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

Gal Lavee , Ehud Rivlin, and Michael Rudzsky

April 2, 2009

Abstract: Understanding Video Events, 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 work 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 work we survey 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. 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 abstraction schemes (e.g. Motion History Images) and 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.

1 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 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: CARETAKER [1], ETISEO [2], AVITRACK [3], ADVISOR [4], BEWARE [5], ICONS [6], VSAM [7], and many others.

Still the problem of video event understanding is 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:

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

The goal of this work 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 above 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. That is, 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 recognize relatively simple events (single actor, known camera angle, pre-segmented event sequences) may identify a discriminating abstraction scheme and utilize a pattern recognition method for event recognition. More involved events (multiple sub-events, 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 [8, 9, 10, 11]. These papers touch only on a subset of the ideas considered here and often consider video event understanding as a sub-area of a related field.

The remainder of this paper is organized as follows: Section 2 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 Figure 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. We devote section 3 to

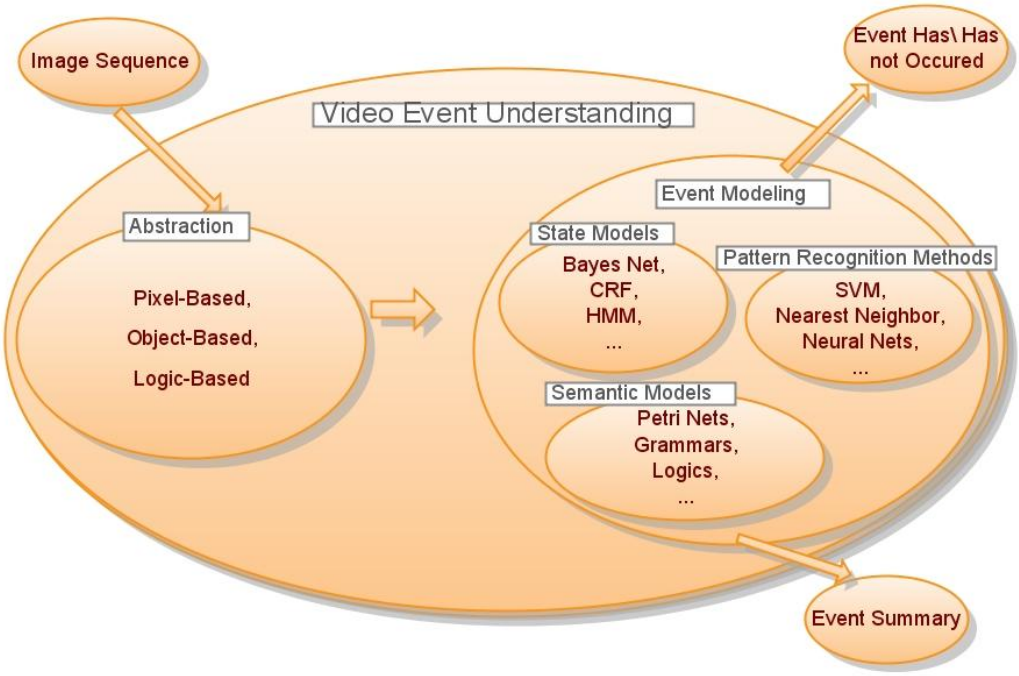


Figure 1: Bird's Eye View of the Video Event Understanding Domain. A video event understanding process takes an image sequence as input and abstracts it into meaningful units. The result of the abstraction is used by the event model to determine if an event of interest has occurred. Output of a video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

description of abstraction approaches.

Event modeling is the sub-domain 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 work. Section 4 discusses the domain of event modeling and offers insight into how approaches within this domain may be classified. Subsequently, Sections 5, 6, and 7 discuss the three main approaches to event modeling prevalent in the literature: Pattern Recognition Methods, State Models, Semantic Models. Each of these Sections discusses the specific formalisms associated with each category of event modeling.

In section 8 we provide an alternative view of the event understanding domain, that of applications. In this view it is more straightforward to observe which techniques of abstraction and event modeling are often utilized together and what are the most explored event domains in this research area. Finally, we conclude the survey in section 9.

2 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 which 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.

One area of research that overlaps in its interpretation of the term "event" is that of perception, a sub-discipline of psychology. In this discipline, several theories exist on what constitutes an event [12, 13, 14]. Computer vision literature has also attempted to categorize occurrences in a video sequence. Nagel [15] defined a semantic hierarchy consisting of the concepts: "change", "event", "verb" and "history". Bobick [16] 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 [17] 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. [11] refer to "object behaviors", defined as time varying data. Aggarwal et al. [8] define "activity/behavior" as a pattern derived from an image sequence. Cohn et al.

[18] define "dynamic behavior" as a set of spatial states whose relationships are defined by temporal logic. Hongeng and Nevatia [19] 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. As an example of this we consider Howarth and Buxton's [20] use of the term "activities" to describe a basic unit of visual features. In a certain sequence these "activities" form a larger semantic unit called "episodes". By contrast, Medioni et al. [21] use the term "activities" to describe higher-level concepts composed of mid and lower-level semantic units. That is "activities" in this context represents a concept similar to "episodes" in the former paper. Furthermore, in Bobick's [16] previously mentioned work the term "activities" pertains to an intermediary step in the semantic hierarchy. This example illustrates the confusion caused by the overloaded terms found throughout the event understanding literature.

In reading the various works some commonalities between the "event" definitions do emerge:

- Events occupy a period of time.
- Events are built of smaller semantic unit building blocks.
- 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 above.

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 [19] "simple/composite event". That is, prefixes will be attached to the term "event" to describe its various

properties, eliminating the need to define different terms to mean types of events with varying properties. We will assign prefixes to represent each of the commonly agreed upon properties of events: sub-event composition, content and temporal composition. Additional prefixes are defined to allow different semantic units(e.g sub-events, super-events) 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" [11], "activity" [20], "action" [22] and "gesture" [23].

The term "sub-event" will be used to refer to component parts of the event. This parallels terms such as "poses" [23], "actions", and "mobile event properties" [19], which provide the building blocks of the event. Sub-events may also be composed of sub-events in a recursive definition.

Complementary to sub-events, super-events are those semantic units composed by the event(s). This corresponds to the terms "episodes" [9] and "scenarios" [19]. As with sub-events, super-events may be defined recursively.

To describe the event composition, we define the terms "composite event" to indicate compositional structure (composed of sub-events) and "atomic event" to indicate no sub-event composition. This definition is in-line with Nagel's [15] description of the semantic hierarchy as well as Bremond's categorization of events [17].

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

Borrowing from Hongeng and Nevatia [19] with regards to temporal composition, we use the terms "single-thread event" and "multi-thread event" to describe the linear sequence and non-linear 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 [16] 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", "behavior" carry some context about what 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.

This proposed terminology is used in subsequent sections of this survey. For example, we can state that Gong's [24] work considers single-thread composite pixel-based events in the "human action" event domain. By contrast, Medioni [21] considers multi-thread composite object-based events in the "aerial surveillance of car behaviors" event domain. The terminology is summarized in Table 1.

We further illustrate the use of this terminology using an example. Suppose one is given a video sequence depicting a pedestrian crossing the street at an intersection. If the pedestrian crossing the street is the event of interest, one could define an event model that recognizes this event as a combination of sub-events such as "`person is walking`" and "`person is in intersection`". Thus, the event will be considered

composite. The sub-event "`person is walking`" may also be composite, that is, it may be composed of lower-level sub-events such as "`legs are moving`". One could also choose the "`person is walking`" sub-event to be atomic. That is, it does not have sub-event composition. The abstraction scheme may be based on tracking the person (i.e. object of interest). Therefore, it can be said that the event is object-based. Furthermore our event of interest, "`the crossing of the street`", has a sequential temporal composition (i.e. sub-events occur in sequence) thus the event can be said to be "single-thread". The event described by this model using our terminology will be referred to as a single-thread composite object-based event. We may describe the event domain as "pedestrian surveillance".

Composition Prefixes

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

Table 1: Event Terminology

3 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 Figure 1) and are an integral part of the event understanding process.

The choice of abstraction is intended to isolate salient properties of the video data especially those that allow useful discrimination between interesting events. Abstraction is thus related to the problem of feature selection. However, feature selection problems usually focus on choosing the most useful of generally simple to extract features (e.g. intensity, edges).

While an abstraction scheme can make use of simple features, many abstraction primitives are more complex aggregations of these simple features (e.g. gradient histograms [25], motion history images [23]) or the output of algorithms that process these features into higher level semantic information (e.g trajectories/bounding boxes).

Abstraction may be a transformation of the low-level input or simply a way of organizing this input. Abstraction approaches may be designed to provide input to a particular event model or to construct informative atomic primitives that can serve as input to a general event model. In this section we will discuss several popular ideas for how to abstract video data.

Along with capturing the important event-discriminating aspects of the video data other main motivations in selecting a particular abstraction scheme are computational feasibility, and ability to complement the chosen event model.

In this section we will discuss three main categories of abstraction approaches: pixel-based, object-based and logic-based abstraction. Each of these approaches is named for the level at which the input is described. Pixel-based abstraction is the category of those abstraction schemes that describe the properties of pixel features in the low-level input. Object-based abstraction approaches describe the low-level input in terms of semantic objects (and their properties). Logic-based abstraction approaches organize the low-level input into statements of semantic knowledge. The following subsections expand on each of these categories with

examples from the literature.

3.1 Pixel-Based Abstraction

In this category we have grouped all those abstraction schemes that rely on pixel or pixel group features such as color, texture and gradient. This scheme is distinguished from other abstraction schemes as it does not attempt to group pixel regions into blobs or objects, but simply computes features based on the salient pixel regions of an input video sequence.

A popular pixel-based abstraction is the organization of low-level input into vectors in an N-dimensional metric space [26, 27, 25, 28, 23, 29, 30]. These kind of representations are used to model many different types of data and methods for classification problems on this kind of abstraction are well understood.

The abstract space may be a selection of image features observed directly from the video frames. In this case evaluation may be done to determine which set of the various possible sets of features is the most useful for discriminating between events of interest (using a particular event model) [27].

A vector representation may also be the result of the application of some mathematical tool such as dynamic time warping (DTW) [26]. A benefit of the vector representation choice of abstraction is that it is fairly general and can be used as input to numerous event models. One drawback of this abstraction approach is that, in some cases, it does not allow for a straightforward semantic interpretation. That is, with the exception of some vector abstractions which allow a meaningful visualization (e.g MHI), the initial abstraction of the video input is meaningless to a human without being processed by an appropriate event model.

The choice of pixel-based abstraction usually follows a researcher's intuition about what are those pixel feature properties that are important for describing video events. Examples of this abstract video input as intuitively meaningful features such as histograms of spatio-temporal gradients [25], spatio-temporal patches [31, 32, 33] and "self-similarity surfaces" [28].

One subclass of these intuitive abstractions that is especially worth noting is that of Motion History Images

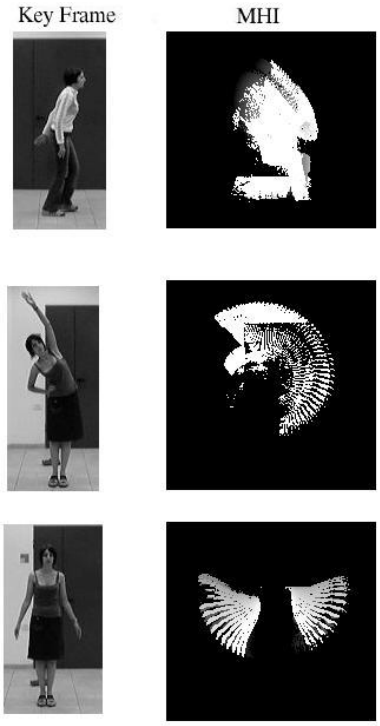


Figure 2: A video sequence containing an event (shown as a key frame in the first column) is abstracted as a single Motion History Image in which the pixel intensity at each pixel represents the amount of motion observed at that spatial location during the video sequence. This is an example of Pixel-Based abstraction (inspired by [23])

(MHI). Originally proposed by Bobick and Davis [23], the MHI is an intensity image that indicates the spatial location of the most recent pixel motion with higher intensity values. Thus allowing a simple abstraction of a the video input that is also easy to visualize (see Figure 2). Other works have expanded on this idea of abstraction [24, 34, 29]

Pixel-based abstraction approaches are used in a wide variety of event domains including: aerobic exercises [23], single person movements [24, 25, 27], multi-person activities [34] and traffic monitoring [29].

3.2 Object-based Abstraction

Object-based abstraction is an 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 [19, 35, 36, 37, 38]. An illustration of this abstraction is shown in figure 3.

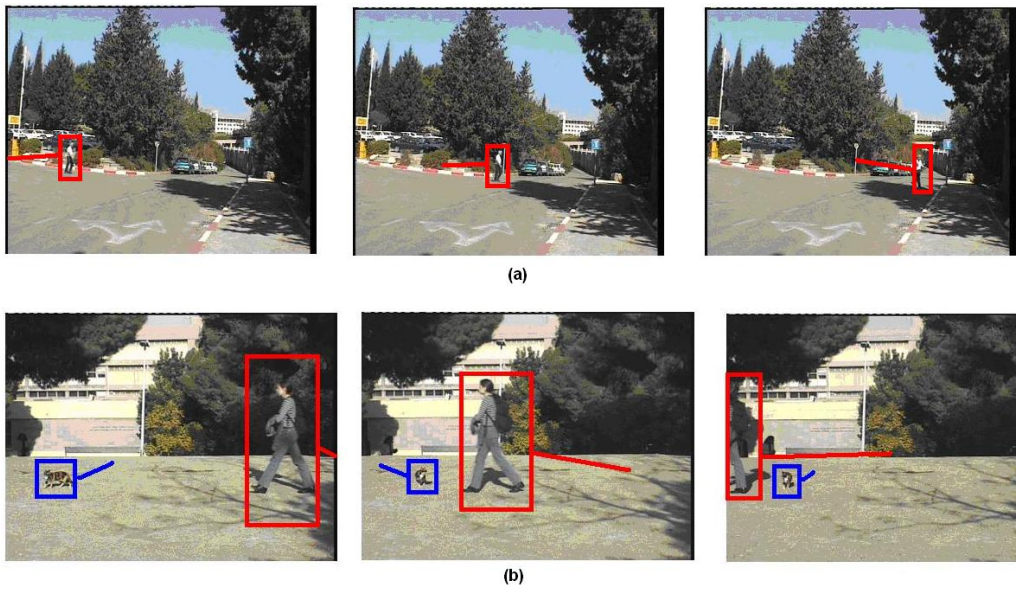


Figure 3: This figure visualizes Object-Based Abstraction. In this type of abstraction scheme objects are located and tracked and the video sequence is abstracted as properties of these object. In (a) a single person is tracked and a bounding box along with a trajectory from its previous location (visualized by a line) are used to abstract the sequence at each frame. In (b) the same scheme is used for two objects (a person and a cat). (inspired by [19])

Object-based abstraction is usually obtained by making use of existing technologies for object detection and visual tracking. These areas are the focus of much attention in the computer vision community and are outside the scope of this work. Interested readers are referred to [39] for more information.

