Front. Electr. Electron. Eng. China 2010, 5(2): 137–142
DOI 10.1007/s11460-009-0078-y

RESEARCH ARTICLE

Xiaofeng WANG, Liang CHANG, Zhixin LI, Zhongzhi SHI

A dynamic description logic based system for video event detection

Abstract Video event detection is an important research area nowadays. Modeling the video event is a key problem in video event detection. In this paper, we combine dynamic description logic with linear time temporal logic to build a logic system for video event detection. The proposed logic system is named as $\mathrm{LTD}_{\mathrm{ALCO}}$ which can represent and inference the static, dynamic and temporal knowledge in one uniform logic system. Based on the $\mathrm{LTD}_{\mathrm{ALCO}}$, a framework for video event detection is proposed. The video event detection framework can automatically obtain the logic description of video content with the help of ontology-based computer vision techniques and detect the specified video event based on satisfiability checking on $\mathrm{LTD}_{\mathrm{ALCO}}$ formulas.

Keywords video event, semantics, dynamic description logics, reasoning, ontology

1 Introduction

With the flourishing digital video resources, efficient methods are required to manage the enormous video resources. Automatically managing video resources according to their content is regarded as one promising way for video resource management [1]. The main problem in content-based video resource management is how to automatically understand the content of the video. There is a direct way to automatically understand the content of the video which is by detecting the events that occur in the video and representing the video content according to the detected events. Here the events refer to the high level semantic concepts that humans perceive when observing videos. Generally, the events can be categorized into classes which are static event and dynamic event. Static event is the occurrence of static concepts in video, like a flower or a mountain, while dynamic event is related to actions, like take off, walking, fighting.

Video event detection is a difficult problem and there are some challenges. These challenges can be summarized as follows.

1) How to translate the low level visual information into high level semantic descriptions.

2) How to detect video events based on the semantic description of video content and background knowledge.

Many efforts in video event detection have been done to overcome these challenges [1–10]. According to the proposed methods, a typical video event detection system can be divided into two components which are information abstraction component and event modeling component. The information abstraction component extracts information units from the digital video while the event modeling component models the video events according to the event definitions. By incorporating the two components, a video event detection system can effectively discover events from the video sequences.

In video event detection system, the event modeling component is very important. As event modeling component bridges the gap between the event description and the information extracted from video sequences. Typically, the event modeling component can be classified into three categories. The first category is pattern recognition based event modeling, such as support vector machine (SVM) based event modeling [2,3]. The second category is state model based event modeling, such as Bayesian network based event modeling, conditional random fields based event modeling [4,5]. The third category is semantic model

Received August 10, 2009; accepted October 28, 2009

Xiaofeng WANG (✉), Zhixin LI, Zhongzhi SHI
The Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China
E-mail: wangxf@ics.ict.ac.cn

Xiaofeng WANG, Zhixin LI
Graduate University of Chinese Academy of Sciences, Beijing 100049, China

Liang CHANG
School of Computer and Control, Guilin University of Electronic Technology, Guilin 541004, China

based event modeling, such as Petri net based event modeling in video of traffic monitoring [6,7], Markov logic networks based event modeling [8] and constraint satisfaction based event modeling [9].

In this paper, a dynamic description logic (DDL) [11–13] based event modeling approach for dynamic event detection is proposed. Different from previous event modeling approaches, the DDL based event modeling approach models video event definition and video content description with DDL logic formulas. Moreover, it allows a video event detection system to discover the complex event by performing logic inference based on background knowledge represented by ontology and DDL logic formulas extracted from video sequences.

2 Linear time temporal dynamic description logic

Traditional description logics can only represent and reason about static knowledge [14]. They lack the ability in dealing with dynamic knowledge. To overcome this shortcoming, a new description logic system named DDL is proposed [11]. DDL introduces dynamic dimension into the description logic, which allows the representation and inference of dynamic knowledge. Moreover, with DDL, the inference that is incorporated with static and dynamic knowledge can be performed in one uniform logic system. The logic system $\mathrm{LTD_{ALCO}}$ is an extension of dynamic description logic system D-ALCO [12]. Comparing with D-ALCO, $\mathrm{LTD_{ALCO}}$ supports the temporal knowledge representation and reasoning. The details about D-ALCO can be found in Ref. [12]. In this paper, we only focus on $\mathrm{LTD_{ALCO}}$.

