Seeing What You're Told: Sentence-Guided Activity Recognition In Video

N. Siddharth* siddharth@iffsid.com
Andrei Barbu* andrei@0xab.com
Jeffrey Mark Siskind* qobi@purdue.edu

Abstract

We present a system that demonstrates how the compositional structure of events, in concert with the compositional structure of language, can interplay with the underlying focusing mechanisms in video action recognition, thereby providing a medium, not only for top-down and bottom-up integration, but also for multi-modal integration between vision and language. We show how the roles played by participants (nouns), their characteristics (adjectives), the actions performed (verbs), the manner of such actions (adverbs), and changing spatial relations between participants (prepositions) in the form of whole sentential descriptions mediated by a grammar, guides the activity-recognition process. Further, the utility and expressiveness of our framework is demonstrated by performing three separate tasks in the domain of multi-activity videos: sentence-guided focus of attention, generation of sentential descriptions of video, and query-based video search, simply by leveraging the framework in different manners.

1. Introduction

The ability to describe the observed world in natural language is a quintessential component of human intelligence. A particular feature of this ability is the use of rich sentences, involving the composition of multiple nouns, adjectives, verbs, adverbs, and prepositions, to describe not just static objects and scenes, but also events that unfold over time. Furthermore, this ability appears to be learned by virtually all children. The deep semantic information learned is multi-purpose: it supports comprehension, generation, and inference. In this work, we investigate the intuition, and the precise means and mechanisms that will enable us to support such ability in the domain of activity recognition in multi-activity videos.

Suppose we wanted to recognize an occurrence of an event described by the sentence *The ball bounced*, in a video. Nominally, we would need to detect the *ball* and its position in the field of view in each frame and determine that the sequence of such detections satisfied the requirements of *bounce*. The sequence of such object detections and their corresponding positions over time constitutes a *track* for that object. In this view, the semantics of an intransitive verb like *bounce* would be formulated as a unary predicate over object tracks. Recognizing occurrences of events described by sentences containing transitive verbs, like *The person approached the ball*, would require detecting and tracking two objects, the *person* and the *ball* constrained by a binary predicate.

In an ideal world, event recognition would proceed in a purely feed-forward fashion: robust and unambiguous object detection and tracking followed by application of the semantic predicates on the recovered tracks. However, the current state-of-the-art in computer vision is far from this ideal. Object detection alone is highly unreliable. The best current average-precision scores on PASCAL VOC hover around 40%-50% [3]. As a result, object detectors suffer from both false positives and false negatives. One way around this is to use detection-based tracking [17], where one biases the detector to overgenerate, alleviating the problem of false negatives, and uses a different mechanism to select among the overgenerated detections to alleviate the problem of false positives. One such mechanism selects detections that are temporally coherent, *i.e.* the track motion being consistent with optical flow. Barbu *et al.* [2] proposed an alternate mechanism that selected detections for a track that satisfied a unary predicate such as one would construct for an intransitive verb like *bounce*. We significantly extend that approach, selecting detections for multiple tracks that collectively satisfy a complex multi-argument predicate representing the semantics of an entire sentence. That predicate is constructed as a conjunction of predicates representing the semantics of individual words in that sentence. For example, given the sentence *The person to the left of the chair approached the trash can*, we construct a logical form.

$$\textsc{person}(P) \wedge \textsc{toTheLeftOf}(P, Q) \wedge \textsc{chair}(Q) \wedge \textsc{approach}(P, R) \wedge \textsc{trashCan}(R)$$

Our tracker is able to simultaneously construct three tracks P, Q, and R, selecting out detections for each, in an optimal fashion that simultaneously optimizes a joint measure of detection score and temporal coherence while also satisfying the above conjunction of predicates. We obtain the aforementioned detections by employing a state-of-the-

*School of Electrical and Computer Engineering, Purdue University, West Lafayette IN 47907-2035

art object detector [5], where we train a model for each object (*e.g. person*, *chair*, *etc.*), which when applied to an image, produces axis-aligned bounding boxes with associated scores indicating strength of detection.