Silhouettes are a popular object-based abstraction used by many event understanding works[40, 41]. Many works further manipulate silhouette data using binning [42], PCA [43], and directionality features [44].

Trajectories are another very popular object-based abstraction approach prevalent in the literature [45, 46, 47]. Trajectory abstraction are often coupled with pattern recognition methods which allow learning a classifier in an unsupervised fashion from training data [48, 49, 50, 51, 52, 53, 54]. Several works study the dimensionality reduction of trajectory abstractions using methods such as PCA[55], ICA [56] and Fourier transform [57].

Trajectory abstractions are used in event domains including: metro surveillance [48], sign language recognition [58], and parking lot surveillance [50].

3.3 Logic-based Abstraction

A type of abstraction which we have dubbed logic-based abstraction is motivated by the observation that the world is not described by multi-dimensional 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.

As in other abstraction schemes we have considered the choice of abstraction is motivated by intuition on what are the important properties of the video input that help discriminate between events.

An example of logic-based abstraction can be seen in work by Siskind [59]. Low-level input is abstracted into line segments associated by kinematic stability concepts such as grounding and support. This abstraction is motivated by the biological model.

Another example is provided in work by Cohn et al [60]. The chosen abstraction scheme focuses mainly on the spatial aspects of the event. A set of qualitative spatial relations is applied to the video sequence and relates important image regions to one another.

The advantage of the logical abstraction is that the representation space after the abstraction is much smaller than the original space of all possible inputs. Also the influence of uncertainty errors on quantitative abstraction approaches is reduced substantially.

In this section we have reviewed three categories of abstraction: pixel-based, object-based and logic-based. Abstraction is the critical first part of the event understanding process. It is an essential precursor to the event models discussed later in this paper.

However, because of the fact that abstraction approaches are closely coupled with event models in the literature not many works provide meaningful evaluation on what is the most powerful abstraction scheme. For example, as discussed above many papers make use of silhouette abstraction coupled with different event models (e.g. Nearest Neighbor, HMM, CRF, etc..) to attempt to classify the same class of events. This analysis, while useful for evaluating the particular event model, tells us nothing as to how influential

the choice of the abstraction scheme is. A useful comparative evaluation of different abstraction schemes using the same event model to classify the events within the same domain is difficult to find in the literature. An exception to this, is found in the sub-set of event understanding often called "action recognition". This event domain generally contains a set of single person events captured from a know camera angle. Recently, the investigation of this domain has gained popularity partly because of the availability of public databases[40, 41], but largely because of the need to compare and contrast different abstraction schemes and their effect on event recognition performance [61, 62, 63, 64, 65].

It is intuitive that the more complex the choice of abstraction the more useful it will be when it comes time to be reasoned on by an event model. However, the computation of some very complex abstractions is prohibitive. These include heavy tracking approaches, multi-dimensional histograms, and others. Thus, for papers that propose full systems of event understanding we often see that more light-weight tracking approaches such as background subtraction in contrast to particle filter or other more computation intensive trackers.

4 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 those events as they occur in the video sequence input.

Many works in event understanding literature focus on this problem of finding a suitable modeling formalism, an event model, to describe the occurrences of interest in a particular domain.

Event models have many aspects and, hence, they may be categorized in several different ways. One such categorization, is the distinction between deterministic and probabilistic event models. Examples of

event models in the deterministic category include Finite State Machines (FSM), Grammars and Petri-Nets, while Bayesian Networks, Hidden Markov Models (HMMs), Conditional Random fields (CRF) and Stochastic grammars all associate a probability score with an event occurrence.

Some event modeling formalisms can be expressed graphically. That is, they can be defined using the mathematical notion of a graph, which utilizes such concepts as "nodes" and "arcs". Yet others, known as "probabilistic graphical models" (or sometimes simply "graphical models"), comply with a stricter definition that requires a graph in which nodes represent stochastic variables and arcs represent dependencies. Petri-Nets and FSM event models lie in the former category while Bayesian nets, HMMs and CRFs are in the latter. There are also models which cannot be fully represented graphically, including multi-variable Support Vector Machines (SVM) and Grammar models.

Another distinction that can be drawn among event models is between generative and discriminative models. In a statistical (probabilistic) framework, discriminative models are defined as those models that directly model the posterior probability for use in classification tasks. This is in contrast to so-called generative models which model and learn the joint distribution over all possible observations and labels and then use this information to extract the posterior probability (i.e. the distribution over class labels given the observations). Intuitively, a discriminative model is only concerned with the decision problem, while a generative model attempts to capture the phenomenon that "generates" both events and observations. Outside of a statistical framework, the distinction between generative and discriminative models is less well-defined.

Works contrasting the application of discriminative and generative of models [66] have shown that discriminative models have lower error rates as the number of training examples becomes very large. However, they also found that generative models tend to converge to their optimal performance much quicker (with less training examples) than their discriminative counterparts. That is, if a smaller body of training data is available the generative approach might be favorable. Additionally, generative models have been found to be more flexible to incomplete training data and new classes and may be better suited to learning complex

patterns [67].

The above 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 above, 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 [17].

"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. The first class of these models, we have named "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 FSMs which conceive video events as fully observable processes of a set of known states. These states may have semantic meaning taken from domain knowledge. The transitions between these states are also specified using knowledge of the domain. This class also includes the set of probabilistic graphical model approaches. These approaches factorize the state into variables (the structure of the graph) according to some semantic knowledge of the domain. The existence (under some structural assumptions) of efficient algorithms for the learning of parameters from training and the inference of hidden node values

also motivate the choice of probabilistic graphical models to model video events. This enables some degree of automated learning of abstract parameters. However, as previously stated, the structure of this type of model is usually specified with some knowledge of the event domain. This knowledge is usually implicit and imposed on the model indirectly (e.g. assigning a semantic label to each state). This class of models largely consists of approaches fully specified using a combination of knowledge based structure specification and automatic parameter learning from training data. In addition to their flexibility, generative models also exhibit adaptive properties. This is in contrast to discriminative models which must be completely relearned for each new decision class added. These properties are probable reasons why the work in this category of event modeling largely favors generative models, although newer work proposes discriminative approaches (e.g. CRF).

Higher-level semantics include sub-event ordering information (including partial ordering), complex temporal, spatial and logical relations among sub-events. Also important 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 we have named "Semantic Models" have been proposed which enable explicit specification of these complex semantic properties. Among these are Petri-Nets 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 modelled by the approaches in each of our categories.

5 Pattern Recognition Methods for Event Recognition

The class of techniques in this section are 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, formulated as a traditional pattern recognition problem. This class includes well studied approaches including Nearest Neighbor, Support Vector Machines, Neural Networks.

The main advantage of the classifiers in this category is that 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. The representational considerations are usually left to the abstraction scheme associated with the event recognition method in the particular event understanding application.

Nearest neighbor(NN) is a well-studied pattern matching technique for classification. An unlabeled example is labeled using its "nearest" labeled neighbor in the database. K-NN is a variation of nearest neighbor where the K nearest neighbors vote on the label of the test example. The notion of closeness is defined by a distance measure decided upon during the model specification. [68]

Nearest Neighbor models can naturally be coupled with a vector abstraction of the data as well as other abstractions assuming an appropriate distance metric between abstraction has been defined.

The choice of the distance measure in the various works utilizing a NN event model for event understanding is usually chosen with respect to the abstraction of the video input. Clever choices of distance measures allow for better event discrimination. Choices range from simple distance measure such as Euclidean [40, 69] and Chi-Squared [25] to more complex choices such as Linear programming based distance [70]. Some choices for distance measures are event-domain dependent such as spatio-temporal region intersection [71] and gradient matrix of motion field [72].

So called template matching techniques [23, 24] also utilize a NN event model.

The NN model does not inherently allow representation of important properties of video events. However, the more dense the coverage over the space of possible abstraction the better the model performs in

classification tasks. Unfortunately, as the database of examples grows so does the recognition time of the algorithm. Because the abstraction of video events is often high-dimensional, a sufficiently dense NN event model is still intractable for recognition. This is especially true when a complicated distance measure is utilized. Furthermore, in many domains it is not possible to collect or store such a database of examples and many approaches utilizing NN use a sparse database of examples to increase the efficiency of both storage and recognition.

Nearest Neighbor event models have been used in event domains including: aerobic exercises [23], single actor actions [25, 40], and Ice skating maneuvers [72].

Support vector machines (SVM) [73, 74]are a group of models designed to find the optimal hyperplane separating two classes in a multi-dimensional space.

Each SVM may be associated with a kernel function to transform the multidimensional space (which is possibly non-separable) into a higher dimensional space which is more likely to be separable.

The kernel function is usually chosen using knowledge of the domain. Additionally, abstraction schemes coupled with SVM event models are often chosen to be quite complex. In addition to a classification from a SVM model we may also get a confidence score. This score is equivalent to the distance in the decision space between a particular example and the decision boundary.

The separating hyperplane is determined in a training phase using labelled training data by solving an optimization problem maximizing the distance between examples with different labels in this higher dimensional space.

The two class SVM can be generalized to a multi-class decision problem. Pittore et al. [75] uses such a scheme for the classification of object-based atomic events in the event domain of office surveillance.

In the event understanding literature SVMs utilize a wide variety of kernel functions including: linear [75], tree-kernel [76], polynomial [77], Radial basis function (RBF) [78].

Like the nearest neighbor model the SVM is reliant on the abstraction scheme associated with it to represent the salient properties of video events. In and of itself it does not have the capacity to capture

these properties. Furthermore, the high-dimensional "decision space" representation created using the kernel function, while allowing better separation of the examples, is abstract and not meaningful outside the context of the SVM decision.

SVMs have been shown to be very powerful for many classification tasks. However, the training phase of SVMs (i.e. calculating the separating hyperplane) can be lengthy in high-dimensional problems.

Additionally, because of its inability to represent event properties such as temporal evolution, an SVM event model is often coupled with complex abstraction schemes which may themselves be difficult to compute efficiently.

SVM classifiers are coupled with various abstraction schemes including both pixel-based [75] and object-based [50] abstractions. SVM classifiers are used in such event domains as parking lot surveillance [50], single person actions [77], and kitchen activities [76].

The general approach to combining classifiers to enhance the accuracy of the final classification is called boosting [79]. Boosting is not in itself an event recognition technique but rather is often applied to pattern recognition classifiers to yield an improved event classifier. We discuss this topic here because boosting approaches are often applied to yield event classifiers for video event understanding.

Most of the popular boosting methodologies are used to construct event models in the literature of video event understanding including: AdaBoost [32, 80] and LPBoost [81]. Other, less well-known, boosting methodologies such as "Knapsack Boost" [82] are also in use.

In some cases the properties of video events have guided the choice of boosting scheme (i.e. how to combine classifiers). TemporalBoost [83, 84, 85] uses the idea of dependency between frames in a video sequence.

Boosting based classifiers are efficient for recognition, but require a lengthy training phase. Additionally, these type of event models do not capture intrinsic properties of video events and thus depend on their associated abstraction scheme to capture these.

Boosted event classifiers have been used in event domains such as single person actions [83, 84], metro surveillance [85], and office surveillance [82].

Another well known pattern recognition technique is the Neural Network. This type of classifier simulates the biological system by linking several decision nodes in layers. This approach has been used to solve complex decision problems and recognize patterns in many domains. Early work in video event understanding explored the use of Neural Networks as an event classifier [86].

The strength of the pattern recognition approaches to event recognition is that they are well understood and are straightforward to both formalize mathematically and implement practically. They also allow learning of the model from the data. However, they do not incorporate semantics, that high-level knowledge that humans use to define video events, such as meaningful state space factorization, temporal extent, and composition in both space and time. Thus, these models are most frequently applied to the recognition of atomic events. The limitations of pattern recognition event methods are addressed by other classes of event models we will explore in subsequent sections.

6 State Event Models

"State" event models are a class of formalisms which are chosen using semantic knowledge of the state of the video event in space and time. Reasonable assumptions about the nature of video events have lead to the development of each of these formalisms. Each of these capture an important aspect or property of video events through their use.

State event models improve on pattern recognition methods in that they intrinsically model the structure of the state space of the event domain. This is used for capturing both the hierarchical nature and the temporal evolution of state, that are inherent to video events. 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, not only capture some inherent properties of video events, but are also well studied and mathematically well formulated to allow for efficient algorithms 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.

A category closely associated with State Models is that of generative models from statistics and machine learning. This is the class of models that attempts to model the phenomenon that "generates" the observation. However, as we shall see not all State event modeling formalisms are generative.

State modeling formalisms include: Finite State Machines (FSMs), Bayesian Networks (BN), Hidden Markov Models (HMM), Dynamic Bayesian Networks (DBN), and Conditional Random Fields.