2.1 Syntax of $\mathrm{LTD_{ALCO}}$

In $\mathrm{LTD_{ALCO}}$, action is the basic element for dynamic knowledge representation. Action includes atomic action and complex action. Atomic action can be represented as

$$\alpha(v_1,v_2,...,v_n) \equiv (P_\alpha,E_\alpha),$$

where α is the name of action, $(v_1,v_2,\ldots,v_n)$ are names of individuals that occur in P_α and E_α , P_α is a set of logic formulas that specify precondition of execution of α, E_α is a set of logic formulas that specify the effect of α after execution. Based on atomic action, the complex action can be defined according to the following rule:

$$\pi,\pi' ::= \alpha(v_1,v_2,...,v_n)|\pi;\pi'. \quad (1)$$

In the rule for actions definition, ";" is named as sequential connection symbol. $\pi;\pi'$ is named as sequential action that represents the sequential execution of π and π'.

Example 1 There are two atomic actions. One is

$$\mathrm{orderBook}(u,v) \equiv$$
$$\{P_{\mathrm{order}} = (\mathrm{customer}(u),\mathrm{book}(v),\neg\mathrm{order}(u,v)),$$
$$E_{\mathrm{order}} = (\mathrm{order}(u,v))\},$$

and the other one is

$$\mathrm{payBook}(u,v) \equiv$$
$$\{P_{\mathrm{pay}} = (\mathrm{customer}(u),\mathrm{book}(v),\mathrm{order}(u,v),\neg\mathrm{payed}(u,v)),$$
$$E_{\mathrm{pay}} = (\mathrm{payed}(u,v))\}.$$

Based on orderBook(u,v) and payBook(u,v), a complex action orderBook(u,v);payBook(u,v) can be defined. orderBook(u,v);payBook(u,v) represents sequential action, which is ordering the book and then paying for the ordered book sequentially.

There are two kinds of formulas in $\mathrm{LTD_{ALCO}}$. One is basic formula and the other is complex formula. The complex formula is formed by connecting basic formulas with formula connection symbols $\wedge$, $\vee$, $\rightarrow$.

Definition 1 The basic formula of $\mathrm{LTD_{ALCO}}$ includes individual assertion formula, temporal formula and action formula. They can be constructed according to the following rules.

1) The individual assertion formula in $\mathrm{LTD_{ALCO}}$ is constructed according to the following rule:

$$\varphi_{\mathrm{ind}},\psi_{\mathrm{ind}} ::= C(u)|R(u,v)|\neg\varphi_{\mathrm{ind}}|\varphi_{\mathrm{ind}} \vee \psi_{\mathrm{ind}}, \quad (2)$$

where u, v represent individuals, C is a concept, R is a role. Moreover, the formula that is in form of $\varphi_{\mathrm{ind}} \wedge \psi_{\mathrm{ind}}$ can also be introduced. The formula $\varphi_{\mathrm{ind}} \wedge \psi_{\mathrm{ind}}$ is equivalent to $\neg\varphi_{\mathrm{ind}} \vee \neg\psi_{\mathrm{ind}}$.

2) The temporal formula in $\mathrm{LTD_{ALCO}}$ is constructed according to following rule:

$$\varphi_{\mathrm{t}},\psi_{\mathrm{t}} ::= \varphi_{\mathrm{ind}} \mathrm{U}^{\pi} \psi_{\mathrm{ind}}|\neg\varphi_{\mathrm{t}}, \quad (3)$$

where φ_{ind}, ψ_{ind} are the individual assertion formulas, π is a sequential action, U is equivalent to the "until" operator in linear time temporal logic, $\varphi_{\mathrm{ind}}\mathrm{U}^{\pi}\psi_{\mathrm{ind}}$ represents linear time temporal logic formula $\varphi_{\mathrm{ind}}\mathrm{U}\psi_{\mathrm{ind}}$ is hold on the execution trajectory of π. Moreover, the temporal formula in the form of $\Diamond\varphi$, $\Box\psi$ can also be introduced. They are equivalent to $\mathrm{true}\mathrm{U}^{\pi}\varphi$ and $\neg\Diamond\neg\psi$, respectively.

