. 3. Trajectory table including tracked objects attributes *Object* = ⟨*Type*, *Attributes*⟩. In the trajectory table header, *O*(*type*) denotes tracked blob and type denotes object category. Here we define "type" as 0 – pedestrian, 1 – car, 2 – bicycle, and 3 – others. Each cell in table denotes blob attributes in one frame, including IsVisible (0 – no, 1 – yes), Location (blob coordinates in frame), and Velocity (velocities in *X*, *Y*-axis). A few related video frames are illustrated in the right.

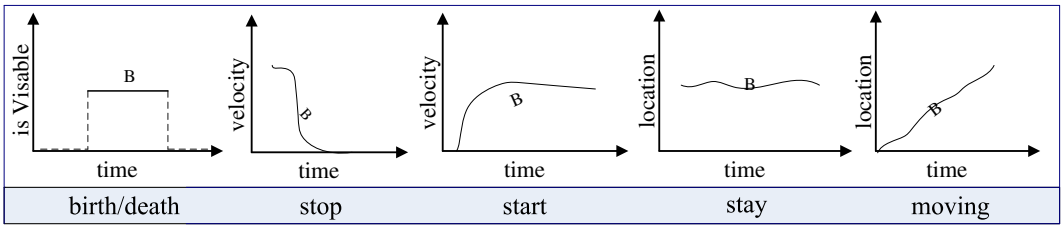

Fig. 4. The prior probability distribution of five event primitives. Each distribution measure the event primitive feature with related attributes of tracked blobs, which are "IsVisible", "Location", and "Velocity".

The prior model of those event primitives can be learned from a number of annotated videos. According to Table 1, each primitive is related to one attribute, and we count the attribute value in a short time period on a set of labeled video. And then compute average value distribution as prior model for each primitive, as shown in Fig. 4. Notice that each primitive is related with only one attribute, $\forall f(Ep_m, A_n), m \in 1, 2, \ldots, 6, n \in \{1, 2, 3\}$. Therefore, as in Eq. (5), the singleton function over each primitive $\phi(t_i)$ can be defined

$$\phi(t_i) = Distance(f(t_i|A_n), p(Ep_m|A_n)) \tag{8}$$

where t_i denotes each terminal node in graph, and is related to event primitive Ep_n with respect to attribute constraint $\phi(t_i)$. Notice that the attributes "Location" (A_2) and "Velocity" (A_3) are two dimensions, and it is straightforward that they need to be projected into one dimension when computing.

3.4. Spatio-temporal relations

The spatio-temporal constraints of events is critical to understanding to hence representing compositional events, and here we define 6 explicit temporal relations with prior histograms, and implicit spatial function to account for spatio-temporal constraints.

We assume the time cost of an event primitive as atomic time unit, for example, the time costs of two event primitives are $TI_p(t_i) = [start_i, end_i]$ and $TI_p(t_j) = [start_j, end_j]$. We thus define six explicit temporal relations based on Allen's interval algebra (Allen and Ferguson, 1994), and extend them to probabilistic form, as shown in Table 2. The logic deterministic descriptions to temporal relations is shown in the second column in Table 2, and the probabilistic definition can be found in third column. The prior distributions of temporal relations thus can be learned from a number of annotated video sequences in a supervised way, as shown in Fig. 5. Therefore, posterior temporal relation $\psi^t(t_i, t_j)$ over pair of nodes, t_i and t_j, can be sampled as follows:

$$\psi^t_m(t_i, t_j) \sim P(R_{tm}), \quad m \in \{1, 2, \ldots, 6\} \tag{9}$$

For spatial constraints, we define spatial relations over one pair of event primitives, based on distance and angle between them, as shown in Fig. 6

$$\psi^s(t_i, t_j) = F(d, \varphi) = \omega_1 D(t_i, t_j) + \omega_2 \varphi(t_i, t_j) \tag{10}$$

where $D(t_i, t_j)$ denotes Euclidean distance of two event primitives (center of tracked blob) and $\varphi(t_i, t_j)$ denotes related angle. ω_1 and ω_2 are set empirically ($\omega_1 = 0.85$ and $\omega_2 = 0.15$).