6.1 Finite State Machines

Finite State Machines (FSM) [87], also known as Finite State Automata, are a formalism useful for modeling the temporal (especially sequential) 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 computationally efficient solution to reasoning about event occurrences. 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. FSMs are also a well studied formalism 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 [88], single actor behavior [89], multiple person interaction [19], and aerial surveillance [21].

The inherent ability of the FSM formalism to capture sequence allows it to be associated with different abstraction types including pixel-based [88] and object-based abstraction [19, 21, 90, 91].

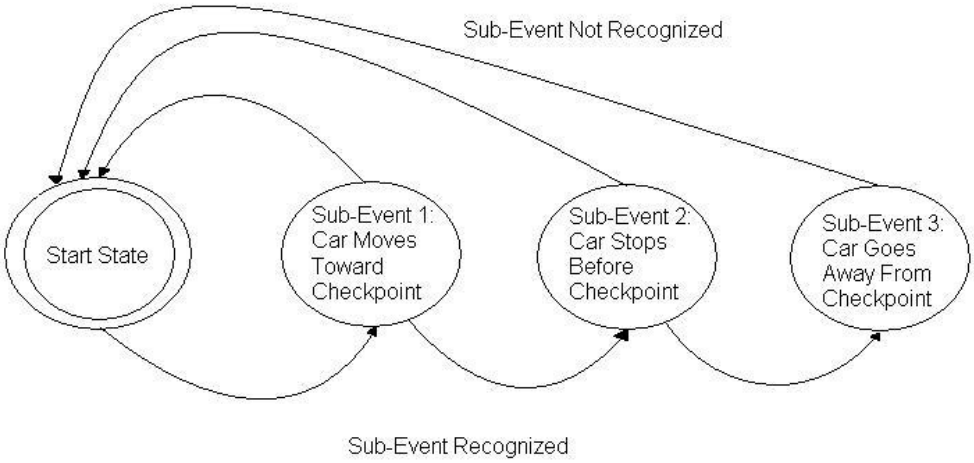


Figure 4: FSM event model used on the "car avoids checkpoint" event a la [21]

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 [92]. Some work has also been done on inferring an FSM event model from user specification [90, 91].

FSMs are a useful tool in video event understanding because of their simplicity, ability to model temporal sequence and their learnabillity.

Extensions to the FSM have been proposed to capture the hierarchal property of video events [93, 94, 89]. Uncertainty in video events has also been addressed through the introduction of probabilities into the FSM framework [94]. It should be noted that in some areas of the event understanding literature the terms of "HMMs" (see section 6.3) 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).

6.2 Bayesian Networks

In order to deal with the inherent uncertainty of observations and interpretation which exists in video events, Bayesian Network event models utilizing probability as a mechanism for dealing with uncertainty,

have been proposed.

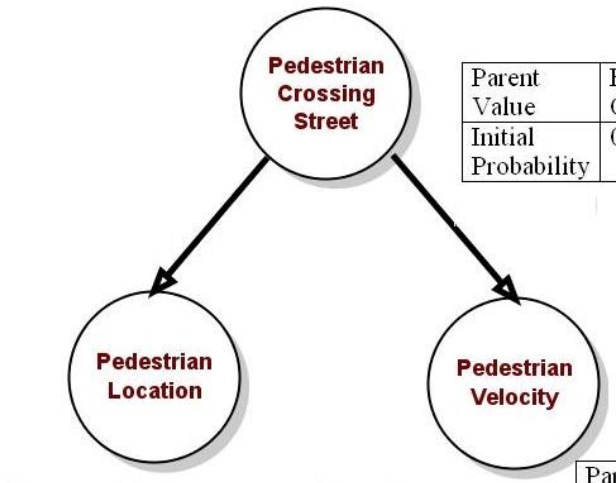
Bayesian Networks (BN) (also known as probabilistic networks, Bayesian Belief networks, or independence diagrams) are a class of directed acyclic graphical models. Nodes in the BN represent random variables which may be discrete (finite set of states) or continuous (described by a parametric distribution). Conditional independence 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 readers are referred to [95, 96].

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/hasn't 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.

While the general inference (i.e. the estimation of hidden variables given observed variables) problem in BNs is NP-hard, efficient algorithms for inference exist under certain BN structure assumptions [96]. BN model parameters may also be learned from training data. Additionally, network structure learning has also been explored in the literature. [97]

As an example of the standard approach to event modelling using BN we consider the "pedestrian crossing street" event. We will choose this event to be atomic, that is having no sub-event composition. Therefore we will have the event decision based on abstraction primitives. We will construct a simple BN as pictured in Figure 5 in which the top node will represent the occurrence of the event (i.e. discrete binary variable). The bottom nodes in the figure correspond to abstraction primitives. The first node corresponds to location of the person and the second to direction of the person. For simplicity we will consider these nodes to be discrete. The first node can take on the values "inside intersection" or "outside intersection". The second node can take on the values "not moving" "moving towards sidewalk" or "moving not towards



Parent Value	Event Has Occurred (E1)	Event Hasn't Occurred (~E1)
Initial Probability	0.85	0.15

Parent Value	Inside Intersection	Outside Intersection
E1	0.85	0.15
~E1	0.10	0.90

Parent Value	Not Moving	Moving Towards Sidewalk	Moving Not Towards Sidewalk
E1	0.01	0.85	0.14
~E1	0.4	0.3	0.3

Figure 5: A naive Bayesian Network depicting an atomic event,"pedestrian crossing street"

sidewalk". As we observe the abstracted video input the value of these nodes will be available at each frame. We still must determine the parameters for the BN. These parameters can be set using semantic domain knowledge (i.e. we know that during "pedestrian crossing street" the person is "inside crosswalk" with a high probability). The parameters can also be learned from training data. In the training data we will have knowledge of the event variables value which will allow us to set the parameters according to observed frequency. In the case of a more complex BN with multiple hidden variables the expectation maximization (EM) algorithm is used for parameter estimation. Once the parameters are estimated we can observe test data and determine whether our event is occurring or not by calculating the joint probability and then marginalizing out other variables to obtain the marginal probability over our event variable.

This method achieves a probability score indicating how likely the event is to have occurred given the input.

BN models do not have an inherent capacity for modelling temporal composition which is an important aspect of video events. Solutions to this problem include single-frame classification [98] and choosing abstraction schemes which encapsulate temporal properties of the low-level input [99, 100]. Naive Bayesian networks, such as the one pictured in Figure 5, appear often 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

sub-event 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 [98, 101], and indoor surveillance of people [100]. More complex BNs have been used in event domains such as parking lot surveillance [102] and recognizing American football plays [99]. Although these networks are large they retain a structure that allows for efficient inference.

Inference algorithms, such as belief propagation, for calculating the marginal distribution (i.e. belief) of a particular node run in time polynomial in the number of states per node in the BN under these structural assumptions.

The BN agent approaches in the literature are more commonly associated with object-based abstractions [98, 101, 100, 99, 102].

Modeling the hierarchy of video events is straightforward within the BN framework. Hongeng and Nevatia [103] model the semantic hierarchy using BN layers. Each layer corresponds to a higher-level semantic units.

A more recent group [33, 104, 105] of works make use of Bayesian Network models adapted from the text and image mining communities. These approaches are also known as Probabilistic Latent Semantic Analysis (pLSA) [106] 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 commonly associated with pixel-based abstractions (e.g. "cuboids" [31, 33])

Bayesian networks are powerful tool in factorizing the state space into variables using semantic knowledge of the domain and specifying a joint distribution over all possible values of these variables succinctly. This formalism naturally models the hierarchical and semantic state of video events. The probabilistic output of BNs is useful for addressing uncertainty. This formalism also allows computationally tractable solutions for inference. The main shortcoming of the BN model is in modelling the temporal aspects of video events.

6.3 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 event model.

Hidden Markov Models (HMM) are a class of directed graphical models extended to model the temporal evolution of the state. The HMM has a specific graph structure associated with it. One variable, representing the hidden state, and one variable, representing the observation, comprise a single "time slice". The "time slices" represent the evolution of the process (event) described by the model over time. Intra-slice arcs indicate the dependence of the observation variable on the state variable. Inter-slice arcs connect the state variable in the previous slice to the state variable in the current slice. This structure describes a model where the observations are dependent only on the current state. The state is only dependent upon the state at the previous "time slice" (the Markov assumption). This structure (see Figure 6a) is imposed to allow efficient inference algorithms.

Since the HMM structure is fixed and repetitive we can define the likelihood of long sequence of states (and corresponding observations) by specifying the following parameters: the initial (distribution over initial state values), the transition (distribution over the state given the previous state), and the emission (distribution over observations given the state) probabilities. The number of parameters required to specify these probabilities depends on the number of states and observation symbols, which are usually determined empirically.

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 the reader is referred to [107, 108].

A common use for 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 not semantically meaningful 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.[109]

A number of early works in the literature employ this approach in the event domains of tennis stroke recognition [110], Sign Language and gesture recognition [111, 112], single-person actions(e.g. "walking", "kneeling") [113]. 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. That is, 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 modelling 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 (Figure 6).

Multi-Observation Hidden Markov Models (MOHMM) [22] use multiple variables to represent the observation. The variables are casually dependent on the state variable, meaning they are conditionally independent of one another given the state. 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. Parameter estimation of MOHMMs is similar to that of HMMs except that additional emission probabilities for each additional observation variable must be defined.

Another approach to reducing the parameters to be learned is altering the network topology (specifically which states are reachable from which other states) [114]. 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) modelled 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 sub-events 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 Hidden Markov Models (PaHMM) [115] the multiple chains of the state are modelled as separate HMMs each with its own observation sequence. Coupled Hidden Markov Models (CHMM) [116, 117, 118] where 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 Multi-Level Hidden Markov Models (DML-HMM) [119] extend the coupling concept by attempting to learn the dependencies between the hidden state chains. That is, the state space is reduced by both separating the state into multiple variables and simplifying the topology.

As expected, these extensions are used in event domains where there are several elements participating in the events of interest including sign language [115], Tai-Chi gestures [116], multiple person interactions [118], and airport tarmac surveillance [119].

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

However, as the topology becomes more complex the efficient exact algorithms associated with the "pure" HMM structure are no longer applicable and must be replaced by approximation algorithms. An experimental comparison of MOHMMs, PaHMMs, CHMMs and DML-HMMs can be found in [22].

Another extension to the basic HMM structure is motivated by the long-term temporal dependence of state variables within a video event. That is, 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

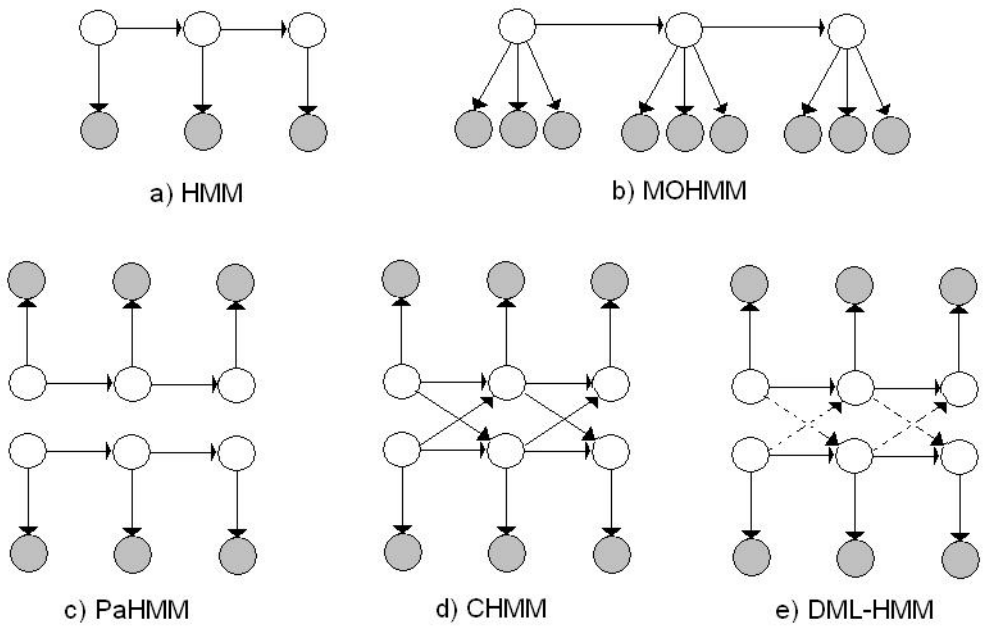


Figure 6: Factorization of the state and observation space. Shaded nodes indicate observed variables.

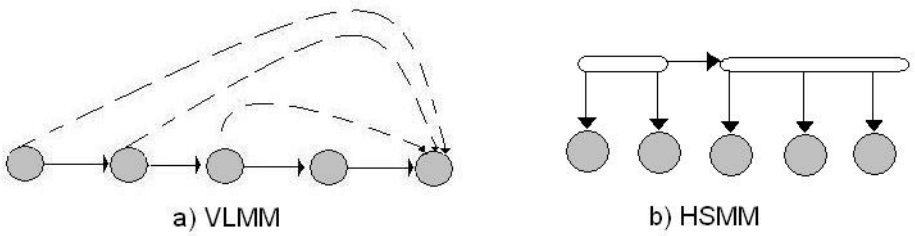


Figure 7: HMM extensions to capture long-term temporal dependency

dependencies or occlusions and other phenomena that cause errors in state estimation. Figure 7 visualizes some of these approaches.

N-order hidden Markov models deal with this problem by amending the Markov assumption to consider the N previous states. Variable Length Markov Models (VLMM) [120, 121] calculates the optimal level of temporal dependence using a divergence criterion. Hidden Semi-Markov Models (HSMM) (sometimes called Semi-HMMs) [122] allow each state to emit multiple observations. That is along with the state variable at each time there will also be a duration variable (observation length).