3) The action formula in $\mathrm{LTD_{ALCO}}$ is constructed according to the following rule:

$$\varphi_a ::= \langle\pi\rangle\varphi_{\mathrm{ind}}|\langle\pi\rangle\varphi_a|\langle\pi\rangle\varphi_{\mathrm{t}}|\neg\varphi_a, \quad (4)$$

where φ_{ind} is the individual assertion formula. The formula that is in the form of $[\pi]\varphi$ can also be introduced. It is equivalent to $\neg\langle\pi\rangle\neg\varphi$.

Definition 2 The complex formula in $\mathrm{LTD_{ALCO}}$ is constructed according to the following rule:

$$\varphi,\psi ::= \varphi_{\mathrm{ind}}|\varphi_{\mathrm{t}}|\varphi_a|\varphi \vee \psi. \quad (5)$$

In addition, the formulas that are in forms of $\varphi \wedge \psi$, $\varphi \rightarrow \psi$, true and false can also be introduced into $\mathrm{LTD_{ALCO}}$. They are equivalent to $\neg(\neg\varphi \vee \neg\psi)$, $\neg\varphi \vee \psi$, $\varphi \vee \neg\varphi$, and $\varphi \wedge \neg\varphi$ respectively. The formulas that are in the forms of $R(u,v)$, $\neg R(u,v)$, $C(u)$, $\neg C(u)$ are called simple formulas. A set containing limited individual assertion formulas is called an ABox of $\mathrm{LTD_{ALCO}}$.

2.2 Semantics of $\mathrm{LTD_{ALCO}}$

According to the definition of action, the state of ABox will be changed after execution of action. In order to embody the change of state of ABox, a space containing multiple possible worlds is employed to represent the semantic model of $\mathrm{LTD_{ALCO}}$. In the semantic model of $\mathrm{LTD_{ALCO}}$, each possible world corresponds to a state of ABox and is associated with a unique interpretation function. The interpretation function maps the role name, concept name and individual name into binary relation, set, and individual of the domain respectively. Every action is interpreted as binary relation on possible worlds which embodies the state change of ABox after action execution.

Definition 3 $\mathrm{LTD_{ALCO}}$ model is a triple $M = (\Sigma,\Delta,I)$, where

1) Σ is a framework that is in the form of $\Sigma = (W,T_{\alpha_0},T_{\alpha_1},\ldots)$. W is the set of possible worlds and each atomic action α is mapped to a binary relation $T_\alpha \subseteq W \times W$;

2) Δ is the domain of discourse;

3) I is the interpretation function. For each possible world w, $I(w) = (\Delta, \cdot I(w))$ maps each concept name C to a set $C^{I(W)} \subseteq \Delta$, each role name R to a binary relation $R^{I(W)} \subseteq \Delta \times \Delta$ and each individual name to an individual of Δ.

Because the temporal formula of $\mathrm{LTD_{ALCO}}$ is based on execution trajectory of action, the execution trajectory needs to be defined before the definition of the semantics on $\mathrm{LTD_{ALCO}}$ model.

Definition 4 For a $\mathrm{LTD_{ALCO}}$ model $M = (\Sigma,\Delta,I)$, an execution trajectory τ in M is a limited sequence $(w_1, w_2,\ldots,w_n)$, and τ corresponds to a sequential action $(\alpha_1, \alpha_2,\ldots,\alpha_{n-1})$, where for each sub sequence (w_i,w_{i+1}) $(1 \leqslant i < n)$ of τ, $(w_i,w_{i+1}) \in T_{\alpha_i}$.

Two trajectories can be connected to form a new trajectory. The Definition of trajectory connection is described as follows.

