

Understanding Video Events: A Survey of Methods for Automatic Interpretation of Semantic Occurrences in Video

Gal Lavee, Ehud Rivlin, and Michael Rudzsky

Abstract—Understanding video events, i.e., the translation of low-level content in video sequences into high-level semantic concepts, is a research topic that has received much interest in recent years. Important applications of this paper include smart surveillance systems, semantic video database indexing, and interactive systems. This technology can be applied to several video domains including airport terminal, parking lot, traffic, subway stations, aerial surveillance, and sign language data. In this paper, we identify the two main components of the event understanding process: abstraction and event modeling. Abstraction is the process of molding the data into informative units to be used as input to the event model. Due to space restrictions, we will limit the discussion on the topic of abstraction. See the study by Lavee *et al.* (Understanding video events: A survey of methods for automatic interpretation of semantic occurrences in video, Technion—Israel Inst. Technol., Haifa, Israel, Tech. Rep. CIS-2009-06, 2009) for a more complete discussion. Event modeling is devoted to describing events of interest formally and enabling recognition of these events as they occur in the video sequence. Event modeling can be further decomposed in the categories of pattern-recognition methods, state event models, and semantic event models. In this survey, we discuss this proposed taxonomy of the literature, offer a unifying terminology, and discuss popular event modeling formalisms (e.g., hidden Markov model) and their use in video event understanding using extensive examples from the literature. Finally, we consider the application domain of video event understanding in light of the proposed taxonomy, and propose future directions for research in this field.

Index Terms—Action, activity, behavior, event, recognition, video.

I. INTRODUCTION

VIDEO events are those high-level semantic concepts that humans perceive when observing a video sequence. Video event understanding attempts to offer solutions to the problem of reconciling this human perception of events with a computer perception. The major challenge in this research area is translating low-level input into a semantically meaningful event description.

Video event understanding is the highest level task in computer vision. It relies on sufficient solutions to many lower level tasks such as edge detection, optical flow estimation, object recognition, object classification, and tracking. The maturity of

Manuscript received May 19, 2008; revised November 16, 2008 and February 19, 2009. First published June 16, 2009; Current version published August 19, 2009. This paper was recommended by Associate Editor E. Trucco.

The authors are with the Department of Computer Science, Technion—Israel Institute of Technology, Haifa 32000, Israel (e-mail: gallavee@cs.technion.ac.il; ehudr@cs.technion.ac.il; rudzsky@cs.technion.ac.il).

Digital Object Identifier 10.1109/TSMCC.2009.2023380

many solutions to these low-level problems has spurred additional interest in utilizing them for higher level tasks such as video event understanding.

Another reason for the large amount of interest in video event understanding is the promise of intelligent systems outfitted with inexpensive cameras enabling such applications as active intelligent surveillance, summarization and indexing of video data, unobtrusive homecare for the elderly, and hands-free human-computer interaction. This interest is exhibited by the amount of research projects approved in this domain including CARE-TAKER [2], ETISEO [3], AVITRACK [4], ADVISOR [5], BEWARE [6], ICONS [7], VSAM [8], and many others.

The problem of video event understanding is still a challenging one for several reasons including noise and uncertainty in the output of low-level computer vision tasks such as object detection and tracking, large variance in the appearance of particular events, similarity in the appearance of different events, and ambiguity in translating semantic (high-level) definitions of events into a formalism for representation and recognition. The main questions in the field of event understanding are as follows.

- 1) How can the meaningful and discriminating aspects of the video sequence input be extracted?
- 2) How can the events of interest be represented and recognized?

The goal of this paper is to organize the methods used in this research domain such that their precise role becomes apparent.

To achieve this, we have divided the broad research domain of video event understanding into categories. We have grouped together approaches to solving the first question before in a category called abstraction. Approaches to answer the second question aim to find a suitable formalism to both describe interesting events in the input video sequence and allow recognizing these events when they occur. These approaches are grouped together in the category of event modeling.

Both abstraction and event modeling are processes of mapping low-level to high-level information. However, we distinguish abstraction from event modeling in that abstraction molds the data into informative primitive units to be used as input to the event model. The event model may then consider spatial, compositional, temporal, logical, and other types of relationships between these primitives in defining the structure of the event, i.e., abstraction and event modeling are two parts of the same process.

Abstraction schemes and event models are chosen with respect to the event domain. Approaches to represent and

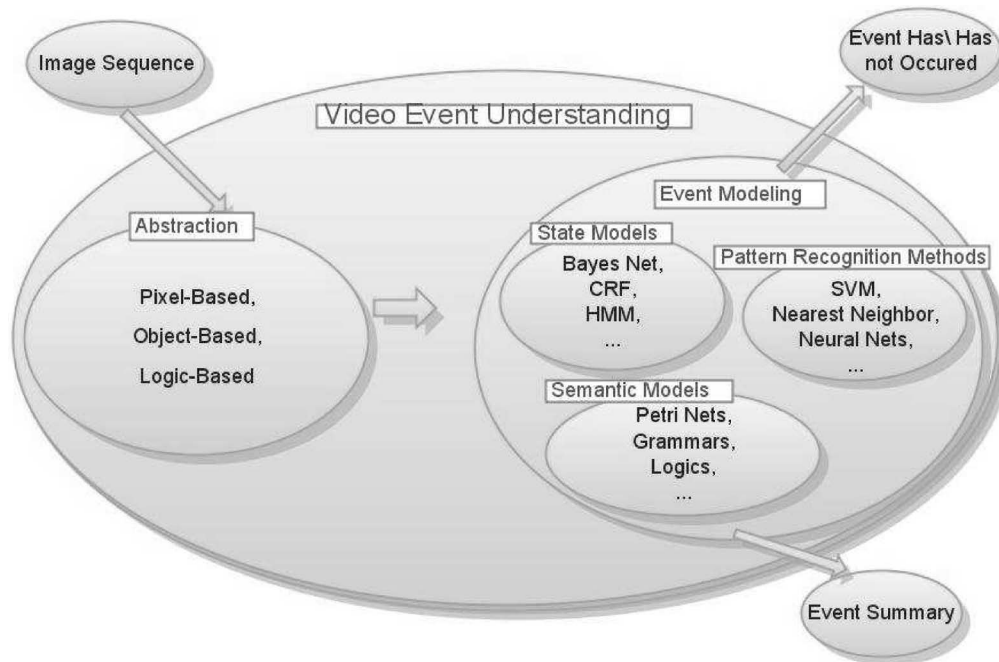


Fig. 1. Bird's eye view of the video event understanding domain. A video event understanding process takes an image sequence as input and abstracts it into meaningful units. The result of the abstraction is used by the event model to determine if an event of interest has occurred. Output of a video event understanding process may be a decision on whether a particular event has occurred or a summary of events in the input sequence.

recognize relatively simple events (single actor, known camera angle, presegmented event sequences) may identify a discriminating abstraction scheme and utilize a pattern-recognition method for event recognition. More involved events (multiple subevents, numerous actors, temporal–spatial relationships) may abstract the video sequence as a set of objects and use a semantic event model to represent and recognize the events of interest.

There have been several previous efforts to survey this area of research [9]–[12]. These papers describe only a subset of the ideas considered here and often consider video event understanding as a subarea of a related field.

The remainder of this paper is organized as follows: Section II discusses the ambiguous terminology in event understanding literature throughout which synonymous or similar terms appear with different meanings. This section also proposes and motivates the terminology used in the remainder of the paper. Later sections discuss the parts of the event understanding process. As illustrated in Fig. 1, we consider the two main parts of this process to be abstraction and event modeling. Abstraction is the problem of translating video sequence inputs into intermediate units understandable by event models. Section II offers a limited (due to space constraints) description of abstraction approaches.

Event modeling is the subdomain of event understanding devoted to describing events of interest formally and determining, using inputs from the abstraction layer, whether such an event has occurred in a particular video sequence. Event modeling approaches have received a lot of attention from the research community, and thus, will make up a large part of the discussion in this paper. Section IV discusses the domain of event modeling and offers insight into how approaches within this

domain may be categorized. Subsequently, Sections V–VII discuss the three main approaches to event modeling prevalent in the literature: pattern-recognition methods, state models, and semantic models. Each of these sections discusses the specific formalisms associated with each category of event modeling.

In Section VIII, we provide an alternative view of the event understanding domain of applications. In this view, it is more straightforward to observe which techniques of abstraction and event modeling are often utilized together and identify the most explored event domains in this research area. Finally, we conclude the survey in Section IX.

II. WHAT IS AN EVENT?

In this paper, we survey a large volume of works under the claim that they all pertain to video events. However, terms appearing throughout the literature, such as “behavior,” “activity,” “action,” “scenario,” “gesture,” and “primitive/complex event” are frequently used to describe essentially the same concepts. This is due to the fact that these concepts have an inherently ambiguous definition in language. In this section, our aim is to disambiguate these terms and propose a uniform terminology that we will use to describe specific works throughout this paper.

The term “event” is used in many disciplines including mathematics, physics, and philosophy as well as in the terminology of culture and technology. The meaning of this term in these areas varies and does not correspond directly to the concept that we wish to address in this paper.

We also attempted to categorize occurrences in a video sequence. Nagel [13] defined a semantic hierarchy consisting of the concepts: “change,” “event,” “verb,” and “history”.

Bobick [14] set out to differentiate between the terms “movement,” “activity,” and “action” based on the knowledge required to classify each of these types of occurrence. By Bobick’s definition, an example of a “movement” is a hand wave. An “activity” might consist of a person walking. An “action” is a yet higher semantic level than an “activity.” An instance of an “action” might be walking into a store. Bremond [15] categorizes events into three main categories, each with increasing complexity: those composed of a single state (primitive events), those composed of a sequential series of states (sequential events), and those defined by complex semantic relationships (composite events).

Unfortunately, no consensus on an event terminology has been reached, and each new work not only uses, but also defines, its own set of terms. This is exemplified by different terms being used to describe roughly similar concepts. For instance, Hu *et al.* [12] refer to “object behaviors,” which is defined as time-varying data. Aggarwal and Cai [9] define “activity/behavior” as a pattern derived from an image sequence. Cohn *et al.* [16] define “dynamic behavior” as a set of spatial states whose relationships are defined by temporal logic. Hongeng and Nevatia [17] describe “simple events” as actions occurring in a linear time sequence.

The compositional and often hierarchical nature of events is a major cause of ambiguity. Unfortunately, different granularities of events are not described by the same terms from paper to paper. In fact, the same terms are often used to describe different levels of the compositional hierarchy.

In reading the various works, some commonalities between the “event” definitions do emerge.

- 1) Events occupy a period of time.
- 2) Events are built of smaller semantic unit building blocks.
- 3) Events are described using the salient aspects of the video sequence input.

We then define a “general event” as an object that possesses these common qualities. A particular event, in which each of these qualities has been explicitly instantiated, is defined by the needs of the application. The various combinations of possible instantiations for each of these qualities, in combination with the fact that these instantiations are often implicit, is the cause of the prevalent ambiguous terminology discussed before.