Parameterized HMMs [123] introduce extra parameters to model events that may have variance that does not affect the event classification. These parameters may have a semantic meaning and may or may not

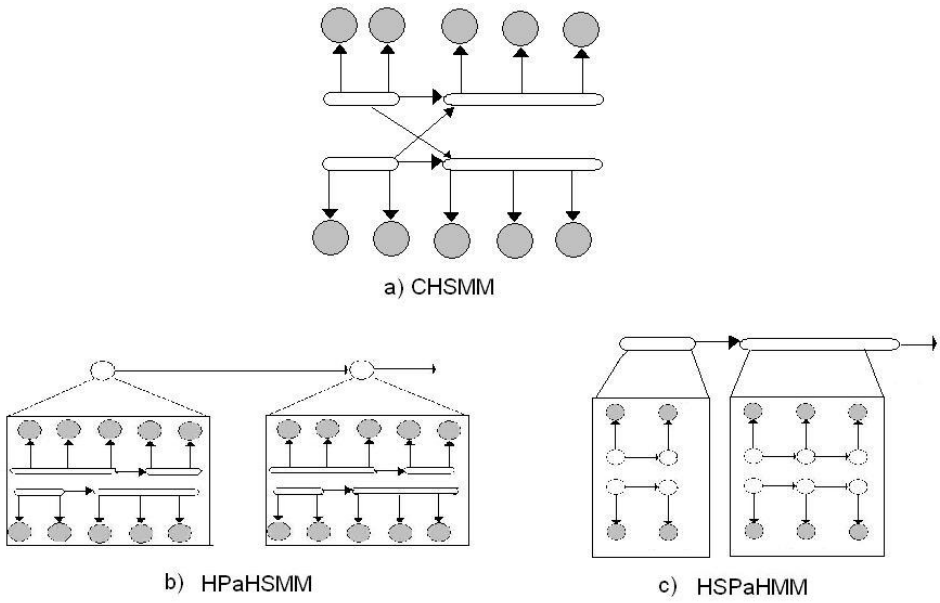


Figure 8: "Hybrid" HMM models capturing the intrinsic properties of video events.

be measurable. These parameterizations prevent classification errors due to variance in this parameter. Secondly, in estimating this parameter extra information about the event is obtained.

Several HMM extensions have been proposed to incorporate the inherent hierarchical composition of video events into the event models. In Hierarchical Markov Models (HHMM) [124, 125], each possible state is represented by a lower-level HMM. In a similar approach Oliver et al. [126] uses 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. That is, 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) [127], Hierarchical Semi-Parallel Hidden Markov Models (HSPaMM) [128] and the Coupled Hidden semi-Markov models (CHSMMs) [129]. Figure 8 illustrates some of these "hybrid" HMMs.

Hidden Markov Models are among the most popular formalisms for modelling video events. As the event being modeled becomes more complex and has interesting properties such as long term dependence and

hierarchical composition, the basic HMM has evolved complicated variations. These extensions attempt to introduce more and more semantics into the formalisms. Unfortunately these semantically enhanced models often come at the cost of tractability. The structural constraints that afford tractability are the original motivation for adopting the HMM and must be adhered to in order to have a practical event model. This means finding a balance between a model that captures the properties of video events well and a model that is realistic for application.

6.4 Dynamic Bayesian Networks

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 which allows such general factorization while still capturing the temporal evolution of state is the Dynamic Bayesian network.

Dynamic Bayesian networks (DBN) generalize Bayesian Networks(BN) with a temporal extent. They can be described formally by intra-temporal dependencies and inter-temporal dependencies. The former is described as a "static" BN and the latter as a special two-slice BN. In this specification the Markov assumption is preserved. 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 semantically meaningful or classification performance enhancing ways. This, however, often comes at the cost of computational tractability. Approximation techniques are usually used to perform learning and inference on general DBNs.

Because exact solutions are not available, general DBNs are not often used for modelling video events. Instead specific DBNs with structural assumptions that yield computationally tractable algorithms (such as the HMM and its variants) are often used.

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 [130, 131, 132, 133].

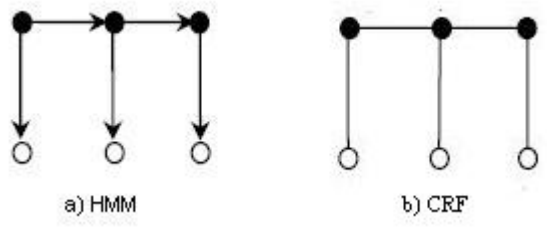


Figure 9: A CRF is a discriminative undirected graphical model with structure inspired by the HMM

DBN approaches have been applied in event domains such as the office environment [130] , assisted living [131, 132, 134], and surveillance of people [135, 133].

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 [135], restricting state transition topology [134, 131, 132].

Apart from learning the DBN parameters from the data many recent work attempt to learn aspects of the model such as model structure [130], abstraction scheme [136], and the number of states each variable takes on [133].

Dynamic Bayesian Networks 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.

6.5 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 as such there is no need for such restrictive assumptions. The adoption of conditional random fields as event models is based on this idea

Conditional Random Fields (CRF), recently introduced in [137], are undirected graphical models that

generalize the Hidden Markov Model by putting feature functions conditioned on the global observation in the place of the transition probabilities. The number of these functions may be arbitrarily set. 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 [138].

In event modelling, CRFs have consistently been shown to outperform HMMs for similar event recognition tasks [139, 140]. 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 [42, 141, 142]. 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 6.3.

CRFs are a recently popular event modeling formalism that is straightforward to apply in cases where HMMs have been applied before and achieve better event recognition result. The tradeoff incurred by this is a significantly longer training time.

7 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 sub-events. For instance, a "Bank Attack" event may be partially defined as ("robber enters zone" during "cashier at position") before ("cashier at safe" during "robber at safe") [143]. This is an example of an event definition which makes

use of temporal and spatial semantic relationships. To allow these kind of semantic relationships to be represented and recognized several event modeling formalisms have been proposed. We have grouped these in the category of "Semantic Event Models".

The class of Semantic Event Models contains event modelling approaches that do not aim to define the entire state space of the event domain as in "State Model" approaches. Semantic knowledge is still used to construct the event model. However, the event model is defined in terms of semantic rules, constraints and relations. That is, there is a large degree of overlap between how humans describe what constitutes an event and how it is defined within these modelling 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 sub-events 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. That is, 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 [143, 36, 37, 23].

In the following sections we will explore such semantic event modelling formalisms including: Grammars, Petri-Nets, Logics and Constraint Satisfaction approaches.

The commonality of all these approaches is that the event model is fully specified in terms of high-level semantic concepts.

7.1 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 [144] are well studied and have been used in several domains including Speech Recognition [145] and Computer Vision [146].

Grammar models specify the structure of video events as sentences composed of words corresponding to abstraction primitives. The grammar formalism allows for mid-level semantic concepts (parts of speech in language processing). In the event model context, these mid-level concepts are used to model composing sub-events. This formalism naturally captures sequence and hierarchical composition as well as long-term temporal dependencies.

When we discuss semantic grammar models in this section we are referring to those approaches that infuse semantics into the grammar rule description rather than those grammars that are simply an equivalent representation of a finite state machine (or HMM).

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

In the domain of video event modeling, grammars are used as follows: Terminals correspond to abstraction primitives. Similarly, non-terminals may correspond to semantic concepts (i.e. sub-events). 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 (as defined by Chomsky's hierarchy of grammar models [147]) efficient polynomial time algorithms exist for parsing [148].

Deterministic semantic grammar models have been used in several event domains including object manipulations [149] and two-person interactions [150].

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 with the same asymptotic time complexity [151].

Stochastic grammars have been used in event domains such as parking lot surveillance [152], card game surveillance [153], complex task recognition (e.g. Japanese tea ceremonies)[154, 155], complex motions [156, 157], and human actions [158].

It is interesting to observe that the event domains in which semantic grammar models are utilized, in general, contain more complex events whose variance in appearance is very large. As we have previously noted, it is the insertion of semantic knowledge into the structure of the event model that allows representation and recognition of these complex events.

Attribute grammars, introduced by Knuth [159], 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 [160] to classify single-thread atomic object-based events in a parking lot surveillance event domain. If the production rule predicates are chosen to be probabilistic, attribute grammars can be considered a special case of stochastic grammars.

Table 2 shows an example attribute grammar describing events in a parking lot scenario. The start symbol is the non-terminal "`PARKING_LOT`". Two non-terminals corresponding to events are "`DROPOFF`" and "`PICKUP`". Terminals represent atomic sub-events. Examples of these are "`person_appear`", "`car_disappear`", and "`carstart`". A set of attributes is associated with each of these terminals. We can then condition

Production Rule	Attribute Rule
Parking Lot → PICKUP | DROPOFF	
CARSTART → carstop carstart CARSTART	
CARSTART→ `carstop carstart`	
CARSTART→`carstart`	
CARSTOP→ `carstop carstart` CARSTOP	
CARSTOP→ carstop	
CARSTAND→ `car_appear carstart` CARSTOP	NotInside(X_1.loc,FOV)
DROPOFF → CARSTAND `person_appear person_disappear` CARSTART	Near(X_2.loc, X_1.loc), Near(X_3.loc,BLdgEnt)
PICKUP→ `person_appear person_disappear` CARSTART `car_disappear`	Near(X_1.loc, BldgEnt), Near(X_3.loc,X_2.loc), NotInside(X_4.loc,FOV)

Table 2: An attribute grammar describing events in the parking lot scenario similar to [160]. Terminals (lower-case letters) correspond to atomic sub-events and non-terminals (upper-case letters) correspond to events and composite sub-events. A location attribute(loc) is associated with each terminal. The notation X_i refers to the ith term on the right hand side of the production rule. Predefined semantic locations are defined by keywords such as BldgEnt (i.e. the entrance to the building) and FOV (i.e field of view)

each of the production rules on these attributes' values. An example of this can be seen in the rule for "`DROPOFF`". The production rule requires the sequence: "`CARSTAND`" (a non-terminal representing the composite sub-event of a car standing in place), "`person_appear`", "`person_disappear`", "`CARSTART`" (a non-terminal representing the composite sub-event of car starting to move). Aside from enforcing this sequence the production rules also enforces the semantic constraint that the location where the car is standing be near the location where the person appeared. This is done by comparing the "`loc`" attribute associated with each component of the right hand side of the production (using the "`Near`" predicate). Although non-terminals do not have attributes explicitly associated with them it is straightforward to derive attributes from their composing terminals.

Due to the inherent non-sequential temporal relationships in many video events, particularly those defined using semantics, many works have attempted to introduce these relations into the grammar event models [152, 153, 150, 161, 157].

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 [162, 163, 164, 165], 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.

Grammar models are well adapted to represent sequence and hierarchical composition in video events. Long-term dependence is also straightforward to express in a grammar model. Stochastic grammars allow reasoning with uncertainty and error correction methods. Attribute grammars allow further semantic knowledge to be introduced into the parsing process. Temporal relations other than sequence are not naturally represented by the grammar model, though there have been extensions to allow capturing these relations to some extent. However, the representation of these complex relations is not straightforward in the grammar formalism and are more naturally represented in other semantic event modelling formalisms such as Petri Nets (see next section).

7.2 Petri Nets

The non-sequential 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 Petri Nets (PN) formalism allows such a graphical representation of the event model and can be used to naturally model non-sequential temporal relations as well as other semantic relations that often occur in video events.

More formally, PNs (introduced in [166]) 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 that transition node (those place nodes with directed arcs going to the transition node) have tokens. Enabled transition nodes may "fire" and effect 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 which imposes 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 interested readers are referred to [167, 168, 169].

In video event understanding, PN event model approaches can generally be categorized into two classes: Object PNs and Plan PNs [170]. We distinguish these approaches by the design choices made in constructing the event model.

The Object PN event model is used to describe single and multi-thread composite object-based events. 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 object 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 [35, 171] and people [37] surveillance. Figure 10 illustrates an example of the Object PN model applied to a left luggage scenario. Each token corresponding to a detected object is inserted into the model at the "`root`" node and propagated onward according to enabling rules on the transition nodes. Several events of interest in this domain are represented as transition nodes in this model. Some of these events may be viewed as sub-events as they must necessarily occur to reach a state that allows recognizing another event. An example of this is the "`person_went_away_from_luggage`" event which is a prerequisite for the "`person_abandoned_luggage`" event in the figure. Gray rectangles in the figure indicate stochastic timed transition whose parameters can be estimated from training data.