Definition 5 For a $\mathrm{LTD_{ALCO}}$ model $M = (\Sigma,\Delta,I)$, τ_1 and τ_2 are two execution trajectories. $|\tau_1|$, $|\tau_2|$ are the length of τ_1 and τ_2, respectively. If $\tau_1[|\tau_1|] = \tau_2[1]$, the trajectory connection of τ_1 and τ_2 is $\tau_1 \cdot \tau_2 := (\tau_1[1],\tau_1[2], \ldots,\tau_1[|\tau_1|],\tau_2[2],\ldots,\tau_2[|\tau_2|])$; If $\tau_1[|\tau_1|] \neq \tau_2[1]$, the result of trajectory connection of τ_1 and τ_2 is null.

In Definition 5, the length of an execution trajectory is the number of possible worlds in the trajectory.

Definition 6 Let $M = (\Sigma = (W,T_{\alpha_0},T_{\alpha_1},\ldots),\Delta, I)$ be a model of $\mathrm{LTD_{ALCO}}$,

1) For each possible world $w_i \in W$, $I(w_i)$ maps each concept name C to a set $C^{I(w_i)} \subseteq \Delta$, each role name R to a binary relation $R^{I(w_i)} \subseteq \Delta \times \Delta$ and each individual name p to an individual of $p^I \in \Delta$.

2) For each formula in M,

$(M,w_i) \vDash C(u)$ iff $u^I \in C^{I(w_i)}$;

$(M,w_i) \vDash R(u,v)$ iff $(u^I,v^I) \in R^{I(w_i)}$;

$(M,w_i) \vDash \neg\varphi$ iff $(M,w_i) \nvDash \varphi$;

$(M,w_i) \vDash \varphi \vee \psi$ iff $(M,w_i) \vDash \varphi$ or $(M,w_i) \vDash \psi$;

$(M,w_i) \vDash \langle\pi\rangle\varphi$ iff there exists a possible world $w' \in W$ which satisfies $(w,w') \in T_\pi$ and $(M,w') \vDash \varphi$;

$(M,w_i) \vDash \varphi \mathrm{U}^\pi \psi$ iff τ is the execution trajectory of action π, where $\tau[1] = w_i$ and $\exists j(j \geqslant 1) \wedge \forall k(1 \leqslant k < j)$ that $(M,\tau[k]) \vDash \varphi \wedge (M,\tau[k]) \vDash \neg\psi \wedge (M,\tau[j]) \vDash \neg\varphi \wedge (M,\tau[j]) \vDash \psi$ is hold.

3) Each action π is mapped to a binary relation $T_\pi \subseteq W \times W$. The sequential action is defined by connection of execution trajectory of sub action

$$T_{\pi;\pi'} := \{\tau \cdot \tau' \mid \tau \in T_\pi \wedge \tau' \in T_{\pi'}\}.$$

3 $\mathrm{LTD_{ALCO}}$ based event detection system

In order to detect the video event, the $\mathrm{LTD_{ALCO}}$ event detection system uses ontology-based video analysis techniques to parse the video and produce $\mathrm{LTD_{ALCO}}$ logic formula sequence to describe the content of the video. Then, by utilizing $\mathrm{LTD_{ALCO}}$ logic reasoning techniques, the event detection system can detect the event from logic formula sequence.

Figure 1 demonstrates the process of video event detection in an $\mathrm{LTD_{ALCO}}$ based event detection system. For a video clip v, n key frames are extracted from v. For each extracted key frame k_i, image segmentation techniques are applied to k_i in order to separate k_i into different parts. For each part, the object recognition techniques are applied to recognize the objects in that part [10,15]. Then, a set of logic formulas to describe the content of k_i is produced according to the result of object recognition. Meanwhile, the object tracking techniques are employed to recognize the same object in different key frames. Finally, a logic set sequence is obtained. Then, logic set sequence is put into an $\mathrm{LTD_{ALCO}}$ based reasoning engine to detect the specified video events from logic set sequence.