We propose a terminology that allows expression of the common qualities of events and eliminates the ambiguity of terms currently in use. Our terminology is inspired by Hongeng and Nevatia’s [17] “simple/composite event,” i.e., prefixes will be attached to the term “event” to describe its various properties, thus eliminating the need to define different terms to refer to types of events with varying properties. We will assign prefixes to represent each of the commonly agreed upon properties of events: subevent composition, and content and temporal composition. Additional prefixes are defined to allow different semantic units (e.g., subevents, superevents) to be related to an event.

A “particular event” (henceforth referred to as an “event”), in our terminology, is defined as an occurrence of interest in a video sequence. This term is equivalent to other terms in the literature including “behavior” [12], “activity” [18], “action” [19], and “gesture” [20].

The term “subevent” will be used to refer to component parts of the event. This parallels terms such as “poses” [20], “actions,” and “mobile event properties” [17], which provide the building blocks of the event. Subevents may also be composed of subevents in a recursive definition.

Complementary to subevents, superevents are those semantic units composed by the event(s). This corresponds to the terms “episodes” [10] and “scenarios” [17]. As with subevents, superevents may be defined recursively.

To describe the event composition, we define the terms “composite event” to indicate compositional structure (composed of subevents) and “atomic event” to indicate no subevent composition. This definition is in line with Nagel’s [13] description of the semantic hierarchy as well as Bremond’s categorization of events [15].

The difference between a composite event and a superevent is that the former is of interest in the event model while the latter is only relevant in its relationship to the event.

With regards to temporal composition, we adopt from Hongeng and Nevatia [17] the terms “single-thread event” and “multithread event” to describe the linear sequence and nonlinear temporal composition, respectively.

Content prefixes refer to the abstraction primitives used to describe an event. We define the term “object-based event” to indicate events modeled using methods such as object detection and tracking. Similarly, the term “pixel-based event” describes events modeled using pixel features such as color, texture, or gradient.

Our terminology departs from previous categorizations by Bobick [14] by allowing extensions describing compositional, temporal, and content aspects of the event as well as reducing the reliance on semantically ambiguous terms such as “activity,” “action,” and “behavior.”

Terms such as “activity,” “gesture,” and “behavior” carry some context about the particular event domain that they each represent. Replacing these terms with the term “event” loses this context. However, it does allow for uniting several works in the literature that while not working within the same event domain do, in fact, apply the same methods, specifically abstraction and event modeling choices. To address this issue, we will introduce another term associated with an event, the “event domain.” The event domain can be a natural language description of precisely what kind of events we are trying to recognize (e.g., gestures in an interactive environment). Defining this event domain will allow us to retain the context carried by the original ambiguous terminology, while disambiguating our objective (i.e., the particular type of event we wish to represent and recognize). In fact, defining the event domain empowers us to give even more context than was originally available in the terminology.

Another concern about our terminology is its application-dependent nature. An “occurrence of interest in a video sequence” certainly depends on the application in question. However, this problem also exists in current terminologies. Furthermore, our goals of unifying approaches to event understanding (abstraction and event modeling) across numerous event domains are independent of these concerns.

TABLE I
EVENT TERMINOLOGY

Atomic (event)	Has no Sub-event composition
Composite (event)	Has Sub-Event composition

Content Prefixes

Pixel-Based (event)	Described by Pixel-Level primitives (e.g. color, texture, gradient)
Object-Based (event)	Described by Object-Level primitives (e.g. size, shape, trajectory)

Temporal Prefixes

Single-Thread (event)	Has Sequential Temporal relationships between Sub-Events
Multi-Thread (event)	Has Non-Sequential Temporal relationships between Sub-Events

Relation to Event of Interest Prefixes

Sub (event)	Component of Event
Super (event)	Composed of Event

This proposed terminology is used in subsequent sections of this survey. The terminology is summarized in Table I.

III. ABSTRACTION

Abstraction is the organization of low-level inputs into various constructs (sometimes called “primitives”) representing the abstract properties of the video data. The motivation for abstraction is to provide an intermediary representation of the video sequence. Although not all papers in the literature focus on abstraction, each work must make decisions on how the low-level input will be presented to the event model (e.g., which features will be used?, will a tracker be applied?). These decisions constitute the abstraction phase (see Fig. 1) and are an integral part of the event understanding process.

Pixel-based abstraction utilizes abstraction schemes that rely on pixel or pixel group features such as color, texture, and gradient. Some notable examples of pixel-based abstraction include gradient histograms [21] and motion history images [20].

Object-based abstraction is an alternative approach based on the intuition that a description of the objects participating in the video sequence is a good intermediate representation for event reasoning. Thus, the low-level input is abstracted into a set of objects and their properties. These properties include speed, position, and trajectory. Examples of object-based abstractions such as bounding boxes and blobs can be found throughout the literature [17], [22]–[25]. Silhouettes are another popular object-based abstraction used by many event understanding works [26], [27].

Another type of abstraction that we have dubbed from logic-based abstraction is motivated by the observation that the world is not described by multidimensional parameterizations of pixel distributions, or even a set of semantic objects and their properties, but rather by a set of semantic rules and concepts, which act upon units of knowledge. Thus, it aims to abstract low-level

input into statements of semantic knowledge (i.e., assertions) that can be reasoned on by a rule-based event model. Examples of logic-based abstraction can be found in [28] and [29].

Due to space restrictions, we will conclude our discussion on abstraction. For expanded discussion and further examples, see [1].

IV. EVENT MODELING

Event modeling is the complementary problem to abstraction (discussed in the previous section). This discipline seeks formal ways to describe and recognize events in a particular domain given the choice of an abstraction scheme. A particular event model is chosen based on both its capacity for representation of salient events in a particular domain and its capacity for recognition of these events as they occur in the video sequence input.

Event models have many aspects, and, hence, they may be categorized into several different ways. One such categorization is the distinction between deterministic and probabilistic event models. Another distinction that can be drawn is between generative and discriminative models. Some event modeling formalisms can be expressed graphically. (For more in-depth discussion of these categories, see [1].)

The aforementioned categorizations are useful in many ways, but they do not fully capture the diversity of event modeling approaches in the event understanding literature. For this reason, this paper proposes a categorization that the authors feel best allows us to describe the domain of event modeling.

We have chosen to organize event models into three different categories: “pattern-recognition methods,” “state models,” and “semantic models.” These are related in some ways to the model categories discussed before, but there is not a direct one-to-one relationship. Not every event modeling approach in the literature necessarily falls exclusively into one of these categories, but we believe they represent the general spirit of approaches within recent event modeling research. This categorization is closely related to the categorization of event models by temporal relationship complexity proposed in [15].

“Pattern-recognition methods” do not generally address the event representation aspect of event modeling and approach the event recognition problem as a traditional pattern recognition/classification problem. Accordingly, traditional approaches to these problems such as support vector machines, neural networks, nearest neighbor classifiers, etc., are applied to the abstraction scheme. Minimal semantic knowledge is needed in building the event classifiers in this category. Often, they may be fully specified from training data.

More recent approaches attempt to model video events using semantic information. We have named the first class of these models as “state models” for the reason that they concentrate on specifying the state space of the model. Often, this state space is reduced or factorized using semantic knowledge. This class of approaches includes finite-state machines (FSMs) and the set of probabilistic graphical model approaches. The existence (under some structural assumptions) of efficient algorithms for the learning of parameters from training as well as recognition motivates the choice of these models to model video events.

Higher level semantics include subevent ordering information (including partial ordering), and complex temporal, spatial, and logical relations among subevents. Also useful is the ability to express and recognize partial events. These properties become important when the event domain includes high-level events, which are best expressed in qualitative terms and natural language. To this end, a group of modeling formalisms that we have named “semantic models” have been proposed, which enable explicit specification of these complex semantic properties. Among these are Petri nets (PNs) and grammar models as well as constraint satisfaction and logic-based approaches. These models are usually fully specified using domain knowledge and are not usually learned from training data.

In the following sections, we will take a more in-depth look at the three categories of event modeling, and explore the various formalisms contained within each category with examples from the literature. Particularly, we will discuss the representational strengths and recognition efficiency of each category. We will also provide discussion on the event types and domains typically modeled by the approaches in each of our categories.

V. PATTERN-RECOGNITION METHODS FOR EVENT RECOGNITION

The class of techniques in this section is not quite event models, in the sense that they do not consider the problem of event representation. Instead, they focus on the event recognition problem, which is formulated as a traditional pattern-recognition problem.

The main advantage of the classifiers in this category is that they are well understood. Usually, they may be fully specified from a set of training data. These approaches are usually simple and straightforward to implement. This simplicity is afforded by excluding semantics (i.e., high-level knowledge about the event domain) entirely from the specification of the classifier.

There are many examples of pattern-recognition methods for event recognition in the literature including [20], [21], [26], [30], and [31] (nearest neighbor), [32]–[36] (SVM), [37]–[42] (boosting), and [43] (neural networks). A more in-depth discussion of these approaches and examples can be found in [1].

VI. STATE EVENT MODELS

“State” event models are a class of formalisms that model the state of the video event in space and time using semantic knowledge. Semantic knowledge about different aspects of the nature of video events have lead to the development of each of these formalisms. Therefore, each of the state event modeling formalisms captures an important aspect or property of video events.

State event models improve on pattern-recognition methods in that they intrinsically model the structure of the state space of the event domain. This modeling capacity generally increases the ability of these event models to represent different types of events, even when coupled with a simple abstraction scheme.

Modeling formalisms in this category are also well studied and mathematically well formulated. This allows for efficient al-

gorithms and sound formulations of problems such as parameter learning and event recognition.

In most, but not all, cases, the semantic information associated with the model structure makes this structure difficult to learn from training data. However, once the model structure is specified, model parameters can often be learned from the training data. This aspect of state models contributes to their popularity allowing them to combine human intuition about the event structure (semantics) and machine learning techniques.

State modeling formalisms include FSMs, Bayesian networks (BNs), hidden Markov models (HMMs), dynamic BNs (DBNs), and conditional random fields (CRFs).

A. Finite-State Machines

An FSM [44], also known as finite-state automata, is a well-studied formalism useful for modeling the temporal aspects of video events. This formalism extends a state transition diagram with start and accept states to allow recognition of processes. FSMs are traditionally deterministic models and provide a computationally efficient solution to the event recognition problem.

The strengths of the FSM model are in its ability to model sequential aspects of video events, its model simplicity, and its ability to be learned from training data. The FSM formalism is also well understood, which allows for straightforward analysis of running time complexity.

FSMs appearing in the literature naturally model single-thread events formed by a sequence of states. Event domains for which FSM event models are utilized include hand gestures [45], single-actor actions [46], multiple person interaction [17], and aerial surveillance [47].

The inherent ability of the FSM formalism to capture sequence allows it to be associated with different abstraction types including pixel-based [45] and object-based abstractions [17], [47]–[49].

The FSM assumption of a fully observable state is not present in other state event modeling formalisms. It allows the event recognition problem to be reduced to accepting/rejecting the process representing the event. Additionally, because all states, input symbols, and state transitions are fully observable, an FSM model may be learned from training data [50]. Some work has also been done on inferring an FSM event model from user specification [48], [49].

Extensions to the FSM have been proposed to capture the hierarchal property of video events [46], [51], [52]. Uncertainty in video events has also been addressed through the introduction of probabilities into the FSM framework [52]. It should be noted that in some areas of the event understanding literature, the terms “HMMs” (see Section VI-C) and “probabilistic FSMs” are used interchangeably. The main distinction is that FSMs assume a fully observable state while HMMs assume a hidden state variable.