Plan PNs are another approach to event modelling that represents each event as a "plan" of sub-events. Each event is represented as a plan, a number of sub-events connected in such a way as to enforce the

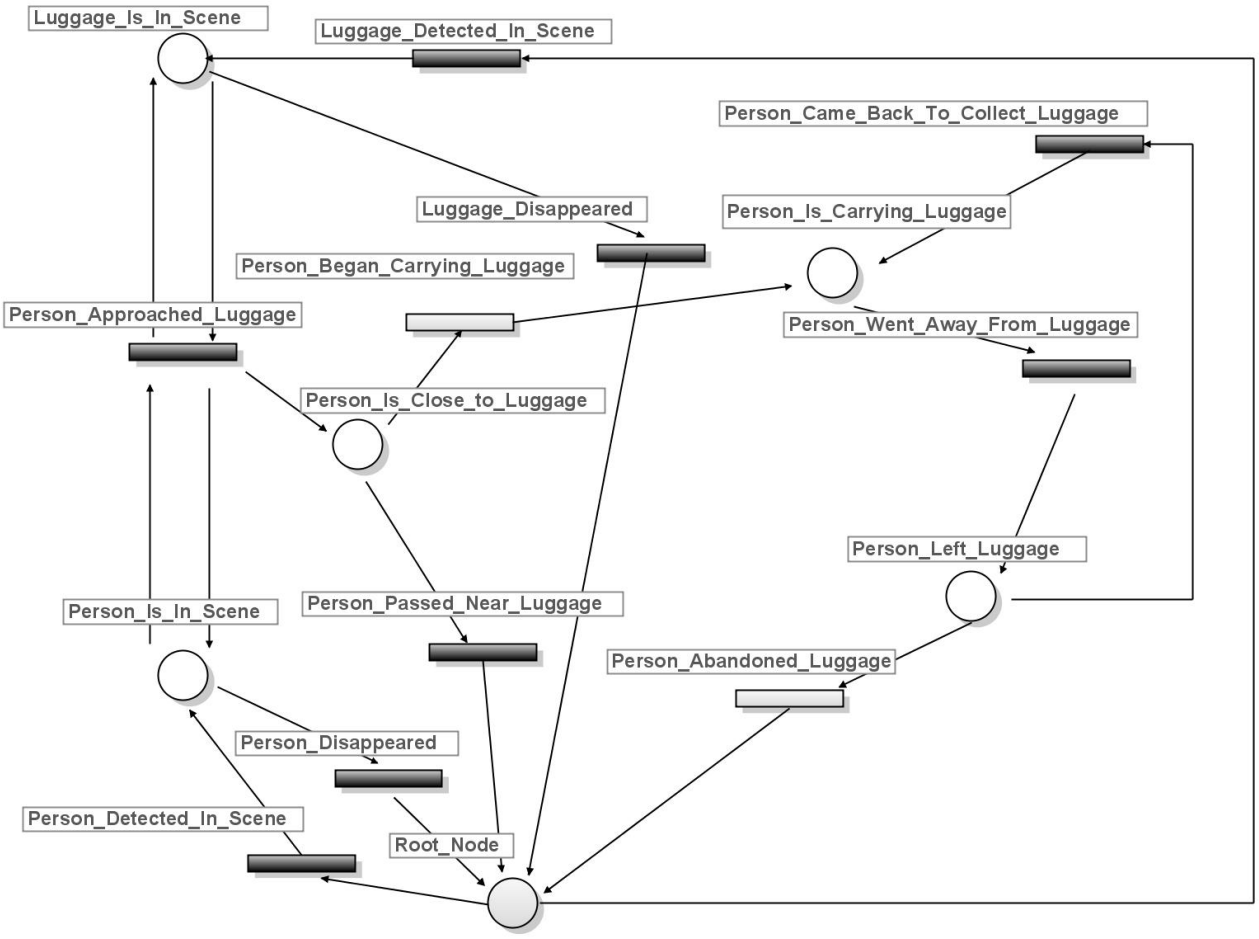


Figure 10: Example of a Object PN Model for the Left Luggage Domain.

temporal and logical relations between them. Each sub-event 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 sub-events 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 Petri-Net event models have been applied to several event domains including: parking lot surveillance [172], people surveillance [173], and complex gesture recognition[174].

An example of a Plan PN model is illustrated in Figure 12 using the "`Car is Parking`" event. Our sub-events will be "`Car enters parking lot`", "`Car stops in parking spot`". These sub-events will be represented by place nodes. An initial transition node T1, intended to detect the first sub-event, requires an object to be a car and to be located in the parking lot, to fire. Once this sub-event is detected(the conditions on transitions T1 are met) a token is placed in the place node corresponding to the "`Car enters parking lot`" sub-event. The state of the PN model now indicates which part of the observation must be queried for the next sub-event, namely the properties described in the next transition node's enabling rule, the proximity of the car to the parking spot and the car's speed. Once the conditions on these properties (transition T2) are met a token will be placed in the place corresponding to the sub-event "`Car stops in parking spot`". Once this sub-event occurs we can trivially declare that our event, "`Car is Parking`" has occurred. To maintain the reference to the object we utilize an extra "condition" place. This place ensures we are referencing the same object in each transition. A logical AND fragment enforces this constraint on the "sink" transition of the event model. It is worthwhile to note that even before the "sink" transition of the plan is reached, there is a semantic notion of how "close" the event is to completion.

Some extensions to the PN formalism event model include timed transitions to allow representation of duration constraints [175], stochastic timed transitions for dealing with uncertainty within duration constraints [37], and associating probabilities with tokens to cope with uncertain observations [173].

In most known works employing PN models for the representation and recognition of video events an object-based abstraction is used [175, 37, 173, 172].

The high-level semantics captured by PN event models include temporal, spatial and logical composition, hierarchy, concurrency and partial ordering. PN event models are usually specified manually using knowledge of the domain. Sub-event temporal and logical relationships are related through known PN fragments which correspond to Allen's temporal relations (Figure 11) and the three logical relations (AND, OR, NOT).

The manual construction allows meaningful semantic concepts to be associated with the place and transition nodes of the PN event model. The semantic nature of PN models makes learning these models from training data infeasible. 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 [170].

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 [37].

A 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 [173].

Petri Nets are an event modelling formalism that allows defining the event domain in terms of high-level semantics such as temporal logical and spatial relations among abstraction primitives/ sub-events, hierarchy, concurrency and partial ordering. This formalism also allows meaningful semantic analysis of "incomplete" events. The high-level semantic nature of the PN formalism requires event models to be fully

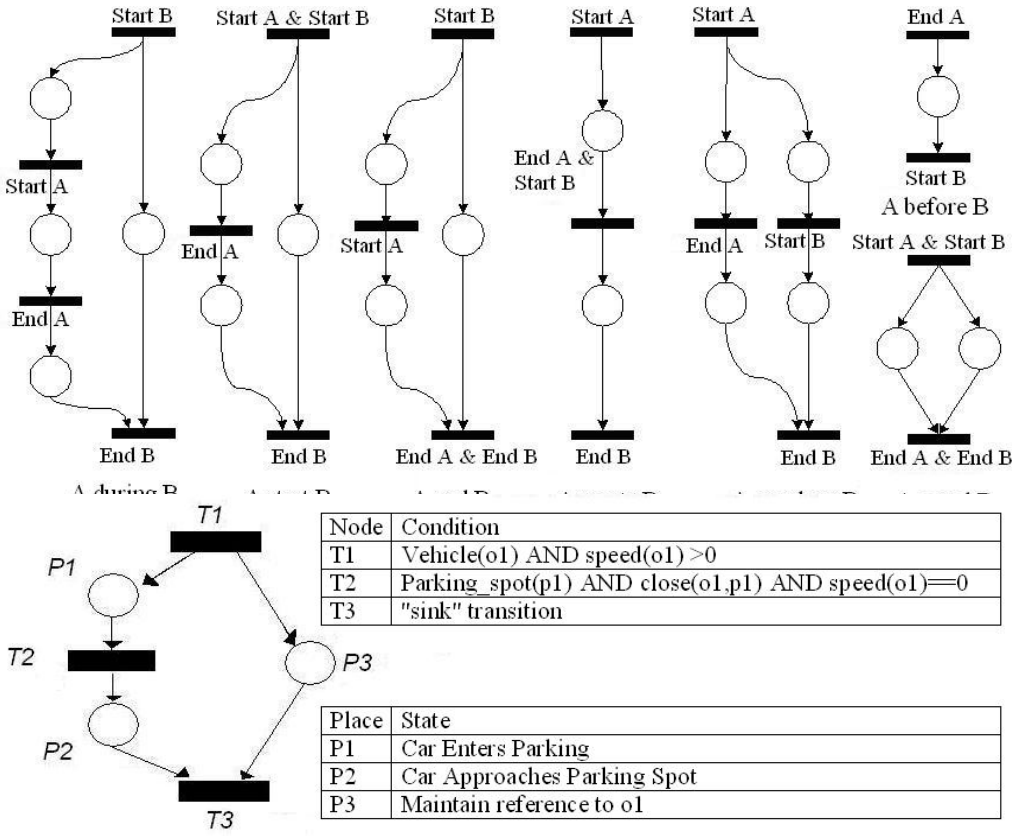


Figure 12: Example of a Plan PN model for the "Car is Parking" event.

specified using knowledge of the event domain.

7.3 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, undirected constraint graphs describing the temporal constraints of atomic sub-events [176, 177].

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 sub-event) in the temporal constraint network [178] and eliminating arcs in the network with less relevance to the solution [179, 143].

Vu et al [179, 143] achieve a speed up 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 [180], home care applications [181], and others [182].

In addition to temporal constraints, more recent work incorporates semantic knowledge about atemporal constraints pertaining to the properties of objects participating in the scene [183, 184, 185, 179, 143]. Description logics [186, 187] 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 [188].

An object-based abstraction is often coupled with the constraint satisfaction event models [183, 184, 185, 179, 143, 176]. Other works in constraint satisfaction event models assume a higher-level abstraction where a video sequence is described in terms of atomic sub-events [178, 176].

Constraint satisfaction event models represent video events as a set of semantic constraints which include spatial, temporal and logical relationships. An event is then recognized by determining whether a particular video sequence abstraction is consistent with these constraints.

7.4 Logic Approaches

The formalization of knowledge using logic is a well studied topic in artificial intelligence. The AI literature has proposed several works discussing how to specify the semantic knowledge of an event domain using well-studied logic. Many of these works discuss the specification of "event calculus" [189, 190].

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 applying the first-order logic framework of Prolog to recognition in the event domain of parking lot surveillance [191]. All semantic relations may be modeled as predicates and their relationships may be specified using the structure of the event inference rules. This specification may be considered an event domain independent part of the model specification. That is, one part of the event model describes predicates and inference rules that define semantic relations, and another part defines predicates that use these semantic relations to describe the events in a particular event domain.

To cope with the uncertainty inherent in video events some extensions to logics approaches have been proposed including multi-valued logics [192], and Markov logics [36].

Logic approaches are a promising direction in event understanding. These kind of event models provide a robust representation and a well-understood recognition technique. It is has not been studied how this class of event models will scale up to problems with many inference rules. Furthermore, semantic relations must be modeled as part of the knowledge base and are not an intrinsic part of the model.

8 Applications

In the introduction to this paper we suggested that the event understanding process can be roughly decomposed into two parts, which we have named abstraction and event modeling, respectively. To reiterate, abstraction is the organization of low-level video sequence data into intermediate units that capture salient and discriminative abstract properties of the video data. Event modeling is defined as the representation of occurrences of interest, using those units ("primitives") generated by the abstraction of

the video sequence, in such a way that allows recognition of these events as they occur in unlabeled video sequences.

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. That is, 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 event modeling focused paper may emphasize an HMMs 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 applied previously in this domain, as well as what methods used in other domains may be applied.

To this end, Table 3 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 in terms of the sub-discipline 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.

From 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 [23, 24, 27, 25]. Similarly, other works, emphasizing event modeling, choose straightforward abstraction schemes [37, 127, 19, 94]. An

object-based abstraction is observed to be popular among those papers that emphasize the event model. As previously stated, we define object-based abstraction as a class of abstraction approaches that utilize detection and tracking to abstract the video sequence as a set of objects and their properties. 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 which appear numerous times in the literature. Not surprisingly, these correspond to promising application of video event understanding.

Work	Abstraction Scheme	Event Model	Event Examples	Emphasis of paper
[25]	Gradient Histograms	Nearest Neighbor	"Walking", "Running", Waving	Abstraction
[27]	Pixel-Based Features	Naive Bayesian	Active, Inactive, Walking,Running, Fighting	Abstraction(Feature Selection)
[23]	Motion History Images	Nearest Neighbor	Aerobics Exercises	Abstraction
[24]	Pixel Energy History	Deviant Event Model	Events in an Office Setting	System
[34]	Pixel Change History	Gaussian Mixture Model	"Browsing", "Entering and Leaving" (Shopping Domain)	System Parameter Learning
[59]	Logical Abstraction	Rules Based in Force Dynamics	"Pick Up Object" "Put Down Object"	System
[43]	Principal components of Object Silhouettes	Nearest Neighbor	"Human Walking", "Dog Running"	Abstraction
[19]	Object-Based Abstraction	Bayesian Networks, Finite State Machines	"Approach", "Blocking", "Stealing" (Surveillance Domain)	Abstraction, Hierarchical Combination of Event Models
[69]	Pixel-Based Abstraction	Nearest Neighbor	"Walk", "Run", "Skip", "Hop", "March"	Abstraction, Distance Measure Comparison
[40]	Space-Time Volume Features	Nearest neighbor	"Jumping-Jack", "Walking", "Running"	Abstraction
[75]	Transformation of Object Features using SVM	Support Vector Machine	"Somebody is Crossing the Corridor Going From Room A to B"	System
[77]	Pixel-Based Abstraction	Support Vector Machines + Voting	"Walk", "Run", "Skip"	System
[83]	Pixel-Based Abstraction Constructed Using Boosting	Boosted Discriminative Classifier	"Talking On Phone", "Yawning with Hand at Mouth", "Putting on Eyeglasses"	Boosting Methodology
[139]	Silhouette Features	Conditional Random Fields	"Walking","Running", "Bending Down"	Event Model
[141]	Head Velocities	Conditional Random Fields	"Head Shakes", "Look Away" (Head Gestures)	Event Model
[193]	Fingertip Trajectories	Nearest Neighbor	"Up", "Down", "Grab", "Rotate", "Stop" (Hand Gestures)	System
[21]	Object-Based Abstraction	Finite State Machine	"A Car Passing Through the Checkpoint", "A Car Avoiding the Checkpoint" (UAV Domain)	Event Model
[194]	Body Part Kinematics	Hierarchical Finite State Machine	"Hand-Shaking", "Kicking", "Pointing" (Two-Person Interactions)	System
[94]	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
[92]	Face and Hand Locations	Finite State Machines	"Hand Wave", "Drawing a Circle", "Drawing a Figure Eight" (Gestures)	Automatic Model Learning
[98]	Object-Based Abstraction	Bayesian Network	"Overtaking", "Following" (UAV Video)	System
[101]	Object-Based Abstraction	Bayesian Network	Vehicle Blocking the Intersection, Turn Left at Intersection	Event Model, Building Event Models from Ontologies

Continued on Next Page...