Table 2
Probabilistic temporal relations definition

Temporal relations (t_i,t_j)	Logic definition	Probabilistic definition
After	$end_i < start_j$	$P(R_{t1}) = P(start_j - end_i)$
Meets	$end_i = start_j$	$P(R_{t2}) = P(start_j - end_i)$
Overlap	$start_i < start_j < end_i$	$P(R_{t3}) = P(\frac{start_j - start_i}{end_i - start_i})$
During	$start_j > start_i$ and $end_j < end_i$	$P(R_{t4}) = P(\frac{start_j - start_i}{end_i - end_j})$
Starts	$start_i = start_j$	$P(R_{t5}) = P(start_j - start_i)$
Finish	$end_i = end_j$	$P(R_{t6}) = P(end_j - end_i)$

4. Event inference

Given a testing video sequence with objects tracking and identified, we should infer $P(G)$ in a learned event And–Or graphs to achieve an event recognition. In other words, we should search each vertex (terminal node) t_i in the video sequence so that the following probability (Eq. (6)) is maximized:

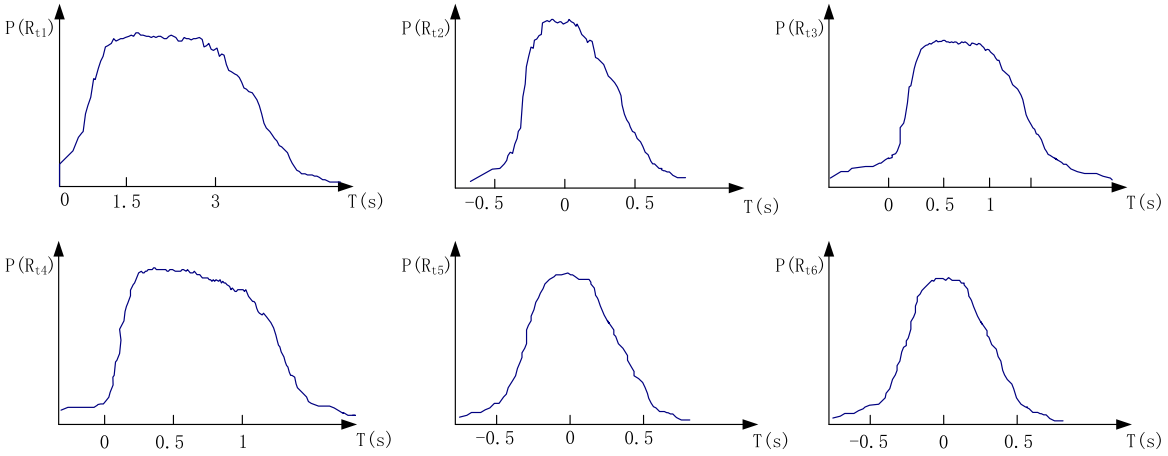

Fig. 5. Prior histogram of temporal relations.

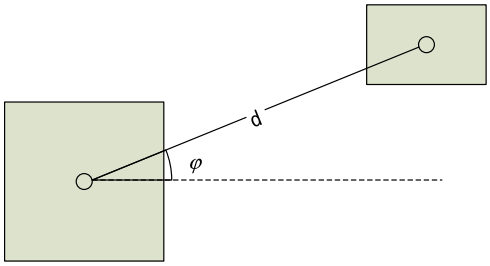

Fig. 6. Distance and angle between pair of event primitives.

$$p(G) = \frac{1}{Z}(\Theta)P(T) \times \exp\left\{\sum_{i\in T}\alpha_i^a\phi_i^a(t_i) + \sum_{\langle i,j\rangle\in V}\beta_{ij}^s\psi^s(t_i,t_j) + \beta_{ij}^t\psi^t(t_i,t_j)\right\} \quad (11)$$

where $P(T)$ is the prior term of graph structure and is defined on the frequencies of all Or-nodes (Eqs. (2) and (4)). Therefore, $P(T)$ can be computed directly when inference, and we should only sample each vertex following MRF model (single attributes constraint and pairwise relations constraint, as in Eq. (5)).

For each vertex, there will be several candidates by primitive events detection (setting a lower threshold for $\psi_i(v_i)$). Assume for an unknown vertex n_i, we have candidates set $CandSet_i = \{v_i\}_{j=1}^{|CandSet_i|}$, then the solution space contains $\prod_i|CandSet_i|$ solutions. It is a huge space, and we thus use Gibbs sampling to travel this space. The computing algorithm is as follows:

Initialization: For each node n_i, we initially set it as the primitive event v_i with the maximal $\phi_i^a(t_i)$.