These extensions to the FSM formalism have attempted to introduce aspects such as hierarchy and uncertainty. These methods have largely been applied to specific event domains and have not been embraced as general solutions. This is largely because

of the availability of other formalisms that are well adapted to such aspects (e.g., the HMM for uncertainty).

B. Bayesian Networks

In order to deal with the inherent uncertainty of observations and interpretation that exists in video events, event models utilizing probability have been proposed.

One such event modeling formalism is the BN. BNs are a powerful tool in factorizing the state space into variables using semantic knowledge of the domain. This formalism naturally models the hierarchical nature of video events. The probabilistic output of BNs is useful for addressing uncertainty.

More formally, BNs (also known as probabilistic networks, Bayesian belief networks, or independence diagrams) are a class of directed acyclic graphical models. Nodes in the BN represent random variables. Conditional independence assumptions between these variables are represented by the structure of the graph. The structure of the BN allows specification of the joint probability over all variables in a succinct form with few parameters, using the notion of conditional independence. For further details, see [53] and [54].

Having such an expression of the joint probability allows us to reason about any node in the network using known values. Often, BN event models will model the event as an unknown or “hidden” binary variable (event has/has not occurred) and the observations (abstraction primitives) as known variables. The BN structure (nodes and arcs) and parameters (conditional and prior probabilities) can be used to estimate the distribution of unknown variables given the value of known variables. This estimation can be done efficiently under certain assumptions about the BN structure [54]. Furthermore, efficient algorithms also exist for learning the BN parameters from data. Learning of the BN structure has also been explored [55].

An example with illustration of the standard approach to event modeling using BN is given in [1].

Naive BNs often appear throughout the event understanding literature. This model is sometimes called an “agent” architecture because several Bayesian “agents” are applied to objects of interest within the video sequence input. This structure is also well adapted to the hierarchical composition inherent to many video events. This is because the probability output of the top node in a subevent network can be easily integrated as an “observation” node in a higher level event model.

Agent architectures have been used in event domains such as aerial surveillance [56], [57] and indoor surveillance of people [58]. More complex BNs have been used in event domains such as parking lot surveillance [59] and recognizing American football plays [60]. Although these networks are large, they retain a structure that allows for efficient inference. The BN agent architecture is most often associated with an object-based abstraction.

Modeling the hierarchy of video events is straightforward within the BN framework. Hongeng *et al.* [61] model the semantic hierarchy using BN layers. Each layer corresponds to a higher level semantic units.

A more recent group [62]–[64] of works make use of BN models adapted from the text and image mining communities. These approaches are also known as probabilistic latent semantic analysis (pLSA) [65] and consider variables representing documents, words, and topics, which, in the event understanding domain, correspond to video sequences, abstraction primitives, and events, respectively. These types of approaches are most often associated with pixel-based abstractions (e.g., “cuboids” [62], [66]).

One major drawback of BN event models is that they do not have an inherent capacity for modeling temporal composition, which is an important aspect of video events. Solutions to this problem include single-frame event classification [56] and choosing abstraction schemes that encapsulate temporal properties of the low-level input [58], [60].

C. Hidden Markov Models

The benefits of a temporal evolution model (like FSM) and a probabilistic model (like BN) are combined within the framework of the hidden Markov model (HMM), i.e., event model. Due to these abilities, along with efficient algorithms for event recognition and parameter learning, HMMs have become one of the most popular formalisms for modeling video events.

HMMs are a class of directed graphical models extended to model the temporal evolution of the state. The classic HMM has a specific graph structure associated with it. This structure describes a model where the current observations are dependent only on the current state. The current state is only dependent upon the state at the previous “time slice” (the Markov assumption).

Since the HMM structure is fixed and repetitive, we can define the likelihood of long sequence of states (and corresponding observations) by specifying a set of parameters. The number of parameters required to specify these probabilities depends on the number of possible states and observation symbols.

There exist well-studied polynomial (in the number of hidden states) time algorithms for evaluation, inference, and learning in HMMs. For further details regarding HMMs, see [67] and [68].

A common use of HMMs in modeling video events is as follows. An HMM event model is defined by observation symbols related to the chosen abstraction scheme. The states of the HMM are usually abstract and their number is chosen empirically. The parameters of the HMM model may be learned from training data or specified manually using knowledge of the event domain. To discriminate between events, such an HMM event model is trained for each event under consideration. Test examples are then evaluated to determine how likely they are to have been generated by each of the HMM models. The event model that yields the highest likelihood score is used to label the test example [69].

A number of early works in the literature employ this approach in the event domains of tennis stroke recognition [70], sign language and gesture recognition [71], [72], and single-person actions (e.g., “walking” and “kneeling”) [73]. The events recognized in these works are mostly a few seconds in length. Furthermore, these methods are generally dependent on

adequate segmentation of the video sequence into event clips, i.e., before we can classify the event in a given video sequence, we must be given a clip known to contain an event (and only one event).

In more recent work, the HMM model has been extended in several ways to adapt to the challenges of modeling video events. One such challenge is the representation of the state and observation spaces within one variable, respectively. As the number of states and observations grow, this representation requires a great deal of parameters to be estimated and therefore a large set of training data (often larger than what is available). To deal with this challenge, solutions factorize the observation space into multiple variables or alter the network topology.

Multiobservation HMMs (MOHMM) [19] use multiple variables to represent the observation. This model reduces the number of parameters to be learned, and thus, makes parameter estimation from a finite set of training data more likely to produce good results.

Another approach to reducing the parameters to be learned is altering the network topology (specifically which states are reachable from which other states) [74]. For certain events, those composed of an ordered sequence of states, a fully connected transition model has unnecessary parameters. An HMM topology, which only allows transitions from one state to the next state in the sequence (without skipping states), would greatly reduce the number of parameters (all parameters not fitting these constraints would be set to zero). This kind of topology is called a casual or left–right HMM (with no-skip constraint).

Often, the event would be more naturally (from a semantic perspective) modeled with two or more state variables, forming state chains over time. Factorizing the state space into these multiple state chains is another way to simplify the event model. These multiple chains could correspond to simultaneous subevents in a composite event or multiple objects interacting within an atomic object-based event. Of course, some way of merging the output likelihoods of these chains while taking into account the dependencies between them is needed.

Several event models with variations on this approach exist. In parallel HMMs (PaHMMs) [75], the multiple chains of the state are modeled as separate HMMs, each with its own observation sequence. In coupled HMMs (CHMMs) [76]–[78], the hidden process chains are coupled in such a way that the current state in a particular chain depends on the previous state of all chains. Dynamic multilevel HMMs (DML-HMMs) [79] extend the coupling concept by attempting to learn the dependencies between the hidden state chains, i.e., the state space is reduced by both separating the state into multiple variables and simplifying the network structure.

As expected, these extensions are used in event domains where there are several elements participating in the events of interest including sign language [75], Tai-Chi gestures [76], multiple person interactions [78], and airport tarmac surveillance [79].

The multiple chains also allow a relaxation of the linear temporal order of the states, i.e., more complex temporal relationships between the state sequences can be modeled in this

way. An experimental comparison of MOHMMs, PaHMMs, CHMMs, and DML-HMMs can be found in [19].

Another extension to the basic HMM structure is motivated by the long-term temporal dependence of state variables within a video event, i.e., the Markov assumption that the current state depends only on the state at a previous time is not necessarily valid. The reason for this may be inherent long-term dependencies or occlusions and other phenomena that cause errors in state estimation.

N -order HMMs deal with this problem by amending the Markov assumption to consider N previous states. Variable-length Markov models (VLMMs) [80], [81] calculate the optimal level of temporal dependence using a divergence criterion. Hidden semi-Markov models (HSMMs) (sometimes called semi-HMMs) [82] allow each state to emit multiple observations, i.e., along with the state variable at each time, there will also be a duration variable (observation length).

Several HMM extensions have been proposed to incorporate the inherent hierarchical composition of video events into the event models. In hierarchical Markov models (HHMMs) [83], [84], each possible state is represented by a lower level HMM. In a similar approach, Oliver *et al.* [85] use a layered HMM (LHMM) event model in the event domain of office surveillance.

Several efforts have been made to integrate the various classes of extensions to the HMM event model into a single formalism, i.e., an event model that models long-term temporal dependence, hierarchical composition, and factorization of the state space into multiple variables. These include the switching hidden semi-Markov model (S-HSMM) [86], hierarchical semi-parallel HMMs (HSPaMMs) [87], and the coupled HSMMs (CHSMMs) [88]. Some of these “hybrid” HMMs are illustrated in [1].

One drawback to the various extensions of the HMM is that as the model topology becomes more complex, the efficient exact algorithms associated with the “classic” HMM structure are no longer applicable and must be replaced by approximation algorithms.

Figures visualizing the various HMM extension topologies are available in [1].

D. Dynamic BNs

As we have seen in the previous section, event models in some cases benefit from a meaningful factorization of the state and observation space. An event modeling formalism that allows such general factorization while still capturing the temporal evolution of state is the DBN.

DBNs generalize BNs (BN) with a temporal extent. HMMs are a special case of DBNs in which the structure is restricted to provide efficient algorithms for learning and inference. All HMM variants previously discussed are also special cases of the DBN. The strength of the general DBN in comparison to HMM is its ability to factorize the state space of the model in a semantically meaningful way. This factorization can enhance the performance of event classification. This, however, often comes at the cost of computational tractability. Approximation techniques are usually used to perform learning and inference

on general DBNs. Thus, DBNs in their general form appear less often as event modeling formalism in the literature.

To overcome the computationally hard inference in DBNs, many of the works in the literature make simplifying assumptions such as restricting the temporal links in the graph [89] and restricting state transition topology [90]–[92].

Because of the rich information about the structure of the event contained in the event model, a relatively simple pixel-based abstraction scheme is coupled with many DBN event models [91]–[94].

DBN approaches have been applied in event domains such as the office environment [93], assisted living [90]–[92], and surveillance of people [89], [94].

Recent work has attempted to learn aspects of the DBN model such as model structure [93], abstraction scheme [95], and the number of states each variable takes on [94].

DBNs in their general form appear less often as event modeling formalism in the literature. Special constrained cases of the DBN (most notably the HMM), however, are quite popular as event models throughout the event understanding community.

E. Conditional Random Fields

One drawback of generative models, in general, and HMMs, in particular, is their dependence on the availability of a prior on the observations (abstraction primitives). This prior is not always known and frequently estimated using assumptions that will yield efficient computation, such as independence between observations given the state (*a la* HMM). In the domain of video events, this is often an invalid assumption. In a discriminative statistical framework, only the conditional distribution is sought (modeled), and there is no need for such restrictive assumptions. The adoption of CRFs as event models is based on this idea.

CRFs, recently introduced in [96], are undirected graphical models that generalize the HMM. Existing known algorithms for HMM problems of inference and evaluation can be extended to CRFs. Learning of CRF parameters can be achieved using convex optimization methods such as conjugate gradient descent [97].

In video event modeling, CRFs have consistently been shown to outperform HMMs for similar event recognition tasks [98], [99]. This is attributed to the ability to choose an arbitrarily dependent abstraction scheme. Furthermore, in a CRF, unlike in the HMM, abstraction feature selection does not have to be limited to the current observation, but can also consider any combination of past and future observations. A major disadvantage of CRF models in comparison to HMMs is their parameter learning time. The optimization procedures like conjugate gradient descent take a significantly longer time than the training of HMMs.