As shown in Fig. 1, by video analysis techniques, a logic formula set sequence $\mathrm{Seq}_v = \{s_1,s_2,\ldots,s_n\}$ is produced to describe the content of video clip v. Each logic formula set $s_i \in \mathrm{Seq}_v$ corresponds to a logic description of the content of a key frame in v.

The reasoning engine demonstrated in Fig. 2 includes:

1) A TBox that contains definitions of concepts;

2) An ABox that contains logic formula sequences produced by video analysis process;

3) An ActBox that contains the definitions of atomic actions and complex actions;

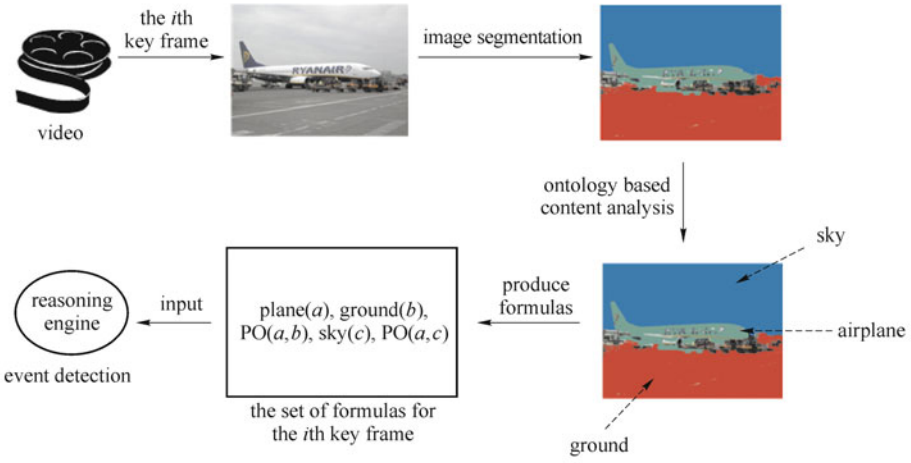

Fig. 1 Process of event detection

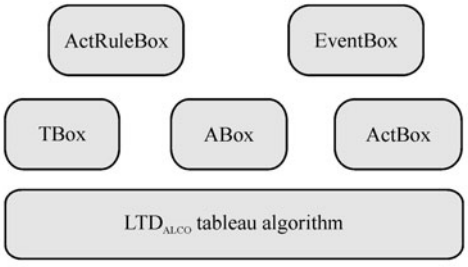

Fig. 2 Reasoning engine for event detection

4) An ActRuleBox that contains a set of rules that specify the characters of the execution trajectories of actions;

5) An EventBox that contains the rules about video events.

The rule in ActRuleBox is in the form of $\varphi U\psi \rightarrow \langle\pi\rangle$true, which means that if a possible word sequence Seq satisfies temporal formula $\varphi U\psi$, then Seq contains an execution trajectory of π. The event rule in EventBox is in the form of $\langle\alpha_1\rangle$true $\wedge$ $\langle\alpha_2\rangle$true $\wedge \cdots \wedge \langle\alpha_n\rangle$true $\rightarrow e$, which means if the actions $\alpha_1,\alpha_2,\ldots,\alpha_n$ occur, then the event e can be detected in the video.

To detect dynamic event, the actions that compose dynamic event should be detected from the video first. Thus, we present an algorithm to detect the actions that occur in the video based on logic set sequence.

Algorithm 1 Let ActionSet be the set of actions that are the consequents of rules in ActRuleBox. $Seq_v=\{s_1, s_2,\ldots, s_n\}$ is the logic formula set sequence obtained from video clip v. AR is the ActRuleBox.

```
Begin
  a: ForEach α_i ∈ ActionSet do
     {
       For i = 1 to n − 1 do
       {
         Let f_{α_i} := Conj(s_i) → ⟨α_i⟩true;
         if (¬f_{α_i} is unsatisfiable) then
         {
           ForEach φUψ → α_i ∈ AR do
           {
             Let π_i be an action and its execution
             trajectory is {s_i, s_{i+1},…,s_n};
             if (¬(φU^{π_i}ψ) is unsatisfiable) then
             {
              add α_i to resultSet;
              goto a;
             }
           }
         }
       }
     }
  Return resultSet;
End
```

For each key frame description s_i and an action α, Algorithm 1 checks whether s_i can satisfy the precondition of action α. If it satisfies the precondition of α, then the algorithm checks whether the sub sequence $Seq_{sub}=\{s_i, s_{i+1},\ldots,s_n\}$ implies the temporal logic formula $\varphi U\psi$, where $\varphi U\psi \rightarrow \alpha_i \in$ AR. If $Seq_{sub} \models \varphi U\psi$, then α occurs in v.