Work	Abstraction Scheme	Event Model	Event Examples	Emphasis of paper
[102]	Object-Based Abstraction	Bayesian Networks	"Vehicle Parked", "Pedestrian Passing By Vehicle"	Event Model
[99]	Object-Based Abstraction	Bayesian Networks	American Football Plays	Event Model
[110]	Mesh Features	Hidden Markov Model	"Forehand Volley", "Backhand Stroke" (Tennis Strokes)	Event Model
[111]	(Hand) Object-Based Abstraction	Hidden Markov Model	American Sign Language Gestures	System
[114]	Pixel-based Abstraction	Entropic Hidden Markov Model	"Entering Room", "At Computer" (Office Domain) "North-South Traffic", "Pedestrians Stopping Traffic" (Traffic Domain)	Event Model
[115]	(Hand) Object-Based Abstraction	Parallel Hidden Markov Model	"Woman", "Try","Teach" (Sign Language Gestures)	System
[116]	(Hand) Object-Based Abstraction	Coupled Hidden Markov Models	"Single Whip", "Cobra" (Tai Chi Gestures)	Event Model
[118]	Object Trajectories	Coupled Hidden Markov Models	"Follow, Reach, and Walk Together" "Approach, Meet, and Go On Separately" (Person Interactions)	System
[119]	Pixel Change History	Gaussian Mixture Model (Sub-events) DML-HMM (Events)	Airport Cargo Loading/Unloading Events	Event Model
[120]	Silhouette /Motion Capture Features	Variable Length Markov Model	Exercise Domain	Event Model
[122]	Object-Based Abstraction	Bayesian Networks, Hidden Semi-Markov Model	"A Car Passing Through the Checkpoint", A Car Avoiding the Checkpoint (UAV Domain)	Event Model
[123]	Not described in detail	Parameterized HMM	"Pointing" and "Size" Gestures	Event Model
[125]	Landmarks	Hierarchical HMM	"Short Meal", "Have Snack" (Kitchen Domain)	Event Model
[127]	Object Trajectories	Switching Hidden Semi-Markov Model	"Eating Breakfast", "Washing Dishes" (Kitchen Domain)	Event Model
[129]	Hand Locations	Coupled Hidden Semi-Markov Models	Sign Language Gesture	Event Model
[135]	Object-Based Abstraction	Recurrent Bayesian Network (DBN)	"Violent Behavior" (Metro Station Domain)	Event Model
[131]	Object-Based Abstraction	Propagation Networks (DBN)	"Glucose Monitor Calibration" (Assistive Technology)	Event Model
[35]	Object-Based Abstraction	Petri Net	"Car Exchange"	Event Modeel
[37]	Object-Based Abstraction	Petri Net	"Visitor Entered the Hall", "Security Check Is Too Long" (Surveillance)	Event Model
[172]	Object-Based Abstraction	Petri Net	"Vehicle Departure", "Arsonist Action" (Parking Lot)	Event Model
[150]	Body Part features	Bayesian Networks, HMM, Grammar	"Shake Hands", "Hug", "Punch" (Two Person Interactions)	Event Model
[152]	Object Trajectories	HMM (Sub-Events),	(Hand Gestures),	Event Model

Continued on Next Page...

Work	Abstraction Scheme	Event Model	Event Examples	Emphasis of paper
		Stochastic Grammar(Events)	(Musical Conducting), (Parking Lot)	
[153]	Object-Based Abstraction	Stochastic Grammar	"Player Removed House Card", "Dealer Dealt Card to Player" (Card Game)	Event Model
[156]	Object Tracks	Hierarchy of Stochastic Grammar	"U-Turns", "S-Turns"	Event Model
[160]	Object-Based Abstraction	Attribute Grammar	"PARKING", "Dropoff" (Parking Lot)	Event Model
[165]	Object-Based Abstraction	Stochastic Grammar	(Convenience Store)	Learning Typical Events from Data

Table 3: A representative sample of the work being done in the domain of video event understanding. Generally, each work employs an abstraction scheme as well as an event model, however, only one of these is the main emphasis of the work. 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.

9 Conclusion and Future Work

In this paper we have presented our view of the domain of video event understanding. In section 2 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 those 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 human knowledgeable 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, efficient recognition algorithms. Such event models as Markov logics [36] and probabilistic Petri-Nets [173] 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 [195, 165, 136, 196, 188] 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] "Caretaker project, http://www.ist-caretaker.org/."

[2] "Etiseo project, http://www-sop.inria.fr/orion/etiseo/."

[3] "Avitrack project, http://www.avitrack.net/."

[4] "Advisor project, http://www-sop.inria.fr/orion/advisor/index.html."

[5] "Beware project, http://www.dcs.qmul.ac.uk/ sgg/beware/."

[6] "Icons project, http://www.dcs.qmul.ac.uk/research/vision/projects/icons/i

[7] "Vsam project, http://www.cs.cmu.edu/ vsam/."

[8] J. K. Aggarwal and Q. Cai, "Human motion analysis: A review," *Computer Vision and Image Understanding: CVIU*, vol. 73, no. 3, pp. 428–440, 1999. [Online]. Available: citeseer.ist.psu.edu/aggarwal99human.html

[9] H. Buxton, "Generative Models for Learning and Understanding Dynamic Scene Activity," in *ECCV Workshop on Generative Model Based Vision*, Copenhagen, Denmark, Jun. 2002.

[10] P. Turaga, R. Chellappa, V. S. Subrahmanian, and O. Udrea, "Machine recognition of human activities: A survey," *IEEE Transactions on Circuits, Systems and Video Technology (special issue on Event Analysis)*, (Accepted).

[11] W. Hu, T. Tan, L. Wang, and S. Maybank, "A survey on visual surveillance of object motion and behaviors," *Systems, Man and Cybernetics, Part C, IEEE Transactions on*, vol. 34, no. 3, pp. 334–352, 2004. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1304418

[12] J. J. Gibson, *The Ecological Approach to Visual Perception*. Lawrence Erlbaum Associates. [Online]. Available: http://www.amazon.ca/exec/obidos/redirect?tag=citeulike04-20&path=ASIN/0898599598

[13] H. Hecht, "The failings of three event perception theories," *Journal for the Theory of Social Behaviour*, vol. 30, pp. 1–25(25), March 2000. [Online]. Available: http://www.ingentaconnect.com/content/bpl/jtsb/2000/00000030/00000001/art00001

[14] I. Rock, *Indirect Perception*. The MIT Press, 1997.

[15] H.-H. Nagel, "From image sequences towards conceptual descriptions," *Image Vision Comput.*, vol. 6, no. 2, pp. 59–74, 1988.

[16] A. Bobick, "Movement, activity, and action: The role of knowledge in the perception of motion," *Royal Society Workshop on Knowledge-based Vision in Man and Machine*, vol. B-352, pp. 1257–1265, 1997.

[17] F. Bremond, "Scene understanding: perception, multi-sensor fusion, spatio-temporal reasoning and activity recognition," Ph.D. dissertation, HDR Universit de Nice-Sophia Antipolis, July 2007.

[18] 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 *Spatial Cognition*, 2003, pp. 232–248.

[19] S. Hongeng and R. Nevatia, "Multi-agent event recognition." in *International Conference on Computer Vision*, 2001, pp. 84–93.

[20] R. Howarth and B. H., "Conceptual descriptions from monitoring and watching image sequences," *Image and Vision Computing*, vol. 18, pp. 105–135(31), January 2000. [Online]. Available: http://www.ingentaconnect.com/content/els/02628856/2000

[21] G. G. Medioni, I. Cohen, F. Bremond, S. Hongeng, and R. Nevatia, "Event detection and analysis from video streams," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 8, pp. 873–889, 2001. [Online]. Available: citeseer.ist.psu.edu/medioni98event.html

[22] T. Xiang and S. Gong, "Beyond tracking: Modelling activity and understanding behaviour," *Int. J. Comput. Vision*, vol. 67, no. 1, pp. 21–51, 2006.

[23] A. Bobick and J. Davis, "The recognition of human movement using temporal templates," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 3, pp. 257–267, March 2001.

[24] S. Gong and J. Ng, "Learning pixel-wise signal energy for understanding semantics," in *British Machine Vision Conference*, 2001.

[25] L. Zelnik-Manor and M. Irani, "Statistical analysis of dynamic actions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 9, pp. 1530–1535, 2006.

[26] A. F. Bobick and A. D. Wilson, "A state-based approach to the representation and recognition of gesture," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 12, pp. 1325–1337, 1997.

[27] J. S.-V. P. Ribeiro, "Human activities recognition from video: modeling, feature selection and classification architecture." in *Workshop on Human Activity Recognition and Modelling*, Oxford, September 2005, pp. 61–70.

[28] E. Shechtman and M. Irani, "Matching local self-similarities across images and videos," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2007.

[29] H. Zhong, J. Shi, and M. Visontai, "Detecting unusual activity in video." in *"IEEE Computer Society Conference on Computer Vision and Pattern Recognition"*, 2004, pp. 819–826.

[30] T. Kim, S. Wong, and R. Cipolla, "Tensor canonical correlation analysis for action classification," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.

[31] P. Dollár, V. Rabaud, G. Cottrell, and S. Belongie, "Behavior recognition via sparse spatio-temporal features," in *IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*, October 2005.

[32] I. Laptev and P. Pérez, "Retrieving actions in movies," in *International Conference on Computer Vision*, Rio de Janeiro, Brazil, October 2007.

[33] J. Niebles, H. Wang, and L. Fei Fei, "Unsupervised learning of human action categories using spatial-temporal words," in *British Machine Vision Conference*, 2006, p. III:1249.

[34] S. Gong and T. Xiang, "Scene event recognition without tracking," *Acta Automatica Sinica.*, vol. 29, no. 3, pp. 321–331, May 2003.

[35] N. Ghanem, D. DeMenthon, D. Doermann, and L. Davis, "Representation and recognition of events in surveillance video using Petri Nets," in *CVPRW '04: Proceedings of the 2004 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'04) Volume 7*, 2004, p. 112.

[36] S. D. Tran and L. S. Davis, "Event modeling and recognition using Markov logic networks," in *European Conference on Computer Vision*, 2008, pp. 610–623.

[37] A. Borzin, E. Rivlin, and M. Rudzsky, "Surveillance interpretation using Generalized Stochastic Petri Nets," in *The International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS)*, 2007.

[38] N. Oliver, B. Rosario, and A. Pentland, "A Bayesian computer vision system for modeling human interactions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 831–843, 2000.

[39] D. A. Forsyth and J. Ponce, *Computer Vision: A Modern Approach.* Prentice Hall, August 2002. [Online]. Available: http://www.amazon.ca/exec/obidos/redirect?tag=citeulike0 20&path=ASIN/0130851981

[40] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri, "Actions as space-time shapes," in *ICCV '05: Proceedings of the Tenth IEEE International Conference on Computer Vision.* Washington, DC, USA: IEEE Computer Society, 2005, pp. 1395–1402.

[41] C. Schuldt, I. Laptev, and B. Caputo, "Recognizing human actions: a local svm approach," in *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, vol. 3, 2004, pp. 32–36 Vol.3. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=13344

[42] L. Wang and D. Suter, "Recognizing human activities from silhouettes: Motion subspace and factorial discriminative graphical model," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.

[43] R. Goldenberg, R. Kimmel, E. Rivlin, and M. Rudzsky, "Behavior classification by eigendecomposition of periodic motions," *Pattern Recognition*, vol. 38, no. 8, pp. 1033–1043, 2005.

[44] M. Singh, A. Basu, and M. Mandal, "Human activity recognition based on silhouette directionality," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 9, pp. 1280–1292, September 2008.

[45] K. Eickhorst, P. Agouris, and A. Stefanidis, "Modeling and comparing spatiotemporal events," in *dg.o '04: Proceedings of the 2004 annual national conference on Digital government research.* Digital Government Research Center, 2004, pp. 1–10.

[46] N. Ikizler and D. Forsyth, "Searching video for complex activities with finite state models," in

IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2007, pp. 1–8.

[47] N. Cuntoor and R. Chellappa, "Epitomic representation of human activities," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.

[48] L. Patino, H. Benhadda, E. Corvee, F. Bremond, and M. Thonnat, "Extraction of activity patterns on large video recordings," *Computer Vision, IET*, vol. 2, no. 2, pp. 108 – 128, June 2008.

[49] C. Piciarelli and G. Foresti, "On-line trajectory clustering for anomalous events detection," *Pattern Recognition Letters*, vol. 27, no. 15, pp. 1835–1842, November 2006.

[50] C. Piciarelli, G. Foresti, and L. Snidaro, "Trajectory clustering and its applications for video surveillance," in *IEEE International Conference on Advanced Video and Signal based Surveillance*, 2005, pp. 40–45.

[51] C. Piciarelli and G. Foresti, "Anomalous trajectory detection using support vector machines," in *IEEE International Conference on Advanced Video and Signal based Surveillance*, 2007, pp. 153–158.

[52] C. Piciarelli, C. Micheloni, and G. Foresti, "Kernel-based unsupervised trajectory clusters discovery," in *International Workshop on Visual Surveillance*, 2008.

[53] S. Gaffney and P. Smyth, "Trajectory clustering with mixtures of regression models," in *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining.* ACM Press, 1999, pp. 63–72.

[54] F. M. Porikli, "Learning object trajectory patterns by spectral clustering." in *International Conference on Multimedia and Expo.* IEEE, 2004, pp. 1171–1174. [Online]. Available: http://dblp.uni-trier.de/db/conf/icmcs/icme2004.html#Porikli04

[55] N. Anjum and A. Cavallaro, "Single camera calibration for trajectory-based behavior analysis," in *Advanced Video and Signal Based Surveillance*, 2007, pp. 147–152.

[56] G. Antonini and J. Thiran, "Counting pedestrians in video sequences using trajectory clustering," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 8, pp. 1008–1020, August 2006.

[57] S. Khalid and A. Naftel, "Classifying spatiotemporal object trajectories using unsupervised learning of basis function coefficients," in *VSSN '05: Proceedings of the third ACM international workshop on Video surveillance & sensor networks.* New York, NY, USA: ACM, 2005, pp. 45–52.

[58] F. Bashir, A. Khokhar, and D. Schonfeld, "Object Trajectory-Based activity classification and recognition using hidden Markov models," *IEEE Transactions on Image Processing*, vol. 16, no. 7, pp. 1912–1919, July 2007.