Several more recent works have attempted to introduce additional structure into the CRF formalism using knowledge of the event domain [100]–[102]. These extensions to the original CRF structure to better capture some inherent properties of the event domain are similar to those extensions for HMMs discussed in Section VI-C.

VII. SEMANTIC EVENT MODELS

While many events can be described as a sequence of a number of states, an interesting subset of events are those defined by the semantic relationships between their composing subevents. The category of “semantic event models” groups several event modeling formalism to allow these kinds of relationships to be represented and recognized.

The class of semantic event models contains event modeling approaches that do not aim to define the entire state space of the event domain as in “state model” approaches. Instead, semantic knowledge is still used to define a set semantic rules, constraints, and relations, i.e., there is a large degree of overlap between how humans describe what constitutes an event and how it is defined within these modeling formalisms. Recognizing an event as it occurs becomes a problem of “explaining” the observation using the available semantic knowledge.

This type of approach allows the event model to capture high-level semantics such as long-term temporal dependence, hierarchy, partial ordering, concurrency, and complex relations among subevents, and abstraction primitives. Additionally, “incomplete” events, those observations that do not constitute a recognized event, can contribute meaningful information, for instance, answering the question of “how far?” is the completion of an event of interest.

Because of the high-level nature of this class of models, they often must be manually specified by a domain expert, i.e., learning model structure and/or parameters is generally infeasible/ill defined. Furthermore, the formalisms in this category of event models are largely deterministic, and the convenient mechanism of probabilistic reasoning to handle uncertainty (both in observation and interpretation) is generally unavailable.

The semantic event models are usually applied in event domains where the events of interest are relatively complex and a particular event has large variance in its appearance [20], [23], [24], [103].

A. Grammars

Language is a basic mechanism used by humans to define and describe video events. It is, therefore, intuitive that formal notions of language, as defined by grammar models, would be natural to model the inherently semantic properties of video events.

Grammar models [104] are well studied and have been used in several domains including speech recognition [105] and computer vision [106]. This formalism naturally captures sequence and hierarchical composition as well as long-term temporal dependencies.

Formally, a grammar model consists of three components: a set of terminals, a set of nonterminals, and a set of production rules.

In the domain of video event modeling, grammars are used as follows: terminals correspond to abstraction primitives. Similarly, nonterminals may correspond to semantic concepts (i.e., subevents). Production rules in an event model correspond to the semantic structure of the event. A semantic grammar event

model makes use of these components to represent a particular event domain.

The recognition of an event is reduced to determining whether a particular video sequence abstraction (sequence of terminals) constitutes an instance of an event. In formal grammar terminology, this process is called parsing. The particular set of production rules used in recognizing the event is called the parse.

For the classes of regular and context-free grammars [107], efficient polynomial-time algorithms exist for parsing [108].

Deterministic semantic grammar models have been used in several event domains including object manipulations [109] and two-person interactions [110].

A straightforward extension allows probabilities to be associated with each production rule. Grammar models utilizing this extension, called stochastic grammars (or sometimes probabilistic grammars), can give a probability score to a number of legal parses. This extension provides this formalism a mechanism to deal with the uncertainty inherent in video events. The parsing algorithm for deterministic grammars has been extended to work for stochastic grammars [111].

Stochastic grammars have been used in event domains such as parking lot surveillance [112], card game surveillance [113], complex task recognition (e.g., Japanese tea ceremonies) [114], [115], complex motions [116], [117], and human actions [118].

Attribute grammars, introduced by Knuth [119], formally associate conditions with each production rule. Each terminal has certain attributes associated with it, and the use of each production rule in a parse is conditioned upon these attributes. The conditions on each production rule introduce additional semantics into the event model and are specified using knowledge of the domain.

In video event understanding, attribute grammars have been used [120] to classify single-thread atomic object-based events in a parking lot surveillance event domain. See [1] for a more detailed example of an attribute grammar. If the production rule predicates are chosen to be probabilistic, attribute grammars can be considered a special case of stochastic grammars.

Due to the inherent nonsequential temporal relationships in many video events, particularly those defined using semantics, many works have attempted to introduce these relations into the grammar event models [110], [112], [113], [117], [121].

Learning of semantic event models including grammar models is a challenging problem. Although several works have explored the problem of automatically learning a grammar model for video event representation [122]–[125], the event description and recognition in semantic terms afforded by grammar approaches can, generally, only be achieved through manual specification of the model using expert domain knowledge.

B. Petri Nets

The nonsequential temporal relations that define many video events require a formalism that captures these relations naturally. Furthermore, as we have seen with BNs and HMMs, graphical formalisms allow a compact visualization of our event model.

The PN formalism allows such a graphical representation of the event model and can be used to naturally model semantic re-

lations that often occur in video events including nonsequential (and sequential) temporal relations, spatial and logical composition, hierarchy, concurrency, and partial ordering. PN event models are usually specified manually using knowledge of the domain.

More formally, PNs (introduced in [126]) are specified as a bipartite graph. Place nodes are represented as circles and transition nodes are represented as rectangles. Place nodes may hold tokens and transition nodes specify the movement of tokens between places when a state change occurs. A transition node is enabled if all input place nodes connected to it (those place nodes with directed arcs going to the transition node) have tokens. Enabled transition nodes may “fire” and affect a change in the distribution of tokens throughout the network. When an enabled transition node fires the tokens in the input place, nodes are deleted and new tokens are placed in each of the output place nodes (those place nodes with directed arcs coming from the transition). Transition nodes can have an enabling rule applied to them that may impose additional conditions on the enabling of the transition. A PN model marking is defined as the instantaneous configuration of tokens in various place nodes in the PN graph. For further details on the PN formalism, see [127]–[129].

In video event understanding, PN event model approaches can generally be categorized into two classes: object PNs and plan PNs [130]. We distinguish these approaches by the design choices made in constructing the event model.

Tokens in the object PN model correspond to video sequence objects and their properties. Place nodes in the object PN represent object states. Transition nodes represent either a change in an object state or the verification of a relation. The enabling rules of conditional transitions are conditioned only on the properties of the tokens (representing objects) in their immediate input place nodes. Particular transition nodes can represent events of interest. Multiple events of interest may be specified within the same object PN model.

The object PN model has been used in the event domains of traffic [22], [131] and people [24] surveillance.

Plan PNs are another approach to event modeling that represents each event as a “plan” of subevents. Each event is represented as a plan, which is a number of subevents connected in such a way as to enforce the temporal and logical relations between them. Each subevent is represented by a place node and can be considered to be occurring when a token is in this place node (these nodes only have a one-token capacity). Transitions between subevents are conditioned on general abstraction properties instead of on specific properties linked to input tokens (objects) as in the object PNs. An event is recognized when the “sink” transition of the plan fires. Unlike in object PNs, plan PNs require a separate model for each event.

Plan PN event models have been applied to several event domains including parking lot surveillance [132], people surveillance [133], and complex gesture recognition [134].

Some extensions to the PN formalism event model include timed transitions to allow representation of duration constraints [135], stochastic timed transitions for dealing with uncertainty within duration constraints [24], and associating probabilities with tokens to cope with uncertain observations [133].

In most known works employing PN models for the representation and recognition of video events, an object-based abstraction is used [24], [132], [133], [135].

An additional advantage of the PN event model is its ability to deal with “incomplete” events. Unlike other models, PNs are able to give a semantically meaningful snapshot of the video input at any time. This ability can be used to give a prediction on the next state or provide a likelihood of reaching a particular event of interest [24].

One drawback of the PN model is that the semantic nature makes learning these models from training data infeasible/ill defined. This raises concerns about the scalability of this approach to larger problems than those illustrated in the various works. Initial research has been done on translating standard knowledge specification formats for video events into PN event models [130].

Another disadvantage of PN event models is their deterministic nature. A recurring criticism of the PN formalism for video event understanding is their reliance on an error-free “perfect abstraction” in contrast to probabilistic formalisms (e.g., BN) that can use their probabilistic mechanism to correct for these errors. Some initial work into extending the PN formalism with such a probabilistic mechanism has been proposed in [133].

C. Constraint Satisfaction

Another approach to representation and recognition of a particular event domain in terms of semantic concepts and relations is to represent the event as a set of semantic constraints on the abstraction and to pose the problem of recognition as one of constraint satisfaction.

The advantage of this approach is that the constraints can be formulated as an ontology for a particular event domain and reused in different applications.

Early work in constraint recognition introduced the notion of chronicles, which are undirected constraint graphs describing the temporal constraints of atomic subevents [136], [137].

The event recognition task in these approaches is reduced to mapping the set of constraint to a temporal constraint network and determining whether the abstracted video sequence satisfies these constraints. While known algorithms exist to solve this problem, it is, in general, computationally intractable (NP-hard in the number of constraints). As a response to this, several event models have been proposed, which approximate the solution by such methods as reducing the domain of each node (representing a subevent) in the temporal constraint network [138] and eliminating arcs in the network with less relevance to the solution [103], [139].

Vu *et al.* [103], [139] achieve a speedup of the algorithm that allows it to be used in real-time surveillance applications. Their method, coupled with an object-based abstraction, has been evaluated extensively in several event domains including airport surveillance [140], home care applications [141], and others [142].

In addition to temporal constraints, more recent work incorporates semantic knowledge about temporal constraints pertaining to the properties of objects participating in the scene [103], [139], [143]–[145]. Description logics [146], [147] offer

a very rich framework for representing video events including compositional hierarchy specification as well as semantic relationships. Learning of these description logic models has also been explored [148].

An object-based abstraction is often coupled with the constraint satisfaction event models [103], [136], [139], [143]–[145]. Other works in constraint satisfaction event models assume a higher level abstraction where a video sequence is described in terms of atomic subevents [136], [138].

D. Logic Approaches

Early works in artificial intelligence (AI) regarding specification of semantic knowledge as a set of logic predicates discuss the specification of “event calculus” [149], [150].

Only recently, however, have logic-based event models been introduced for video event understanding. In this type of event model, knowledge about an event domain is specified as a set of logic predicates. A particular event is recognized using logical inference techniques such as resolution. These techniques are not tractable in general, but are useful as long as the number of predicates, inference rules, and groundings (usually corresponding to the number of objects in the video sequence) are kept low.

Initial work applies the first-order logic framework of Prolog to recognition in the event domain of parking lot surveillance [151].

To cope with the uncertainty inherent in video events, some extensions to logics approaches have been proposed including multivalued logics [152] and Markov logics [23].

It has not been studied how this class of event models will scale up to problems with many inference rules.

VIII. APPLICATIONS

In Section I, we suggested that the event understanding process can be roughly decomposed into two parts, which we have named abstraction and event modeling, respectively.

In the literature of event understanding, some works focus on the first part of the process (abstraction), others on the second part (event modeling), and still others focus on a particular pairing of an abstraction scheme with an event model, usually for application to a particular event domain (a “system”). Regardless of each paper’s emphasis, a choice is made for approaches for both abstraction and event modeling, i.e., a paper that focuses on a particular event modeling formalism must still select an abstraction scheme to illustrate the application of their model. For instance, a paper focused on event modeling may emphasize an HMM’s ability to recognize gestures over other types of models, but minimizes the discussion on why particular video sequence features were chosen as input to the HMM.