Gibbs sampling: We sort the nodes by the number of candidates in ascending order, then use this order as the visiting order of Gibbs sampling to accelerate the inference.

For each node n_i with its neighbors $n_j \in N(i)$, we update it by sampling

$$p(n_i = t_i|n_j = t_j\forall j\neq i) \propto \exp\left\{-\alpha_i^a\phi_i^a(t_i) - \sum_{j\in N(i)}\beta_{ij}^s\psi_{i,j}(t_i,t_j) - \beta_{ij}^t\psi_{i,j}(t_i,t_j)\right\} \quad (12)$$

5. Experiments

The proposed event representation approach was embedded in an visual surveillance system, whose architecture is similar with Collins et al. (2000).

We test the semantic event recognition on public LHI dataset (Yao et al., 2007). We first learn parameters of event primitives

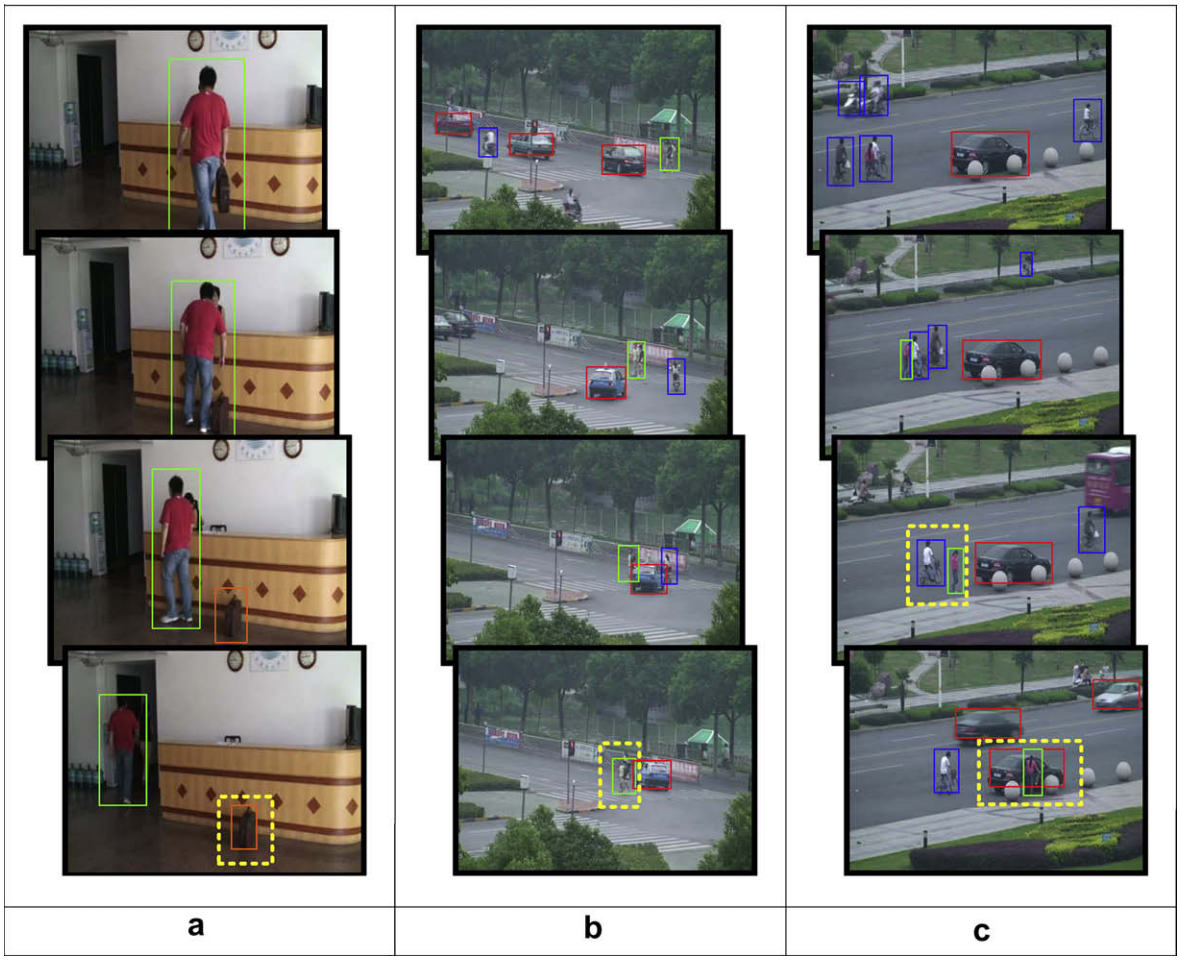

Fig. 7. Representative examples of event recognition. In this figure, different colors denote different tracked object categories (green – pedestrian, red – car, blue – bicycle/motorbike, golden – unknown object), and yellow dashed box indicates event happening. If this figure is not viewed by colors, note each solid box and dashed box indicate the identified object and event happening respectively. The event "a coming pedestrian left behind an unknown object (a suitcase)" is detected in (a), the event "a car is going across zebra line where a pedestrian crossing" is in (b), and the event "a bicycle is dropping off a pedestrian" and event " the pedestrian is entering a waiting car" are both detected in (c). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Event description	Recall Precision	False Alarm
A car stopped and dropping off a pedestrian	85%	11%
A coming pedestrian left behind a unknown object	90%	13%
A pedestrian is loitering in the scene	95%	11%
A car is going across a forbidden part	96%	3%
A pedestrian is turning over a wall	82%	7%

Fig. 8. Recall precision and false alarm on five semantic events recognition.

Event Description	The Proposed Approach		Flat HMM	
	Recall Precision	False Alarm	Recall Precision	False Alarm
A coming car is picking up a waiting pedestrian	83%	15%	79%	14%
A watched object is being stolen	92%	8%	87%	8%

Fig. 9. Comparisons with the flat HMM based method on two semantic events recognition.

and spatio-temporal relations on a number of (more than 200) common labeled event instances, as described in Sections 3.3 and 3.4. We then manually label specific event components for each semantic event using those learned event primitives, and the And–Or graph representations are built up, as described in Section 3.2. Based on these learned And–Or graph models, four representative events in both indoor and outdoor are parsed