[59] J. M. Siskind, "Visual event classification via force dynamics," in *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence.* AAAI Press / The MIT Press, 2000, pp. 149–155.

[60] 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*, ser. Lecture Notes in Computer Science, C. Freksa, C. Habel, and K. Wender, Eds. Springer, 2003, pp. 232–248.

[61] J. Liu, S. Ali, and M. Shah, "Recognizing human actions using multiple features," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition.* IEEE Computer Society, 2008.

[62] P. Natarajan and R. Nevatia, "View and scale invariant action recognition using multiview shape-flow models," in *IEEE Computer Society Conference on Computer Vision and Pat-*

tern Recognition. IEEE Computer Society, 2008.

[63] M. D. Rodriguez, J. Ahmed, and M. Shah, "Action mach a spatio-temporal maximum average correlation height filter for action recognition," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE Computer Society, 2008.

[64] R. Souvenir and J. Babbs, "Learning the viewpoint manifold for action recognition," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE Computer Society, 2008.

[65] C. Thurau and V. Hlavac, "Pose primitive based human action recognition in videos or still images," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE Computer Society, 2008.

[66] A. Y. Ng and M. I. Jordan, "On discriminative vs. generative classifiers: A comparison of logistic regression and naive Bayes." in *Neural Information Processing Systems*, 2001, pp. 841–848.

[67] I. Ulusoy and C. M. Bishop, "Generative versus discriminative methods for object recognition," in *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 2*. Washington, DC, USA: IEEE Computer Society, 2005, pp. 258–265.

[68] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, August 2006. [Online]. Available: http://www.amazon.ca/exec/obidos/redirect?tag=citeulike09-20&path=ASIN/0387310738

[69] O. Masoud and N. Papanikolopoulos, "A method for human action recognition," *Image and Vision Computing*, vol. 21, no. 8, pp. 729–743, August 2003.

[70] H. Jiang, M. S. Drew, and Z. N. Li, "Successive convex matching for action detection," in *CVPR '06: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Washington, DC, USA: IEEE Computer Society, 2006, pp. 1646–1653.

[71] Y. Ke, R. Sukthankar, and M. Hebert, "Event detection in crowded videos," in *IEEE International Conference on Computer Vision*, October 2007.

[72] E. Shechtman and M. Irani, "Space-time behavior based correlation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, June 2005, pp. 405–412.

[73] N. Cristianini and J. Shawe-Taylor, *An introduction to support Vector Machines: and other kernel-based learning methods*. New York, NY, USA: Cambridge University Press, 2000. [Online]. Available: http://portal.acm.org/citation.cfm?id=345662

[74] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Min. Knowl. Discov.*, vol. 2, no. 2, pp. 121–167, 1998.

[75] M. Pittore, C. Basso, and A. Verri, "Representing and recognizing visual dynamic events with support vector machines," in *International Conference on Image Analysis and Processing*, 1999, pp. 18–23.

[76] M. Fleischman, P. Decamp, and D. Roy, "Mining temporal patterns of movement for video content classification," in *MIR '06: Proceedings of the 8th ACM international workshop on Multimedia information retrieval*. New York, NY, USA: ACM Press, 2006, pp. 183–192.

[77] D. B. D. Cao, O. Masoud and N. Papanikolopoulos, "Online motion classification using support vector machines." in *Proceedings of IEEE Int. Conf. on Robotics and Automation*, New Orleans,USA, 2004.

[78] D. Xu and S. F. Chang, "Visual event recognition in news video using kernel methods with multi-level temporal alignment," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Minneapolis, USA, June 2007.

[79] Y. Freund and R. Schapire, "A short introduction to boosting," *J. Japan. Soc. for Artif. Intel.*, vol. 14, no. 5, pp. 771–780, 1999. [Online]. Available: citeseer.ist.psu.edu/freund99short.html

[80] D. Minnen, T. Westeyn, and T. Starner, "Recognizing soldier activities inthe field," in *4th International Workshop on Wearable and Implantable Body Sensor Networks*, Aachen, Germany, 2007.

[81] S. Nowozin, G. Bakir, and K. Tsuda, "Discriminative subsequence mining for action classification," in *Eleventh IEEE International Conference on Computer Vision*, 10 2007, pp. 1919–1923. [Online]. Available: http://www.kyb.mpg.de/bs/people/nowozin/pboost/

[82] F. Lv, J. Kang, R. Nevatia, I. Cohen, and G. Medioni, "Automatic tracking and labeling of human activities in a video sequence," in *Sixth IEEE International Workshop on Performance Evaluation of Tracking and Surveillance PETS-ECCV*, Prague, Czech Republic, 2004.

[83] P. Smith, N. da Vitoria Lobo, and M. Shah, "Temporalboost for event recognition," in *ICCV '05: Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*. Washington, DC, USA: IEEE Computer Society, 2005, pp. 733–740.

[84] P. Ribeiro, P. Moreno, and J. Santos Victor, "Boosting with temporal consistent learners: An application to human activity recognition," in *International Symposium on Visual Computing*, 2007, pp. I: 464–475.

[85] P. Canotilho and R. P. Moreno, "Detecting luggage related behaviors using a new temporal boost algorithm," in *IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*, 2007.

[86] H. Vassilakis, A. J. Howell, and H. Buxton, "Comparison of feedforward (tdrbf) and generative (tdrgbn) network for gesture based control," in *GW '01: Revised Papers from the International Gesture Workshop on Gesture and Sign Languages in Human-Computer Interaction*. London, UK: Springer-Verlag, 2002, pp. 317–321.

[87] A. Gill, *Introduction to the Theory of Finite State Machines*. 330 W 42 St New York, N. Y.: McGraw-Hill Book Co. Inc.., 1962.

[88] K. H. Jo, Y. Kuno, and Y. Shirai, "Manipulative hand gesture recognition using task knowledge for human computer interaction," in *FG '98: Proceedings of the 3rd. International Conference on Face & Gesture Recognition*. Washington, DC, USA: IEEE Computer Society, 1998, p. 468.

[89] F. Lv and R. Nevatia, "Single view human action recognition using key pose matching and viterbi path searching," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.

[90] F. Bremond and M. Thonnat, "Analysis of human activities described by image sequences," in *Proc. of the International Florida Artificial Intelligence Research Symposium (FLAIRS'97)*, May 1997.

[91] F. Bremond, M. Thonnat, and M. Zuniga, "Video understanding framework for automatic behavior recognition," *Behavior Research Methods*, vol. 3, no. 38, pp. 416–426, 2006.

[92] P. Hong, T. S. Huang, and M. Turk, "Gesture modeling and recognition using finite state machines," in *FG '00: Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition 2000*. Washington, DC, USA: IEEE Computer Society, 2000, p. 410.

[93] 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 *ACM International Conference on Image and Video Retrieval*, 2003, pp. 394–403.

[94] D. Mahajan, N. Kwatra, S. Jain, P. Kalra, and S. Banerjee, "A framework for activity recognition and detection of unusual activities," in *Indian Conference on Computer Vision, Graphics and Image Processing*, 2004.

[95] F. V. Jensen, *Bayesian Networks and Decision Graphs*, ser. Information Science and Statistics. Springer, July 2001. [Online]. Available: http://www.amazon.ca/exec/obidos/redirect?tag=citeulike09-20&path=ASIN/0387952594

[96] J. Pearl, *Probabilistic reasoning in intelligent systems: networks of plausible inference*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1988.

[97] Jordan, *Learning in Graphical Models (Adaptive Computation and Machine Learning)*. The MIT Press, November 1998. [Online]. Available: http://www.amazon.ca/exec/obidos/redirect?tag=citeulike09-20&path=ASIN/0262600323

[98] H. Buxton and S. Gong, "Visual surveillance in a dynamic and uncertain world," *Artificial Intelligence*, vol. 78, no. 1-2, pp. 431–459, 1995. [Online]. Available: citeseer.csail.mit.edu/buxton95visual.html

[99] S. S. Intille and A. F. Bobick, "A framework for recognizing multi-agent action from visual evidence," in *Proceedings of the sixteenth national conference on Artificial intelligence and the eleventh Innovative applications of artificial intelligence conference innovative applications of artificial intelligence*, 1999, pp. 518–525. [Online]. Available: citeseer.ist.psu.edu/intille99framework.html

[100] F. Lv, X. Song, B. Wu, V. K. Singh, and R. Nevatia, "Left-luggage detection using Bayesian inference." in *Ninth IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*, 2006.

[101] R. P. Higgins, "Automatic event recognition for enhanced situational awareness in uav video." in *Military Communications Conference*, 2005.

[102] P. Remagnino, T. Tan, and K. Baker, "Multi-agent visual surveillance of dynamic scenes," *Image and Vision Computing*, vol. 16, no. 8, pp. 529–532, June 1998.

[103] S. Hongeng, R. Nevatia, and F. Bremond, "Video-based event recognition: activity representation and probabilistic recognition methods," *Comput. Vis. Image Underst.*, vol. 96, no. 2, pp. 129–162, 2004.

[104] S. Wong, T. Kim, and R. Cipolla, "Learning motion categories using both semantic and structural information," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–6.

[105] X. Wang, X. Ma, and W. Grimson, "Unsupervised activity perception by hierarchical Bayesian models," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.

[106] T. Hofmann, "Probabilistic latent semantic indexing," in *SIGIR '99: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA: ACM, 1999, pp. 50–57.

[107] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Readings in Speech Recognition*, pp. 267–296, 1990.

[108] Z. Ghahramani and M. I. Jordan, "Factorial hidden Markov models," in *Proc. Conf. Advances in Neural Information Processing Systems, NIPS*, D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo, Eds., vol. 8. MIT Press, 1995, pp. 472–478. [Online]. Available: citeseer.ist.psu.edu/article/ghahramani97factorial.html

[109] S. Gong and H. Buxton, "On the visual expectations of moving objects," in *ECAI '92: Proceedings of the 10th European conference on Artificial intelligence*. New York, NY, USA: John Wiley & Sons, Inc., 1992, pp. 781–784.

[110] J. Yamato, J. Ohya, and K. Ishii, "Recognizing human action in time-sequential images using hidden Markov models," in *International Conference on Computer Vision*, 1992, pp. 379–385.

[111] T. Starner and A. Pentland, "Real-time american sign language recognition from video using hidden Markov models," in *IEEE Symposium on Computer Vision*, 1995, p. 5B. [Online]. Available: citeseer.ist.psu.edu/starner96realtime.html

[112] J. Schlenzig, E. Hunter, and R. Jain, "Recursive identification of gesture inputs using hidden Markov models," in *Second IEEE Workshop on Applications of Computer Vision*, 1994, pp. 187–194.

[113] A. Ogale, A. Karapurkar, G. Guerra-Filho, and Y. Aloimonos, "View-invariant identification of pose sequences for action recognition." in *Video Analysis and Content Extraction Workshop (VACE)*, Tampa, FL, USA, 2004.

[114] M. Brand and V. Kettnaker, "Discovery and segmentation of activities in video," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 844–851, 2000.

[115] C. Vogler and D. Metaxas, "A framework for recognizing the simultaneous aspects of american sign language," *Comput. Vis. Image Underst.*, vol. 81, no. 3, pp. 358–384, 2001.

[116] M. Brand, N. Oliver, and A. Pentland, "Coupled hidden Markov models for complex action recognition," in *"IEEE Computer Society Conference on Computer Vision and Pattern Recognition"*. Los Alamitos, CA, USA: IEEE Computer Society, 1997, p. 994.

[117] M. Brand, "The "inverse hollywood problem": From video to scripts and storyboards via causal analysis." in *Proceedings of the Fourteenth National Conference on Artificial Intelligence and ninth Conference on Innovative Applications of Artificial Intelligence*, 1997, pp. 132–137.

[118] N. M. Oliver, B. Rosario, and A. Pentland, "A Bayesian computer vision system for modeling human interactions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 831–843, 2000. [Online]. Available: citeseer.ist.psu.edu/article/oliver99bayesian.html

[119] S. Gong and T. Xiang, "Recognition of group activities using dynamic probabilistic networks," in *ICCV '03: Proceedings of the Ninth IEEE International Conference on Computer Vision*. Washington, DC, USA: IEEE Computer Society, 2003, p. 742.

[120] A. Galata, N. Johnson, and D. Hogg, "Learning variable-length Markov models of behavior," *Comput. Vis. Image Underst.*, vol. 81, no. 3, pp. 398–413, 2001.

[121] A. Galata, A. Cohn, D. Magee, and D. Hogg, "Modeling interaction using learnt qualitative spatio-temporal relations and variable length Markov models," in *Conf on Artificial Intelligence*, Lyon, 2002. [Online]. Available: citeseer.ist.psu.edu/galata02modeling.html

[122] S. Hongeng and R. Nevatia, "Large-scale event detection using semi-hidden Markov models," in *ICCV '03: Proceedings of the Ninth IEEE International Conference on Computer Vision*. Washington, DC, USA: IEEE Computer Society, 2003, p. 1455.

[123] A. D. Wilson and A. F. Bobick, "Recognition and interpretation of parametric gesture," in *International Conference on Computer Vision*, 1998, pp. 329–336. [Online]. Available: citeseer.ist.psu.edu/wilson98recognition.html

[124] S. Fine, Y. Singer, and N. Tishby, "The hierarchical hidden Markov model: Analysis and applications," *Machine Learning*, vol. 32, no. 1, pp. 41–62, 1998. [Online]. Available: citeseer.ist.psu.edu/fine98hierarchical.html

[125] 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 Computer Society Conference on Computer Vision and Pattern Recognition*. Washington, DC, USA: IEEE Computer Society, 2005, pp. 955–960.

[126] N. Oliver, E. Horvitz, and A. Garg, "Layered representations for human activity recognition," in *Fourth IEEE International Conference on Multimodal Interfaces,*, 2002, pp. 3– 8.

[127] 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 *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern*

Recognition (CVPR'05) - Volume 1. Washington, DC, USA: IEEE Computer Society, 2005, pp. 838–845.