Usually, these choices are not made randomly, but are rather tuned to accentuate the strengths of the paper’s emphasis (the event model in the example). For this reason, the information on the grouping of abstraction schemes and event models and which event domains they have been applied to is interesting for future research. Those interested in applications of event understanding to a particular event domain can see what methods have been

TABLE II
REPRESENTATIVE SAMPLE OF THE WORK BEING DONE IN THE DOMAIN OF VIDEO EVENT UNDERSTANDING

Work	Abstraction Scheme	Event Model	Event Examples	Emphasis of paper
[21]	Gradient Histograms	Nearest Neighbor	"Walking", "Running", "Waving"	Abstraction
[153]	Pixel-Based Features	Naive Bayesian	"Active", "Inactive", "Walking", "Running", "Fighting"	Abstraction (Feature Selection)
[20]	Motion History Images	Nearest Neighbor	Aerobics Exercises	Abstraction
[30]	Pixel Energy History	Deviant Event Model	Events in an Office Setting	System
[156]	Pixel Change History	Gaussian Mixture Model	"Browsing", "Entering and Leaving" (Shopping Domain)	System Parameter Learning
[28]	Logical Abstraction	Rules Based in Force Dynamics	"Pick Up Object", "Put Down Object"	System
[157]	Principal components of Object Silhouettes	Nearest Neighbor	"Human Walking", "Dog Running"	Abstraction
[17]	Object-Based Abstraction	Bayesian Network, Finite State Machines	"Approach", "Blocking", "Stealing" (Surveillance Domain)	Abstraction, Hierarchical Combination of Event Models
[158]	Pixel-Based Abstraction	Nearest Neighbor	"Walk", "Run", "Skip", "Hop", "March"	Abstraction, Distance Measure Comparison
[26]	Space-Time Volume Features	Nearest neighbor	"Jumping-Jack", "Walking", "Running"	Abstraction
[32]	Transformation of Object Features using SVM	Support Vector Machine	"Somebody is Crossing the Corridor Going From Room A to B"	System
[34]	Pixel-Based Abstraction	Support Vector Machines + Voting	"Walk", "Run", "Skip"	System
[40]	Pixel-Based Abstraction Constructed Using Boosting	Boosted Discriminative Classifier	"Talking On Phone", "Yawning with Hand at Mouth", "Putting on Eyeglasses"	Boosting Methodology
[98]	Silhouette Features	Conditional Random Fields	"Walking", "Running", "Bending Down"	Event Model
[101]	Head Velocities	Conditional Random Fields	"Head Shakes", "Look Away" (Head Gestures)	Event Model
[47]	Object-Based Abstraction	Finite State Machine	"A Car Passing Through the Checkpoint", "A Car Avoiding the Checkpoint" (UAV Domain)	Event Model
[159]	Body Part Kinematics	Hierarchical Finite State Machine	"Hand-Shaking", "Kicking", "Pointing" (Two-Person Interactions)	System
[52]	Object-Based Abstraction	Hierarchical Finite State Machine	"Walking Past a Standing Car", "Opening the Door and Getting In", "Unusual Events"	Event Model, Learning Event Model
[50]	Face and Hand Locations	Finite State Machines	"Hand Wave", "Drawing a Circle", "Drawing a Figure Eight" (Gestures)	Automatic Model Learning
[56]	Object-Based Abstraction	Bayesian Network	"Overtaking", "Following" (UAV Video)	System
[59]	Object-Based Abstraction	Bayesian Network	"Vehicle Parked", "Pedestrian Passing By Vehicle"	Event Model
[60]	Object-Based Abstraction	Bayesian Network	American Football Plays	Event Model
[70]	Mesh Features	Hidden Markov Model	"Forehand Volley", "Backhand Stroke" (Tennis Strokes)	Event Model
[71]	(Hand) Object-Based Abstraction	Hidden Markov Model	American Sign Language Gestures	System
[74]	Pixel-based Abstraction	Entropic Hidden Markov Model	"Entering Room", "At Computer" (Office Domain) "North-South Traffic", "Pedestrians Stopping Traffic" (Traffic Domain)	Event Model
[75]	(Hand) Object-Based Abstraction	Parallel Hidden Markov Model	"Woman", "Try", "Teach" (Sign Language Gestures)	System
[78]	Object Trajectories	Coupled Hidden Markov Models	"Follow, Reach, and Walk Together" "Approach, Meet, and Go On Separately" (Person Interactions)	System
[79]	Pixel Change History	Gaussian Mixture Model (Sub-events) DML-HMM (Events)	Airport Cargo Loading/Unloading Events	Event Model
[80]	Silhouette/Motion Capture Features	Variable Length Markov Model	Exercise Domain	Event Model
[82]	Object-Based Abstraction	Bayesian Networ, Hidden Semi-Markov Model	"A Car Passing Through the Checkpoint", A Car Avoiding the Checkpoint (UAV Domain)	Event Model
[84]	Landmarks	Hierarchical HMM	"Short Meal", "Have Snack" (Kitchen Domain)	Event Model
[86]	Object Trajectories	Switching Hidden Semi-Markov Model	"Eating Breakfast", "Washing Dishes" (Kitchen Domain)	Event Model
[88]	Hand Locations	Coupled Hidden Semi-Markov Models	Sign Language Gesture	Event Model
[89]	Object-Based Abstraction	Recurrent Bayesian Network (DBN)	"Violent Behavior" (Metro Station Domain)	Event Model
[91]	Object-Based Abstraction	Propagation Networks (DBN)	"Glucose Monitor Calibration" (Assistive Technology)	Event Model
[22]	Object-Based Abstraction	Petri Net	"Car Exchange"	Event Model
[24]	Object-Based Abstraction	Petri Net	"Visitor Entered the Hall", "Security Check Is Too Long" (Surveillance)	Event Model
[132]	Object-Based Abstraction	Petri Net	"Vehicle Departure", "Arsonist Action" (Parking Lot)	Event Model
[112]	Object Trajectories	HMM (Sub-Events), Stochastic Grammar (Events)	(Hand Gestures), (Musical Conducting), (Parking Lot)	Event Model
[113]	Object-Based Abstraction	Stochastic Grammar	"Player Removed House Card", "Dealer Dealt Card to Player" (Card Game)	Event Model
[120]	Object-Based Abstraction	Attribute Grammar	"PARKING", "Dropoff" (Parking Lot)	Event Model
[125]	Object-Based Abstraction	Stochastic Grammar	(Convenience Store)	Learning Typical Events from Data

Generally, each work employs an abstraction scheme as well as an event model; however, only one of these is the main emphasis of the paper. The event domain is usually chosen to illustrate the usefulness of the specific abstraction scheme or event model emphasized in the paper. An exception to this are those papers that present a "system" of an abstraction scheme and event model targeted especially at a specific event domain. Recurrent event domains in the literature correspond to useful applications of eventual event understanding systems. These include unmanned aerial video, sign language recognition, and surveillance of people and cars.

applied previously in this domain, as well as what methods used in other domains may be applied.

To this end, Table II organizes many of the major works in the field of video event understanding surveyed in this paper, and gives a rough idea of the chosen abstraction scheme, event model, and event domain. The emphasis of the paper is in terms of the subdiscipline of event understanding (i.e., abstraction or event modeling). Papers that emphasize a coupling of approaches for a particular domain are listed as “system” in this column. Other emphases such as boosting and learning are explicitly stated.

By observing the table, we can conclude that, in general, a balance exists between the complexity of the abstraction scheme and that of the event model. Correspondingly, there is a group of works that emphasize abstraction utilizing a simple well-understood event modeling formalism [20], [21], [30], [153]. Similarly, other works, emphasizing event modeling, choose straightforward abstraction schemes [17], [24], [52], [86]. An object-based abstraction is observed to be popular among those papers that emphasize the event model.

Most recent works in event modeling are embracing formalisms with explicit representation of time (FSM, HMM, DBN). This aspect of the model is perhaps the most crucial for modeling video events.

The table also reveals the popular event domains being investigated within the literature. Unmanned aerial vehicle (UAV) surveillance, parking lot surveillance, two-person interaction, and sign language gestures are among the event domains that appear numerous times in the literature. Not surprisingly, these correspond to promising application of video event understanding.

IX. CONCLUSION AND FUTURE WORK

In this paper, we have presented our view of the domain of video event understanding. In Section II, we presented a terminology whose purpose is to resolve ambiguity within the community. The remainder of this paper has focused on grouping a number of problems, approaches to their solutions, and components of these approaches in a meaningful way.

While we believe this grouping is appropriate, it must be conceded that there is some variance within this class of problems. As such, it is unlikely that a general solution (i.e., some combination of methods) exists that will provide the best results for all event understanding problems. Rather, we can think of the many methods for abstraction and event modeling as a toolbox with each tool being called upon to address a specific type of problem. For this analogy to be apt, we must have a good understanding of both our tools (i.e., methods for abstraction and event modeling) and our problems (i.e., various event domains).

This is achieved by understanding the discriminating aspects of the event domain and applying those into the choice of abstraction. Furthermore, the structure of the event domain must be understood and used to select the event model.

The categorization into abstraction/event model subprocesses is introduced in this paper and is not prevalent in the community. It may be for this reason that we have seen approaches to

event understanding that mostly emphasize one or the other of these aspects. Future work, which takes into account this categorization, may provide insight on which abstraction scheme/event model pairings are the most useful for a particular event domain. Additionally, it would be informative to study how sensitive the recognition rates of a particular event model are to the chosen abstraction scheme.

From the body of work examined in this paper, it is also apparent that the popularity of probabilistic models is increasing. These models grow more complex as they attempt to better capture the structure of the events being modeled. This increase in model complexity necessitates more parameters to be estimated and more assumptions to be made. Other work has introduced semantic event models that do well to capture the structure of the event (they are built by a knowledgeable human in the event domain); however, they are unable to intrinsically capture uncertainty and often are less efficient in the event recognition phase.

The ideal event model would combine the advantages of these approaches: robust representational capability including semantic relations, dealing with uncertainty, and efficient recognition algorithms. Such event models as Markov logics [23] and probabilistic PNs [133] are a step in this direction.

Finally, it is apparent that automatic inference of event models from data is essential for adaptability and scalability of real event understanding systems. However, save for a few works [95], [125], [148], [154], [155], this aspect of event understanding has not been explored. This is in large part due to the fact that many of the currently used formalisms do not easily lend themselves to tractable approaches to model learning.