based on attribute grammar following in inference algorithm in Section 4, as shown in Fig. 7. The object category recognition is provided by surveillance system, and different colors in this figure denote different tracked object categories (green – pedestrian, red – car, blue – bicycle/motorbike, golden – unknown object), and yellow dashed box indicates event happening. The event "a coming pedestrian left behind an unknown object (a suitcase)" was in Fig. 7a, the event "a car is going across zebra line where a pedestrian crossing" is in Fig. 7b, and the event "a bicycle is dropping off a pedestrian" and event " the pedestrian is entering a waiting car" are both detected in Fig. 7c.

We show the quantitative results on recognition of 7 more events, and the testing set of each is of size 400 event instances (200 positive and 200 negative examples), and the lengths of event video sequences are from 200 to 500 s frames, depending on the event complexity. The recall precision and false alarm are shown in Figs. 8 and 9.

Experiments are also performed to compare the proposed method with the flat HMM model based method (Wilson and Bobick, 1999) on two selected events, and the result is shown in Fig. 9. The experiments are concerned on two events, namely, "A coming car is picking up a pedestrian" and "A watched object is being stolen". The proposed approach outperforms the HMM based approach in recall precision and comparable in false alarm rate, due to our approach accounts for events variations in both temporal domain and event primitives compositions. For example, in the event "A coming car is picking up a pedestrian", it can either be "the pedestrian stand still and then the car get near to the pedestrian" or "the car stop and then the pedestrian get near to the car". The events model structure thus need to be flexible and relations between the involved agents are modeled explicit, as our approach, while the HMM-based methods are often fixed events temporal structure without explicit relations definition.

6. Summary

In this paper, an attribute graph grammar is presented for event representation and recognition. With this representation, one specific event can be represented by an "event parse graph", and one category of variable event can be modeled with an "event And–Or graph". We also illustrate event inference algorithm given a learned event And–Or graph in a testing video. The experiments show the event recognition on both indoor and outdoor video, and quantitative results of recognition rate and false alarm on public LHI dataset (Yao et al., 2007) are also presented to validate the proposed approach.