The unsatisfiability of $\neg(\text{Conj}(s_i) \rightarrow \langle\alpha_i\rangle\text{true})$ ensures that s_i satisfies the precondition of α. The unsatisfiability of $\neg(\varphi U^{\pi_i}\psi)$ ensures that Seq_v implies $\varphi U\psi$. Furthermore, if Seq_v implies $\varphi U\psi$, then it can be inferred that α occurs in video clip v. The unsatisfiability of $\neg(\text{Conj}(s_i) \rightarrow \langle\alpha_i\rangle\text{true})$

and $\neg(\varphi U^{\pi_i}\psi)$ can be determined by tableau algorithm of LTD_{ALCO}. Moreover, it can be proved that if $\neg(\varphi U^{\pi_i}\psi)$ is unsatisfied, then $Seq_v \models \varphi U^{\pi_i}\psi$.

4 Case study

By Algorithm 1, the action occurring in the video clip can be detected. Based on the detected actions, event detection system can discover dynamic event from the video. In order to demonstrate the process of Algorithm 1, an example of event detection with LTD_{ALCO} based event detection system is given.

Because region connection calculus 8 (RCC8) [16] is used to describe the spacial relation between different objects in a key frame, a brief introduction to RCC8 is given first. The spatial relations that RCC8 can represent are shown in Fig. 3 where DC(A,B) represents A is disconnected with B; EC(A,B) represents A is externally connected with B; PO(A,B) represents A is partially overlapping with B; TPP(A,B) represents A is a tangential proper part of B; $TPP^{-}(A,B)$ is the inverse of TPP(A,B); NTPP(A,B) represents A is non-tangential proper part of B; $NTPP^{-}(A,B)$ is the inverse of NTPP(A,B); EQ(A,B) represents A is equal to B.

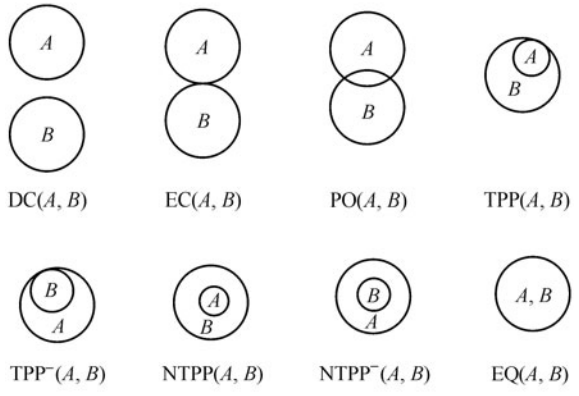

Fig. 3 RCC 8

Suppose we want to detect the take-off of an airplane in video, thus knowledge related to take-off event are put into the reasoning engine. In ActBox, an action named $\alpha_{take\text{-}off}$ is defined. The rule $plane(a) \wedge ground(b) \wedge sky(c) \wedge (PO(a,b)UPO(a,c)) \rightarrow \langle\alpha_{take\text{-}off}\rangle true$ is added to ActBox. In EventBox, take-off event is defined as $\langle\alpha_{take\text{-}off}\rangle true \rightarrow Event_{take\text{-}off}$ which means if action $\alpha_{take\text{-}off}$ occurs in a video, then take-off event occurs in the video.

Let v' be a video clip and $seq_{v'}=\{s_1,s_2,\ldots,s_m\}$ be the logic formula set sequence of v'. According to the Algorithm 1, $plane(a) \wedge ground(b) \wedge sky(c) \wedge \neg(PO(a,b)UPO(a,c))$ is unsatisfiable, thus take-off event is detected in video v'.