REFERENCES

- [1] G. Lavee, M. Rudzsky, E. Rivlin, and A. Borzin, “Understanding video events: A survey of methods for automatic interpretation of semantic occurrences in video,” Technion—Israel Inst. Technol., Haifa, Israel, Tech. Rep. CIS-2009-06, 2009.
- [2] Caretaker Project. (2006). [Online]. Available: <http://www.ist-caretaker.org/>
- [3] Etiseo Project. (2004). [Online]. Available: <http://www-sop.inria.fr/orion/etiseo/>
- [4] Avitrack Project [Online]. Available: <http://www.avitrack.net/>
- [5] Advisor Project. (2000). [Online]. Available: <http://www-sop.inria.fr/orion/advisor/>
- [6] Beware Project. (2007). [Online]. Available: <http://www.dcs.qmul.ac.uk/ssg/beware/>
- [7] Icons Project. (2000). [Online]. Available: <http://www.dcs.qmul.ac.uk/research/vision/projects/icons/>
- [8] Vsam Project. (1997). [Online]. Available: <http://www.cs.cmu.edu/vsam/>
- [9] J. K. Aggarwal and Q. Cai, “Human motion analysis: A review,” *Comput. Vis. Image Understanding*, vol. 73, no. 3, pp. 428–440, 1999.
- [10] H. Buxton, “Generative models for learning and understanding dynamic scene activity,” presented at the ECCV Workshop Generative Model Based Vis., Copenhagen, Denmark, Jun. 2002.
- [11] P. Turaga, R. Chellappa, V. S. Subrahmanian, and O. Udrea, “Machine recognition of human activities: A survey,” *IEEE Trans. Circuits, Syst. Video Technol.*, vol. 18, no. 11, pp. 1473–1488, 2008.
- [12] W. Hu, T. Tan, L. Wang, and S. Maybank, “A survey on visual surveillance of object motion and behaviors,” *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 34, no. 3, pp. 334–352, Aug. 2004.
- [13] H.-H. Nagel, “From image sequences towards conceptual descriptions,” *Image Vis. Comput.*, vol. 6, no. 2, pp. 59–74, 1988.
- [14] A. Bobick, “Movement, activity, and action: The role of knowledge in the perception of motion,” *Roy. Soc. Workshop Knowl.-Based Vis. Man Mach.*, vol. B-352, pp. 1257–1265, 1997.

- [15] F. Bremond, "Scene understanding: Perception, multi-sensor fusion, spatio-temporal reasoning and activity recognition," Ph.D. dissertation, HDR Université de Nice-Sophia Antipolis, Nice Cedex, France, Jul. 2007.
- [16] A. G. Cohn, D. R. Magee, A. Galata, D. Hogg, and S. M. Hazarika, "Towards an architecture for cognitive vision using qualitative spatio-temporal representations and abduction," in *Proc. Spatial Cognit.*, 2003, pp. 232–248.
- [17] S. Hongeng and R. Nevatia, "Multi-agent event recognition," in *Proc. Int. Conf. Comput. Vis.*, 2001, pp. 84–93.
- [18] R. Howarth and H. Buxton, "Conceptual descriptions from monitoring and watching image sequences," *Image Vis. Comput.*, vol. 18, no. 2, pp. 105–135, Jan. 2000.
- [19] T. Xiang and S. Gong, "Beyond tracking: Modelling activity and understanding behaviour," *Int. J. Comput. Vis.*, vol. 67, no. 1, pp. 21–51, 2006.
- [20] A. Bobick and J. Davis, "The recognition of human movement using temporal templates," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 3, pp. 257–267, Mar. 2001.
- [21] L. Zelnik-Manor and M. Irani, "Statistical analysis of dynamic actions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 9, pp. 1530–1535, Sep. 2006.
- [22] N. Ghanem, D. DeMenthon, D. Doermann, and L. Davis, "Representation and recognition of events in surveillance video using Petri nets," in *Proc. Conf. Comput. Vis. Pattern Recognit. Workshop (CVPRW 2004)*, vol. 7, pp. 112–121.
- [23] S. D. Tran and L. S. Davis, "Event modeling and recognition using Markov logic networks," in *Proc. Eur. Conf. Comput. Vis.*, 2008, pp. 610–623.
- [24] A. Borzin, E. Rivlin, and M. Rudzsky, "Surveillance interpretation using generalized stochastic Petri nets," in *Proc. Int. Workshop Image Anal. Multimedia Interactive Serv. (WIAMIS)*, 2007, 4 pp.
- [25] N. Oliver, B. Rosario, and A. Pentland, "A Bayesian computer vision system for modeling human interactions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 831–843, Aug. 2000.
- [26] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri, "Actions as space-time shapes," in *Proc. 10th IEEE Int. Conf. Comput. Vision (ICCV 2005)*, Washington, DC: IEEE Comput. Soc., 2005, pp. 1395–1402.
- [27] C. Schuldt, I. Laptev, and B. Caputo, "Recognizing human actions: A local SVM approach," in *Proc. 17th Int. Conf. Pattern Recognit. (ICPR 2004)*, vol. 3, pp. 32–36.
- [28] J. M. Siskind, "Visual event classification via force dynamics," in *Proc. 17th Nat. Conf. Artif. Intell. 12th Conf. Innovative Appl. Artif. Intell.*, Menlo Park, CA/Cambridge, MA: AAAI Press/MIT Press, 2000, pp. 149–155.
- [29] A. G. Cohn, D. Magee, A. Galata, D. Hogg, and S. Hazarika, "Towards an architecture for cognitive vision using qualitative spatio-temporal representations and abduction," in *Spatial Cognition III* (Lecture Notes in Computer Science), C. Freksa, C. Habel, and K. Wender, Eds. New York: Springer-Verlag, 2003, pp. 232–248.
- [30] S. Gong and J. Ng, "Learning pixel-wise signal energy for understanding semantics," in *Proc. Br. Mach. Vis. Conf.*, 2001, pp. 1183–1189.
- [31] E. Shechtman and M. Irani, "Space-time behavior based correlation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2005, vol. 1, pp. 405–412.
- [32] M. Pittore, C. Basso, and A. Verri, "Representing and recognizing visual dynamic events with support vector machines," in *Proc. Int. Conf. Image Anal. Process.*, 1999, pp. 18–23.
- [33] M. Fleischman, P. Decamp, and D. Roy, "Mining temporal patterns of movement for video content classification," in *Proc. 8th ACM Int. Workshop Multimedia Inf. Retrieval (MIR)*, New York: ACM Press, 2006, pp. 183–192.
- [34] D. B. D. Cao, O. Masoud, and N. Papanikolopoulos, "Online motion classification using support vector machines," in *Proc. IEEE Int. Conf. Robot. Autom.*, New Orleans, LA, 2004, pp. 2291–2296.
- [35] D. Xu and S. F. Chang, "Visual event recognition in news video using kernel methods with multi-level temporal alignment," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Minneapolis, MN, Jun. 2007, pp. 1–8.
- [36] C. Piciarelli, G. Foresti, and L. Snidaro, "Trajectory clustering and its applications for video surveillance," in *Proc. IEEE Int. Conf. Adv. Video Signal Based Surveillance*, 2005, pp. 40–45.
- [37] I. Laptev and P. Pérez, "Retrieving actions in movies," presented at the Int. Conf. Comput. Vis., Rio de Janeiro, Brazil, Oct. 2007.
- [38] D. Minnen, T. Westeyn, and T. Starner, "Recognizing soldier activities in the field," presented at the 4th Int. Workshop Wearable Implantable Body Sens. Netw., Aachen, Germany, 2007.
- [39] S. Nowozin, G. Bakir, and K. Tsuda, "Discriminative subsequence mining for action classification," in *Proc. 11th IEEE Int. Conf. Comput. Vis.*, 2007, pp. 1919–1923.
- [40] P. Smith, N. da Vitoria Lobo, and M. Shah, "Temporal boost for event recognition," in *Proc. 10th IEEE Int. Conf. Comput. Vision (ICCV)*, vol. 1, Washington, DC: IEEE Comput. Soc., 2005, pp. 733–740.
- [41] P. Ribeiro, P. Moreno, and J. S. Victor, "Boosting with temporal consistent learners: An application to human activity recognition," in *Proc. Int. Symp. Vis. Comput.*, 2007, pp. 464–475.
- [42] P. Canotilho and R. P. Moreno, "Detecting luggage related behaviors using a new temporal boost algorithm," in *Proc. IEEE Int. Workshop Perform. Eval. Tracking Surveillance*, 2007, pp. 1–6.
- [43] H. Vassilakis, A. J. Howell, and H. Buxton, "Comparison of feedforward (TDRBF) and generative (TDRGBN) network for gesture based control," in *Revised Papers From the International Gesture Workshop on Gesture and Sign Languages in Human-Computer Interaction (GW)*. London, U.K.: Springer-Verlag, 2002, pp. 317–321.
- [44] A. Gill, *Introduction to the Theory of Finite State Machines*. New York: McGraw-Hill, 1962.
- [45] K. H. Jo, Y. Kuno, and Y. Shirai, "Manipulative hand gesture recognition using task knowledge for human computer interaction," in *Proc. 3rd Int. Conf. Face Gesture Recognit. (FG)*. Washington, DC: IEEE Comput. Soc., 1998, pp. 468–473.
- [46] F. Lv and R. Nevatia, "Single view human action recognition using key pose matching and Viterbi path searching," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2007, pp. 1–8.
- [47] G. G. Medioni, I. Cohen, F. Bremond, S. Hongeng, and R. Nevatia, "Event detection and analysis from video streams," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 8, pp. 873–889, Aug. 2001.
- [48] F. Bremond and M. Thonnat, "Analysis of human activities described by image sequences," in *Proc. Int. Florida Artif. Intell. Res. Symp. (FLAIRS 1997)*, Daytona Beach, FL, May.
- [49] F. Bremond, M. Thonnat, and M. Zuniga, "Video understanding framework for automatic behavior recognition," *Behav. Res. Methods*, vol. 3, no. 38, pp. 416–426, 2006.
- [50] P. Hong, T. S. Huang, and M. Turk, "Gesture modeling and recognition using finite state machines," in *Proc. 4th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG)*. Washington, DC: IEEE Comput. Soc., 2000, pp. 410–415.
- [51] S. Park, J. Park, and J. K. Aggarwal, "Video retrieval of human interactions using model-based motion tracking and multi-layer finite state automata," in *Proc. ACM Int. Conf. Image Video Retrieval*, 2003, pp. 394–403.
- [52] D. Mahajan, N. Kwatra, S. Jain, P. Kalra, and S. Banerjee, "A framework for activity recognition and detection of unusual activities," in *Proc. Indian Conf. Comput. Vis., Graph. Image Process.*, 2004, pp. 1–7.
- [53] F. V. Jensen, *Bayesian Networks and Decision Graphs* (Information Science and Statistics). New York: Springer-Verlag, Jul. 2007.
- [54] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Francisco, CA: Morgan Kaufmann, 1988.
- [55] Jordan, *Learning in Graphical Models (Adaptive Computation and Machine Learning)*. Cambridge, MA: MIT Press, Nov. 1998.
- [56] H. Buxton and S. Gong, "Visual surveillance in a dynamic and uncertain world," *Artif. Intell.*, vol. 78, no. 1/2, pp. 431–459, 1995.
- [57] R. P. Higgins, "Automatic event recognition for enhanced situational awareness in UAV video," in *Proc. Military Commun. Conf.*, 2005, pp. 1706–1711.
- [58] F. Lv, X. Song, B. Wu, V. K. Singh, and R. Nevatia, "Left-luggage detection using Bayesian inference," in *Proc. 9th IEEE Int. Workshop Perform. Eval. Tracking Surveillance*, 2006, pp. 83–90.
- [59] P. Remagnino, T. Tan, and K. Baker, "Multiagent visual surveillance of dynamic scenes," *Image Vis. Comput.*, vol. 16, no. 8, pp. 529–532, Jun. 1998.
- [60] S. S. Intille and A. F. Bobick, "A framework for recognizing multi-agent action from visual evidence," in *Proc. 16th Nat. Conf. Artif. Intell. 11th Innovative Appl. Artif. Intell. Conf. Innovative Appl. Artif. Intell.*, 1999, pp. 518–525.
- [61] S. Hongeng, R. Nevatia, and F. Bremond, "Video-based event recognition: Activity representation and probabilistic recognition methods," *Comput. Vis. Image Understanding*, vol. 96, no. 2, pp. 129–162, 2004.