Acknowledgements

This work is done at the Lotus Hill Institute and is supported by China 863 Program (Grant Nos. 2006AA01Z339, 2006AA01Z121, and 2006AA02Z4E5), NSFC (Grant No. 60673198). The data used in this paper were provided by the Lotus Hill Annotation Project (Yao et al., 2007).

References

Allen, J., Ferguson, G., 1994. Actions and events in interval temporal logic. J. Logic Comput. 4 (5), 531–579.
Binder, J., Koller, D., Russell, S., Kanazawa, K., 1997. Adaptive probabilistic networks with hidden variables. Machine Learn. 29, 213–244.
Bobick, A., Wilson, A., 1997. A state-based approach to the representation and recognition of gesture. IEEE Trans. PAMI 19 (12), 1325–1337.
Buxton, H., Gong, S., 1995. Visual surveillance in a dynamic and uncertain world. Artif. Intell. 78 (1), 431–459.
Chen, H., Xu, Z., Liu, Z., Zhu, S.C., 2006. Composite templates for cloth modeling and sketching. In: Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), vol. 1, pp. 943–950.
Chi, Z., Geman, S., 1998. Estimation of probabilistic context-free grammars. Comput. Linguist. 24 (2), 299–305.
Collins, R., Lipton, A., Kanade, T., Fujiyoshi, H., Duggins, D., Tsin, Y., Tolliver, D., Enomoto, N., Hasegawa, O., 2000. A System for Video Surveillance and Monitoring. Technical Report CMU-RI-TR-00-12. Robotics Institute, Carnegie Mellon University, May 2000.
Han, F., Zhu, S.C., 2005. Bottom-up/top-down image parsing by attribute graph grammar. In: Proc. of IEEE Internat. Conf. on Computer Vision (ICCV), vol. 2, pp. 17–21.
Haritaoglu, I., Harwood, D., Davis, L.S., 2000. W^4: Real-time surveillance of people and their activities. IEEE Trans. PAMI 22 (8), 809–830.
Ivanov, Y., Bobick, A., 2000. Recognition of visual activities and interactions by stochastic parsing. IEEE Trans. PAMI 22 (8), 852–872.
Li, Zheng, Gong, Haifeng, Zhu, Song-Chun, Sang, Nong, 2007. Dynamic feature cascade for multiple object tracking with trackability analysis. Energy minimization methods in computer vision and pattern recognition. Lecture Notes in Computer Science, vol. 4697. Springer. pp. 350–361.
Lin, L., Peng, S., Porway, J., Zhu, S.C., Wang, Y., 2007. An empirical study of object category recognition: Sequential testing with generalized samples. In: Proc. IEEE Internat. Conf. on Computer Vision (ICCV).
Lin, L., Zhu, S.C., Wang, Y., 2007. Layered graph match with graph editing. In: Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR).
Medioni, G., Cohen, I., Bremond, F., Hongeng, S., Nevatia, R., 2001. Event detection and analysis from video streams. IEEE Trans. PAMI 23 (8), 873–889.
Oliver, N.M., Rosario, B., Pentland, A.P., 2000. A Bayesian computer vision system for modeling human interactions. IEEE Trans. PAMI 22 (8), 831–843.
Porway, J. Yao, B., Zhu, S.C. 2007. Learning an And–Or Graph for Modeling and Recognizing Object Categories. Technical Report. Department of Statistics, UCLA.
Stauffer, C., Grimson, W.E.L., 1999. Adaptive background mixture models for real-time tracking. In: Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), vol. 2, pp. 244–252.
Wilson, A.D., Bobick, A.F., 1999. Parametric Hidden Markov Models for gesture recognition. IEEE Trans. PAMI 21 (9), 884–900.
Xu, D., Chang, S.F., 2007. Visual event recognition in news video using Kernel Methods with multi-level temporal alignment. In: Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR).
Yao, B., Yang, X., Zhu, S.C., 2007. Introduction to a large scale general purpose groundtruth dataset: Methodology, annotation tool, and benchmarks. Energy minimization methods in computer vision and pattern recognition. Lecture Notes in Computer Science, vol. 4697. Springer. pp. 169–183.
Zhu, S.C., Mumford, David, 2007. A stochastic grammar of images. Foundat. Trends Comput Graphics Vision 2 (4), 259–362.