5 Conclusions

In this paper, we propose a dynamic description logic system LTD_{ALCO} that can support linear time temporal logic reasoning. Moreover, an LTD_{ALCO} based video event detection system is presented for video event detection. The LTD_{ALCO} based video event detection system detects the executed action by logic reasoning and discover the event that is related to the detected action. Compared with previous works, our work is a knowledge based method for video event detection, which is similar to the way by which humans perceive the video event. Thus, our work is promising and can detect more high level semantic related video events.

Acknowledgements This work was supported by the National Natural Science Foundation of China (Grant Nos. 60933004, 60903141, 60903079, 60775030 and 60775035), the National Basic Research Program of China (No. 2007CB311004), National High Technology Research and Development Program of China (No. 2007AA01Z132), and the National Science and Technology Pillar Program (No. 2006BAC08B06).

References

1. Smoliar S W, Zhang H J. Content-based video indexing and retrieval. IEEE MultiMedia, 1994, 1(2): 62–72
2. Pittore M, Basso C, Verri A. Representing and recognizing visual dynamic events with support vector machines. In: Proceedings of the 10th International Conference on Image Analysis and Processing. 1999, 18–23
3. Piciarelli C, Foresti G, Snidaro L. Trajectory clustering and its applications for video surveillance. In: Proceedings of IEEE Conference on Advanced Video and Signal Based Surveillance. 2005, 40–45
4. Hongeng S, Nevatia R, Bremond F. Video-based event recognition: activity representation and probabilistic recognition methods. Computer Vision and Image Understanding, 2004, 96(2): 129–162
5. Sminchisescu C, Kanaujia A, Li Z, Metaxas D. Conditional models for contextual human motion recognition. In: Proceedings of the Tenth IEEE International Conference on Computer Vision. 2005, 2: 1808–1815
6. Ghanem N, DeMenthon D, Doermann D, Davis L. Representation and recognition of events in surveillance video using Petri nets. In: Proceedings of 2004 Conference on Computer Vision and Pattern Recognition Workshop. 2004, 112
7. Ghanem N M. Petri net models for event recognition in surveillance video. Dissertation for the Doctoral Degree. Maryland: University of Maryland, 2007
8. Tran S D, Davis L S. Event modeling and recognition using Markov logic networks. In: Proceedings of the 10th European Conference on Computer Vision. 2008, 610–623
9. Fusier F, Valentin V, Brémond F, Thonnat M, Borg M, Thirde D, Ferryman J. Video understanding for complex activity recognition. Machine Vision and Applications, 2007, 18(3–4): 167–188
10. Snoek C G M, Huurnink B, Hollink L, de Rijke M, Schreiber G,

Worring M. Adding semantics to detectors for video retrieval. IEEE Transactions on Multimedia, 2007, 9(5): 975–986

11. Shi Z Z, Dong M K, Jiang Y C, Zhang H J. A logic foundation for the semantic Web. Science in China. Series F (Information Sciences), 2005, 48(2): 161–178 (in Chinese)
12. Chang L, Shi Z Z, Qiu L R, Lin F. A tableau decision algorithm for dynamic description logic. Chinese Journal of Computers, 2008, 31 (6): 896–909 (in Chinese)
13. Shi Z Z, Chang L. Reasoning about semantic web services with an approach based on dynamic description logics. Chinese Journal of Computers, 2008, 31(9): 1599–1611 (in Chinese)
14. Baader F, Calvanese D, McGuinness D, Nardi D, Patel-Schneider P. The Description Logic Handbook: Theory, Implementation and Applications. Cambridge: Cambridge University Press, 2002
15. Maillot N E, Thonnat M. Ontology based complex object recognition. Image and Vision Computing, 2008, 26(1): 102–113
16. Randell D A, Cui Z, Cohn A G. A spatial logic based on regions and connection. In: Proceedings of the 3rd International Conference on Knowledge Representation and Reasoning. 1992, 165–176