[128] P. Natarajan and R. Nevatia, "Hierarchical multi-channel hidden semi Markov models." in *Proceedings of the International Joint Conference on Artificial Intelligence*, M. M. Veloso, Ed., 2007, pp. 2562–2567. [Online]. Available: http://dblp.uni-trier.de/db/conf/ijcai/ijcai2007.html#NatarajanN07

[129] ——, "Coupled hidden semi Markov models for activity recognition," *Proceedings of the IEEE Workshop on Motion and Video Computing*, vol. 0, p. 10, 2007.

[130] N. Oliver and E. Horvitz, "A comparison of HMMs and Dynamic Bayesian networks for recognizing office activities." in *User Modeling*, 2005, pp. 199–209.

[131] Y. Shi, Y. Huang, D. Minnen, A. Bobick, and I. Essa, "Propagation networks for recognition of partially ordered sequential action," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 02, pp. 862–869, 2004.

[132] Y. Shi and A. F. Bobick, "P-net: A representation for partially-sequenced, multi-stream activity," *Proceedings of the 2003 Conference on Computer Vision and Pattern Recognition Workshop*, vol. 04, p. 40, 2003.

[133] J. Muncaster and Y. Ma, "Activity recognition using dynamic Bayesian networks with automatic state selection," *Proceedings of the IEEE Workshop on Motion and Video Computing*, vol. 0, p. 30, 2007.

[134] B. Laxton, J. Lim, and D. Kriegman, "Leveraging temporal, contextual and ordering constraints for recognizing complex activities in video," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.

[135] N. Moenne-Loccoz, F. Bremond, and M. Thonnat, "Recurrent Bayesian network for the recognition of human behaviors from video." in *International Conference on Computer Vision Systems*, 2003, pp. 68–77.

[136] Y. Shi, A. Bobick, and I. Essa, "Learning temporal sequence model from partially labeled data," in *CVPR '06: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Washington, DC, USA: IEEE Computer Society, 2006, pp. 1631–1638.

[137] J. Lafferty, A. McCallum, and F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proc. 18th International Conf. on Machine Learning*. Morgan Kaufmann, San Francisco, CA, 2001, pp. 282–289. [Online]. Available: citeseer.ist.psu.edu/lafferty01conditional.html

[138] 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. MIT Press, 2006.

[139] C. Sminchisescu, A. Kanaujia, Z. Li, and D. Metaxas, "Conditional models for contextual human motion recognition," in *IEEE International Conference on Computer Vision*, 2005, pp. 1808–1815.

[140] T. Wang, J. Li, Q. Diao, W. Hu, Y. Zhang, and C. Dulong, "Semantic event detection using conditional random fields," in *CVPRW '06: Proceedings of the 2006 Conference on Computer Vision and Pattern Recognition Workshop*. Washington, DC, USA: IEEE Computer Society, 2006, p. 109.

[141] L. Morency, A. Quattoni, and T. Darrell, "Latent-dynamic discriminative models for continuous gesture recognition," MIT, Tech. Rep. MIT-CSAIL-TR-2007-002, 2007.

[142] H. Ning, W. Xu, Y. Gong, and T. S. Huang, "Latent pose estimator for continuous action recognition," in *European Conference on Computer Vision*, 2008, pp. 419–433.

[143] V. T. Vu, F. Brémond, and M. Thonnat, "Automatic video interpretation: A recognition algorithm for temporal scenarios based on pre-compiled scenario models," in *International Conference on Computer Vision Systems*, 2003, pp. 523–533.

[144] A. V. Aho and J. D. Ullman, *The Theory of Parsing, Translation, and Compiling*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1972.

[145] F. Jelinek, *Statistical Methods for Speech Recognition*. The MIT Press, January 1998. [Online]. Available: http://www.amazon.ca/exec/obidos/redirect?tag=citeulike09-20&path=ASIN/0262100665

[146] G. Chanda and F. Dellaert, "Grammatical methods in computer vision: An overview," Georgia Institute of Technology, Tech. Rep. GIT-GVU-04-29, 2004.

[147] N. Chomsky, *Syntactic structures*. Mouton, 1957.

[148] J. Earley, "An efficient context-free parsing algorithm," *Commun. ACM*, vol. 13, no. 2, pp. 94–102, 1970.

[149] M. Brand, "Understanding manipulation in video," in *FG '96: Proceedings of the 2nd International Conference on Automatic Face and Gesture Recognition (FG '96)*. Washington, DC, USA: IEEE Computer Society, 1996, p. 94.

[150] M. S. Ryoo and J. K. Aggarwal, "Recognition of composite human activities through context-free grammar based representation," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Washington, DC, USA: IEEE Computer Society, 2006, pp. 1709–1718.

[151] A. Stolcke, "An efficient probabilistic context-free parsing algorithm that computes prefix probabilities," in *Computational Linguistics, MIT Press for the Association for Computational Linguistics*, 1995, vol. 21. [Online]. Available: citeseer.ist.psu.edu/article/stolcke95efficient.html

[152] 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, 2000.

[153] D. Moore and I. Essa, "Recognizing multitasked activities using stochastic context-free grammar," in *CVPR Workshop on Models vs Exemplars in Computer Vision*, 2001. [Online]. Available: citeseer.ist.psu.edu/moore01recognizing.html

[154] M. Yamamoto, H. Mitomi, F. Fujiwara, and T. Sato, "Bayesian classification of task-oriented actions based on stochastic context-free grammar," in *FGR '06: Proceedings of the 7th International Conference on Automatic Face and Gesture Recognition (FGR06)*. Washington, DC, USA: IEEE Computer Society, 2006, pp. 317–323.

[155] D. Minnen, I. Essa, and T. Starner, "Expectation grammars: leveraging high-level expectations for activity recognition," in *Proc. Computer Vision and Pattern Recognition*, vol. 2, June 2003, pp. 626–632. [Online]. Available: citeseer.ist.psu.edu/616536.html

[156] D. Lymberopoulos, A. S. Ogale, A. Savvides, and Y. Aloimonos, "A sensory grammar for inferring behaviors in sensor networks," in *IPSN '06: Proceedings of the fifth international conference on Information processing in sensor networks*. New York, NY, USA: ACM Press, 2006, pp. 251–259.

[157] Z. Zhang, K. Huang, and T. Tan, "Multi-thread parsing for recognizing complex events in videos," in *European Conference on Computer Vision*, 2008, pp. 738–751.

[158] A. Ogale, A. Karapurkar, and Y. Aloimonos, "View-invariant modeling and recognition of human actions using grammars," in *IEEE Workshop on Dynamical Vision*, 2005.

[159] D. E. Knuth, "Semantics of context-free languages." *Mathematical Systems Theory*, vol. 2, no. 2, pp. 127–145, 1968.

[160] S. W. Joo and R. Chellappa, "Attribute grammar-based event recognition and anomaly detection," in *CVPRW '06: Proceedings of the 2006 Conference on Computer Vision and Pattern Recognition Workshop*. Washington, DC, USA: IEEE Computer Society, 2006, p. 107.

[161] Z. Zhang, K. Huang, and T. Tan, "Complex activity representation and recognition by extended stochastic grammar." in *Asian Conference on Computer Vision*, 2006, pp. 150–159.

[162] K. Cho, H. Cho, and K. Um, "Human action recognition by inference of stochastic regular grammars." in *Structural, Syntactic, and Statistical Pattern Recognition*, 2004, pp. 388–396.

[163] ——, "Inferring stochastic regular grammar with nearness information for human action recognition," in *International Conference on Image Analysis and Recognition*, 2006, pp. II: 193–204.

[164] G. Guerra-Filho and Y. Aloimonos, "Learning parallel grammar systems for a human activity language," UM Computer Science Department, Tech. Rep. CS-TR-4837, 2006.

[165] K. Kitani, Y. Sato, and A. Sugimoto, "Recovering the basic structure of human activities from a video-based symbol string," in *Proceedings of the IEEE Workshop on Motion and Video Computing*, 2007, pp. 9–9.

[166] C. A. Petri, "Kommunikation mut automaten," Ph.D. dissertation, Schriften des IIM Nr. 2, Bonn, 1962.

[167] D. Kartson, G. Balbo, S. Donatelli, G. Franceschinis, and G. Conte, *Modelling with Generalized Stochastic Petri Nets*. New York, NY, USA: John Wiley & Sons, Inc., 1994.

[168] T. Murata, "Petri Nets: Properties, analysis and applications," *Proceedings of the IEEE*, vol. 77, no. 4, pp. 541–580, 1989. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=24143

[169] P. J. Haas, *Stochastic Petri Nets: Modelling, stability, simulation.* New York: Springer-Verlag, 2002.

[170] G. Lavee, A. Borzin, E. Rivlin, and M. Rudzsky, "Building Petri Nets from video event ontologies," in *International Symposium on Visual Computing*, 2007, pp. 442–451.

[171] N. M. Ghanem, "Petri Net models for event recognition in surveillance videos," Ph.D. dissertation, University of Maryland, 2007.

[172] C. Castel, L. Chaudron, and C. Tessier, "What is going on? a high level interpretation of sequences of images," in *European Conference on Computer Vision*, 1996. [Online]. Available: citeseer.ist.psu.edu/castel96what.html

[173] 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 Transactions on Multimedia*, Accepted.

[174] Y. Nam, N. Wohn, and H. Lee-Kwang, "Modeling and recognition of hand gesture using colored Petri Nets." *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, vol. 29, no. 5, pp. 514–521, 1999.

[175] N. Ghanem, D. Doermann, L. Davis, and D. DeMenthon, "Mining Tools for Surveillance Video," in *Proceedings in SPIE 16th International Symposium on Electronic Imaging*, January 2004, pp. 5307 259–270.

[176] C. Dousson, P. Gaborit, and M. Ghallab, "Situation recognition: Representation and algorithms," 1993, pp. 166–172.

[177] M. Ghallab, "On chronicles: Representation, on-line recognition and learning." in *5th International Conference on Principles of Knowledge Representation and Reasoning*, USA, November 1996.

[178] C. S. Pinhanez and A. F. Bobick, "Human action detection using pnf propagation of temporal constraints," in *CVPR '98: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Washington, DC, USA: IEEE Computer Society, 1998, p. 898.

[179] V. Vu, F. Bremond, and M. Thonnat, "Temporal constraints for video interpretation," in *15th European Conference on Artificial Intelligence*, Lyon, FRANCE.

[180] F. Fusier, V. Valentin, F. Bremond, M. Thonnat, M. Borg, D. Thirde, and J. Ferryman, "Video understanding for complex activity recognition," *Mach. Vision Appl.*, vol. 18, no. 3, pp. 167–188, 2007.

[181] N. Zouba, F. Bremond, and M. Thonnat, "Monitoring activities of daily living (adls) of elderly based on 3d key human postures," in *4th International Cognitive Vision Workshop, ICVW 2008*, Santorini, Greece, May 2008.

[182] V. T. VU, "Temporal scenarios for automatic video interpretation," Ph.D. dissertation, Universite de Nice Sophia Antipolis, France, 2004.

[183] N. Chelq and M. Thonnat, "Realtime image sequence interpertation for video-surveillance applications," in *International Conference on Image Processing (ICIP '96)*, Lausanne, Switzerland, 1996.

[184] N. Rota and M. Thonnat, "Activity recognition from video ssequences using declarative models," in *14th European Conference on Artificial Intelligence (ECAI 200)*, 2000.

[185] ——, "Video sequence interpretation for visual surveillance," in *VS '00: Proceedings of the Third IEEE International Workshop on Visual Surveillance (VS'2000)*. Washington, DC, USA: IEEE Computer Society, 2000, p. 59.

[186] K. Terzic, L. Hotz, and B. Neumann, "Division of work during behaviour recognition - the scenic approach," in *Workshop on Behaviour Monitoring and Interpretation*, 2007, pp. 144–159.

[187] B. Neumann and R. Möller, "On scene interpretation with description logics," *Image Vision Comput.*, vol. 26, no. 1, pp. 82–101, 2008.

[188] J. Hartz and B. Neumann, "Learning a knowledge base of ontological concepts for high-level scene interpretation," in *ICMLA '07: Proceedings of the Sixth International Conference on Machine Learning and Applications*. Washington, DC, USA: IEEE Computer Society, 2007, pp. 436–443.

[189] M. Shanahan, "An abductive event calculus planner," *J. Log. Program.*, vol. 44, no. 1-3, pp. 207–240, 2000.

[190] ——, "Representing continuous change in the event calculus," in *Proceedings of the European conference on Artificial intelligence*, 1990, pp. 598–603.

[191] V. D. Shet, D. Harwood, and L. S. Davis, "Vidmap: video monitoring of activity with prolog," in *Proceedings. IEEE Conference on Advanced Video and Signal Based Surveillance, 2005.*, 2005.

[192] ——, "Multivalued default logic for identity maintenance in visual surveillance," in *European Conference on Computer Vision*, 2006.

[193] J. Davis and M. Shah, "Visual gesture recognition," in *Visual Image and Signal Processing*, vol. 141, April 1994, pp. 101–110.

[194] S. Park and J. Aggarwal, "A hierarchical Bayesian network for event recognition of human actions and interactions," *Multimedia Systems*, vol. 10, no. 2, pp. 164–179, August 2004. [Online]. Available: http://dx.doi.org/10.1007/s00530-004-0148-1

[195] C. Town, "Ontology-driven Bayesian networks for dynamic scene understanding," in *CVPRW '04: Proceedings of the 2004 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'04) Volume 7*. Washington, DC, USA: IEEE Computer Society, 2004, p. 116.

[196] A. Toshev, F. Bremond, and M. Thonnat, "An apriori-based method for frequent composite event discovery in videos," in *ICVS '06: Proceedings of the Fourth IEEE International Conference on Computer Vision Systems*. Washington, DC, USA: IEEE Computer Society, 2006, p. 10.

[197] *2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2008), 24-26 June 2008, Anchorage, Alaska, USA*. IEEE Computer Society, 2008.