- [62] J. Niebles, H. Wang, and L. Fei-Fei, "Unsupervised learning of human action categories using spatial-temporal words," in *Proc. Br. Mach. Vis. Conf.*, 2006, vol. 3, pp. 1249–1258.
- [63] S. Wong, T. Kim, and R. Cipolla, "Learning motion categories using both semantic and structural information," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2007, pp. 1–6.
- [64] X. Wang, X. Ma, and W. Grimson, "Unsupervised activity perception by hierarchical Bayesian models," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2007, pp. 1–8.
- [65] T. Hofmann, "Probabilistic latent semantic indexing," in *Proc. 22nd Annu. Int. ACM SIGIR Conf. Res. Dev. Inf. Retrieval*, 1999, pp. 50–57.
- [66] P. Dollár, V. Rabaud, G. Cottrell, and S. Belongie, "Behavior recognition via sparse spatio-temporal features," in *Proc. IEEE Int. Workshop Perform. Eval. Tracking Surveillance*, Oct. 2005, pp. 65–72.
- [67] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Readings Speech Recognit.*, vol. 77, no. 2, pp. 267–296, 1990.
- [68] Z. Ghahramani and M. I. Jordan, "Factorial hidden Markov models," in *Proc. Conf. Advances in Neural Information Processing Systems, NIPS*, vol. 8, D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo, Eds. Cambridge, MA: MIT Press, 1995, pp. 472–478.
- [69] S. Gong and H. Buxton, "On the visual expectations of moving objects," in *Proc. 10th Eur. Conf. Artif. Intell. (ECAI)*. New York: Wiley, 1992, pp. 781–784.
- [70] J. Yamato, J. Ohya, and K. Ishii, "Recognizing human action in time-sequential images using hidden Markov models," in *Proc. Int. Conf. Comput. Vis.*, 1992, pp. 379–385.
- [71] T. Starner and A. Pentland, "Real-time American sign language recognition from video using hidden Markov models," in *Proc. IEEE Symp. Comput. Vis.*, 1995, pp. 265–270.
- [72] J. Schlenzig, E. Hunter, and R. Jain, "Recursive identification of gesture inputs using hidden Markov models," in *Proc. 2nd IEEE Workshop Appl. Comput. Vis.*, 1994, pp. 187–194.
- [73] A. Ogale, A. Karapurkar, G. Guerra Filho, and Y. Aloimonos, "View-invariant identification of pose sequences for action recognition," presented at the Video Anal. Content Extraction Workshop (VACE), Tampa, FL, 2004.
- [74] M. Brand and V. Kettner, "Discovery and segmentation of activities in video," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 844–851, Aug. 2000.
- [75] C. Vogler and D. Metaxas, "A framework for recognizing the simultaneous aspects of American sign language," *Comput. Vis. Image Understanding*, vol. 81, no. 3, pp. 358–384, 2001.
- [76] M. Brand, N. Oliver, and A. Pentland, "Coupled hidden Markov models for complex action recognition," in *IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.* Los Alamitos, CA: IEEE Comput. Soc., 1997, pp. 994–999.
- [77] M. Brand, "The inverse hollywood problem: From video to scripts and storyboards via causal analysis," in *Proc. 14th Nat. Conf. Artif. Intell. 9th Conf. Innovative Appl. Artif. Intell.*, 1997, pp. 132–137.
- [78] N. M. Oliver, B. Rosario, and A. Pentland, "A Bayesian computer vision system for modeling human interactions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 831–843, Aug. 2000.
- [79] S. Gong and T. Xiang, "Recognition of group activities using dynamic probabilistic networks," in *Proc. 9th IEEE Int. Conf. Comput. Vis. (ICCV)*. Washington, DC: IEEE Comput. Soc., 2003, pp. 742–749.
- [80] A. Galata, N. Johnson, and D. Hogg, "Learning variable-length Markov models of behavior," *Comput. Vis. Image Understanding*, vol. 81, no. 3, pp. 398–413, 2001.
- [81] A. Galata, A. Cohn, D. Magee, and D. Hogg, "Modeling interaction using learnt qualitative spatio-temporal relations and variable length Markov models," presented at the Conf. Artif. Intell., Lyon, France, 2002.
- [82] S. Hongeng and R. Nevatia, "Large-scale event detection using semi-hidden Markov models," in *Proc. 9th IEEE Int. Conf. Comput. Vis. (ICCV)*. Washington, DC: IEEE Comput. Soc., 2003, pp. 1455–1462.
- [83] S. Fine, Y. Singer, and N. Tishby, "The hierarchical hidden Markov model: Analysis and applications," *Mach. Learn.*, vol. 32, no. 1, pp. 41–62, 1998.
- [84] N. T. Nguyen, D. Q. Phung, S. Venkatesh, and H. Bui, "Learning and detecting activities from movement trajectories using the hierarchical hidden Markov models," in *IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.* Washington, DC: IEEE Comput. Soc., 2005, pp. 955–960.
- [85] N. Oliver, E. Horvitz, and A. Garg, "Layered representations for human activity recognition," in *Proc. 4th IEEE Int. Conf. Multimodal Interfaces*, 2002, pp. 3–8.
- [86] T. V. Duong, H. H. Bui, D. Q. Phung, and S. Venkatesh, "Activity recognition and abnormality detection with the switching hidden semi-Markov model," in *Proc. 2005 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1. Washington, DC: IEEE Comput. Soc., 2005, pp. 838–845.
- [87] P. Natarajan and R. Nevatia, "Hierarchical multi-channel hidden semi Markov models," in *Proc. Int. Joint Conf. Artif. Intell.*, M. M. Veloso, Ed., 2007, pp. 2562–2567.
- [88] P. Natarajan and R. Nevatia, "Coupled hidden semi Markov models for activity recognition," in *Proc. IEEE Workshop Motion Video Comput.*, Feb. 2007, 10 pp.
- [89] N. Moenne-Loccoz, F. Bremond, and M. Thonnat, "Recurrent Bayesian network for the recognition of human behaviors from video," in *Proc. Int. Conf. Comput. Vis. Syst.*, 2003, pp. 68–77.
- [90] B. Laxton, J. Lim, and D. Kriegman, "Leveraging temporal, contextual and ordering constraints for recognizing complex activities in video," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2007, pp. 1–8.
- [91] Y. Shi, Y. Huang, D. Minnen, A. Bobick, and I. Essa, "Propagation networks for recognition of partially ordered sequential action," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2004, vol. 2, pp. 862–869.
- [92] Y. Shi and A. F. Bobick, "P-net: A representation for partially-sequenced, multi-stream activity," in *Proc. 2003 Conf. Comput. Vis. Pattern Recognit. Workshop*, vol. 4, pp. 40–46.
- [93] N. Oliver and E. Horvitz, "A comparison of HMMs and dynamic Bayesian networks for recognizing office activities," in *Proc. User Model.*, 2005, pp. 199–209.
- [94] J. Muncaster and Y. Ma, "Activity recognition using dynamic Bayesian networks with automatic state selection," in *Proc. IEEE Workshop Motion Video Comput.*, 2007, pp. 30–30.
- [95] Y. Shi, A. Bobick, and I. Essa, "Learning temporal sequence model from partially labeled data," in *Proc. 2006 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*. Washington, DC: IEEE Comput. Soc., 2006, pp. 1631–1638.
- [96] J. Lafferty, A. McCallum, and F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proc. 18th Int. Conf. Mach. Learn.* San Francisco, CA: Morgan Kaufmann, 2001, pp. 282–289.
- [97] C. Sutton and A. McCallum, "An introduction to conditional random fields for relational learning," in *Introduction to Statistical Relational Learning*, L. Getoor and B. Taskar, Eds. Cambridge, MA: MIT Press, 2006.
- [98] C. Sminchisescu, A. Kanaujia, Z. Li, and D. Metaxas, "Conditional models for contextual human motion recognition," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2005, pp. 1808–1815.
- [99] T. Wang, J. Li, Q. Diao, W. Hu, Y. Zhang, and C. Dulong, "Semantic event detection using conditional random fields," in *Proc. 2006 Conf. Comput. Vis. Pattern Recognit. Workshop (CVPRW)*. Washington, DC: IEEE Comput. Soc., 2006, pp. 109–109.
- [100] L. Wang and D. Suter, "Recognizing human activities from silhouettes: Motion subspace and factorial discriminative graphical model," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2007, pp. 1–8.
- [101] L. Morency, A. Quattoni, and T. Darrell, "Latent-dynamic discriminative models for continuous gesture recognition," Massachusetts Inst. Technol. (MIT), Cambridge, Tech. Rep. MIT-CSAIL-TR-2007-002, 2007.
- [102] H. Ning, W. Xu, Y. Gong, and T. S. Huang, "Latent pose estimator for continuous action recognition," in *Proc. Eur. Conf. Comput. Vis.*, 2008, pp. 419–433.
- [103] V. T. Vu, F. Brémond, and M. Thonnat, "Automatic video interpretation: A recognition algorithm for temporal scenarios based on pre-compiled scenario models," in *Proc. Int. Conf. Comput. Vis. Syst.*, 2003, pp. 523–533.
- [104] A. V. Aho and J. D. Ullman, *The Theory of Parsing, Translation, and Compiling*. Upper Saddle River, NJ: Prentice-Hall, 1972.
- [105] F. Jelinek, *Statistical Methods for Speech Recognition*. Cambridge, MA: MIT Press, Jan. 1998.
- [106] G. Chanda and F. Dellaert, "Grammatical methods in computer vision: An overview," Georgia Inst. Technology, Atlanta, Tech. Rep. GIT-GVU-04-29, 2004.

- [107] N. Chomsky, *Syntactic Structures*. Hague, The Netherlands: Mouton, 1957.
- [108] J. Earley, "An efficient context-free parsing algorithm," *Commun. ACM*, vol. 13, no. 2, pp. 94–102, 1970.
- [109] M. Brand, "Understanding manipulation in video," in *Proc. 2nd Int. Conf. Autom. Face Gesture Recognit. (FG)*. Washington, DC: IEEE Comput. Soc., 1996, pp. 94–99.
- [110] M. S. Ryoo and J. K. Aggarwal, "Recognition of composite human activities through context-free grammar based representation," in *IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.* Washington, DC: IEEE Comput. Soc., 2006, pp. 1709–1718.
- [111] A. Stolcke, "An efficient probabilistic context-free parsing algorithm that computes prefix probabilities," *Comput. Linguistics*, vol. 21, pp. 165–201, 1995.
- [112] Y. A. Ivanov and A. F. Bobick, "Recognition of visual activities and interactions by stochastic parsing," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 852–872, Aug. 2000.
- [113] D. Moore and I. Essa, "Recognizing multitasked activities using stochastic context-free grammar," in *CVPR Workshop Models vs Exemplars Comput. Vis.*, 2001, pp. 770–776.
- [114] M. Yamamoto, H. Mitomi, F. Fujiwara, and T. Sato, "Bayesian classification of task-oriented actions based on stochastic context-free grammar," in *Proc. 7th Int. Conf. Autom. Face Gesture Recognit. (FGR)*. Washington, DC: IEEE Comput. Soc., 2006, pp. 317–323.
- [115] D. Minnen, I. Essa, and T. Starner, "Expectation grammars: Leveraging high-level expectations for activity recognition," in *Proc. Comput. Vis. Pattern Recognit.*, Jun. 2003, vol. 2, pp. 626–632.
- [116] D. Lymberopoulos, A. S. Ogale, A. Savvides, and Y. Aloimonos, "A sensory grammar for inferring behaviors in sensor networks," in *Proc. 5th Int. Conf. Inf. Process. Sensor Netw. (IPSN)*. New York: ACM Press, 2006, pp. 251–259.
- [117] Z. Zhang, K. Huang, and T. Tan, "Multi-thread parsing for recognizing complex events in videos," in *Proc. Eur. Conf. Comput. Vis.*, 2008, pp. 738–751.
- [118] A. Ogale, A. Karapurkar, and Y. Aloimonos, "View-invariant modeling and recognition of human actions using grammars," in *Proc. IEEE Workshop Dynam. Vis.*, 2005, pp. 115–126.
- [119] D. E. Knuth, "Semantics of context-free languages," *Math. Syst. Theory*, vol. 2, no. 2, pp. 127–145, 1968.
- [120] S. W. Joo and R. Chellappa, "Attribute grammar-based event recognition and anomaly detection," in *Proc. 2006 Conf. Comput. Vis. Pattern Recognit. Workshop (CVPRW)*. Washington, DC: IEEE Comput. Soc., 2006, pp. 107–107.
- [121] Z. Zhang, K. Huang, and T. Tan, "Complex activity representation and recognition by extended stochastic grammar," in *Proc. Asian Conf. Comput. Vis.*, 2006, pp. 150–159.
- [122] K. Cho, H. Cho, and K. Um, "Human action recognition by inference of stochastic regular grammars," in *Proc. Struct., Syntactic, Stat. Pattern Recognit.*, 2004, pp. 388–396.
- [123] K. Cho, H. Cho, and K. Um, "Inferring stochastic regular grammar with nearness information for human action recognition," in *Proc. Int. Conf. Image Anal. Recognit.*, 2006, pp. 193–204.
- [124] G. Guerra-Filho and Y. Aloimonos, "Learning parallel grammar systems for a human activity language," *Comput. Sci. Dept., Univ. Maryland, College Park, Tech. Rep. CS-TR-4837*, 2006.
- [125] K. Kitani, Y. Sato, and A. Sugimoto, "Recovering the basic structure of human activities from a video-based symbol string," in *Proc. IEEE Workshop Motion Video Comput.*, 2007, pp. 9–17.
- [126] C. A. Petri, "Kommunikation mut automaten," Ph.D. dissertation, Schriften des IIM Nr. 2, Bonn, Germany, 1962.
- [127] D. Kartson, G. Balbo, S. Donatelli, G. Franceschinis, and G. Conte, *Modelling With Generalized Stochastic Petri Nets*. New York: Wiley, 1994.
- [128] T. Murata, "Petri nets: Properties, analysis and applications," *Proc. IEEE*, vol. 77, no. 4, pp. 541–580, Apr. 1989.
- [129] P. J. Haas, *Stochastic Petri Nets: Modelling, Stability, Simulation*. New York: Springer-Verlag, 2002.
- [130] G. Lavee, A. Borzin, E. Rivlin, and M. Rudzsky, "Building Petri nets from video event ontologies," in *Proc. Int. Symp. Vis. Comput.*, 2007, pp. 442–451.
- [131] N. M. Ghanem, "Petri net models for event recognition in surveillance videos," Ph.D. dissertation, Univ. Maryland, College Park, 2007.
- [132] C. Castel, L. Chaudron, and C. Tessier, "What is going on? A high level interpretation of sequences of images," presented at the Eur. Conf. Comput. Vis., Cambridge, U.K., 1996.
- [133] M. Albanese, V. Moscato, R. Chellappa, A. Picariello, P. T. V. S. Subrahmanian, and O. Udrea, "A constrained probabilistic Petri-net framework for human activity detection in video," *IEEE Trans. Multimedia*, no. 6, pp. 982–996, Nov. 2008.
- [134] Y. Nam, N. Wahn, and H. Lee-Kwang, "Modeling and recognition of hand gesture using colored Petri nets," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 29, no. 5, pp. 514–521, Sep. 1999.
- [135] N. Ghanem, D. Doermann, L. Davis, and D. DeMenthon, "Mining tools for surveillance video," in *Proc. SPIE 16th Int. Symp. Electron. Imag.*, Jan. 2004, vol. 5307, pp. 259–270.
- [136] C. Dousson, P. Gaborit, and M. Ghallab, "Situation recognition: Representation and algorithms," in *Proc. Int. Joint Conf. Artif. Intell.*, 1993, pp. 166–172.
- [137] M. Ghallab, "On chronicles: Representation, on-line recognition and learning," in *Proc. 5th Int. Conf. Principles Knowl. Representation Reason.*, Nov. 1996, pp. 597–606.
- [138] C. S. Pinhanez and A. F. Bobick, "Human action detection using PNF propagation of temporal constraints," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*. Washington, DC: IEEE Comput. Soc., 1998, pp. 898–904.
- [139] V. Vu, F. Bremond, and M. Thonnat, "Temporal constraints for video interpretation," presented at the 15th Eur. Conf. Artif. Intell., Lyon, France, 2002.
- [140] F. Fugier, V. Valentin, F. Bremond, M. Thonnat, M. Borg, D. Thirde, and J. Ferryman, "Video understanding for complex activity recognition," *Mach. Vis. Appl.*, vol. 18, no. 3, pp. 167–188, 2007.
- [141] N. Zouba, F. Bremond, and M. Thonnat, "Monitoring activities of daily living (ADLs) of elderly based on 3d key human postures," presented at the 4th Int. Cognit. Vis. Workshop (ICVW 2008), Santorini, Greece, May.
- [142] V. T. Vu, "Temporal scenarios for automatic video interpretation," Ph.D. dissertation, Universite de Nice Sophia Antipolis, Nice, France, 2004.
- [143] N. Chelq and M. Thonnat, "Real-time image sequence interpretation for video-surveillance applications," presented at the Int. Conf. Image Process. (ICIP 1996), Lausanne, Switzerland.
- [144] N. Rota and M. Thonnat, "Activity recognition from video sequences using declarative models," presented at the 14th Eur. Conf. Artif. Intell. (ECAI 2000), Berlin, Germany.
- [145] N. Rota and M. Thonnat, "Video sequence interpretation for visual surveillance," in *Proc. 3rd IEEE Int. Workshop Vis. Surveillance (VS)*. Washington, DC: IEEE Comput. Soc., 2000, pp. 59–68.
- [146] K. Terzic, L. Hotz, and B. Neumann, "Division of work during behaviour recognition—The scenic approach," in *Proc. Workshop Behav. Monitoring Interpretation*, 2007, pp. 144–159.
- [147] B. Neumann and R. Möller, "On scene interpretation with description logics," *Image Vis. Comput.*, vol. 26, no. 1, pp. 82–101, 2008.
- [148] J. Hartz and B. Neumann, "Learning a knowledge base of ontological concepts for high-level scene interpretation," in *Proc. 6th Int. Conf. Mach. Learn. Appl. (ICMLA)*. Washington, DC: IEEE Comput. Soc., 2007, pp. 436–443.
- [149] M. Shanahan, "An abductive event calculus planner," *J. Log. Program.*, vol. 44, no. 1–3, pp. 207–240, 2000.
- [150] M. Shanahan, "Representing continuous change in the event calculus," in *Proc. Eur. Conf. Artif. Intell.*, 1990, pp. 598–603.
- [151] V. D. Shet, D. Harwood, and L. S. Davis, "Vidmap: Video monitoring of activity with prolog," in *Proc. IEEE Conf. Adv. Video Signal Based Surveillance*, 2005, pp. 224–229.
- [152] V. D. Shet, D. Harwood, and L. S. Davis, "Multivalued default logic for identity maintenance in visual surveillance," in *Proc. Eur. Conf. Comput. Vis.*, 2006, pp. 119–132.
- [153] J. S.-V. P. Ribeiro, "Human activities recognition from video: Modeling, feature selection and classification architecture," in *Proc. Workshop Human Activity Recognit. Model.*, Oxford, U.K., Sep. 2005, pp. 61–70.
- [154] C. Town, "Ontology-driven Bayesian networks for dynamic scene understanding," in *Proc. 2004 Conf. Comput. Vis. Pattern Recognit. Workshop (CVPRW)*, vol. 7. Washington, DC: IEEE Comput. Soc., 2004, pp. 116–116.
- [155] A. Toshev, F. Bremond, and M. Thonnat, "An apriori-based method for frequent composite event discovery in videos," in *Proc. 4th IEEE Int. Conf. Comput. Vis. Syst. (ICVS)*. Washington, DC: IEEE Comput. Soc., 2006, pp. 10–17.
- [156] S. Gong and T. Xiang, "Scene event recognition without tracking," *Acta Autom. Sinica*, vol. 29, no. 3, pp. 321–331, May 2003.
- [157] R. Goldenberg, R. Kimmel, E. Rivlin, and M. Rudzsky, "Behavior classification by eigendecomposition of periodic motions," *Pattern Recognit.*, vol. 38, no. 8, pp. 1033–1043, 2005.

- [158] O. Masoud and N. Papanikolopoulos, "A method for human action recognition," *Image Vis. Comput.*, vol. 21, no. 8, pp. 729–743, Aug. 2003.
- [159] S. Park and J. Aggarwal, "A hierarchical Bayesian network for event recognition of human actions and interactions," *Multimedia Syst.*, vol. 10, no. 2, pp. 164–179, Aug. 2004.



Michael Rudzsky received the Ph.D. degree in physics and mathematics from the Institute of Space Research, Moscow, Russia, in 1980.

Till 1990, he was with the Scientific and Industrial Association for Space Research, Baku, Azerbaijan. He is currently a Research Fellow with the Department of Computer Science, Technion—Israel Institute of Technology, Haifa, Israel. His current research interests include computer vision, pattern recognition, and compression of images.



Gal Lavee received the B.Sc. degree in computer science from the University of North Texas, Denton, and the M.Sc. degree in computer science from the University of Texas at Dallas, Richardson. He is currently working toward the Ph.D. degree at the Technion—Israel Institute of Technology, Haifa, Israel.

His current research interests include video event understanding, knowledge representation, and learning of semantic knowledge.



Ehud Rivlin received the B.Sc. and M.Sc. degrees in computer science and the M.B.A. degree from Hebrew University, Jerusalem, Israel, and the Ph.D. degree from the University of Maryland, College Park.

He is currently an Associate Professor with the Department of Computer Science, Technion—Israel Institute of Technology, Haifa, Israel. His current research interests include machine vision and robot navigation.