We represent the semantics of lexical items like *person*, *to the left of*, *chair*, *approach*, and *trash can* with predicates over tracks like PERSON(P), TOTHELEFTOF(P, Q), CHAIR(Q), APPROACH(P, R), and TRASHCAN(R). These predicates are in turn represented as regular expressions (*i.e.* finite state recognizers or FSMs) over features extracted from the sequence of detection positions, shapes, and sizes as well as their temporal derivatives. For example, the predicate TOTHELEFTOF(P, Q) might be a single state FSM where, on a frame-by-frame basis, the centers of the detections for P are constrained to have a lower x-coordinate than the centers of the detections for Q. The actual formulation of the predicates (Table 2) is far more complex to deal with noise and variance in real-world video. What is central is that the semantics of *all* parts of speech, namely nouns, adjectives, verbs, adverbs, and prepositions (both those that describe spatial-relations and those that describe motion), is uniformly represented by the same mechanism: predicates over tracks formulated as finite state recognizers over features extracted from the detections in those tracks.

We refer to this capacity as the *Sentence Tracker*, which is a function $\mathcal{S} : (D, \Phi) \mapsto (\tau, Z)$, that takes as input an overgenerated set D of detections along with a complex sentential predicate Φ and produces a score τ together with a set Z of tracks that satisfy Φ while optimizing a linear combination of detection scores and temporal coherence. This can be used for three distinct purposes:

focus of attention One can apply the sentence tracker to the same video D, that depicts multiple simultaneous events taking place in the field of view with different participants, with two different sentences Φ_1 and Φ_2. In other words, one can compute $(\tau_1, Z_1) = \mathcal{S}(D, \Phi_1)$ and $(\tau_2, Z_2) = \mathcal{S}(D, \Phi_2)$ to yield two different sets of tracks Z_1 and Z_2 corresponding to the different sets of participants in the different events described by Φ_1 and Φ_2. We demonstrate this in section 4.1.

generation One can take a video D as input and systematically search the space of all possible Φ that correspond to sentences that can be generated by a context-free grammar and find that sentence that corresponds to the Φ^* for which $(\tau^*, Z^*) = \mathcal{S}(D, \Phi^*)$ yields the maximal τ^*. This can be used to generate a sentence that describes an input video D. We demonstrate this in section 4.2.

retrieval One can take a collection $\mathcal{D} = \{D_1, \ldots, D_n\}$ of videos (or a single long video chopped into short clips) along with a sentential query Φ, compute $(\tau_i, Z_i) = \mathcal{S}(D_i, \Phi)$ for each D_i, and find the clip D_i with maximal score τ_i. This can be used to perform sentence-based video search. We demonstrate this in section 4.3.

However, we first present the two central algorithmic contributions of this work. In section 2 we present the details of the sentence tracker, the mechanism for efficiently constraining several parallel detection-based trackers, one for each participant, with a conjunction of finite state recognizers. In section 3 we present lexical semantics for a small vocabulary of 17 lexical items (5 nouns, 2 adjectives, 4 verbs, 2 adverbs, 2 spatial-relation prepositions, and 2 motion prepositions) all formulated as finite state recognizers over features extracted from detections produced by an object detector, together with compositional semantics that maps a sentence to a semantic formula Φ constructed from these finite state recognizers where the object tracks are assigned to arguments of these recognizers.

2. The Sentence Tracker

Barbu *et al.* [2] address the issue of selecting detections for a track that simultaneously satisfies a temporal-coherence measure and a single predicate corresponding to an intransitive verb such as *bounce*. Doing so constitutes the integration of top-down high-level information, in the form of an event model, with bottom-up low-level information in the form of object detectors. We provide a short review of the relevant material in that work to introduce notation and provide the basis for our exposition of the sentence tracker.

$$\max_{j^1,\ldots,j^T} \sum_{t=1}^{T} f(b^t_{j^t}) + \sum_{t=2}^{T} g(b^{t-1}_{j^{t-1}}, b^t_{j^t}) \tag{1}$$

The first component is a detection-based tracker. For a given video with T frames, let j be the index of a detection and b^t_j be a particular detection in frame t with score $f(b^t_j)$. A sequence $\langle j^1, \ldots, j^T \rangle$ of detection indices, one for each frame t, denotes a track comprising detections $b^t_{j^t}$. We seek a track that maximizes a linear combination of aggregate detection score, summing $f(b^t_j)$ over all frames, and a measure of temporal coherence, as formulated in Eq. 1. The temporal coherence measure aggregates a local measure g computed between pairs of adjacent frames, taken to be the negative Euclidean distance between the center of $b^t_{j^t}$ and the forward-projected center of $b^{t-1}_{j^{t-1}}$ computed with optical flow. Eq. 1 can be computed in polynomial time using dynamic-programming with the Viterbi [15] algorithm. It does so by formulating a lattice, whose rows are indexed by j and whose columns are indexed by t, where the node at row j and column t is the detection b^t_j. Finding a track thus reduces to finding a path through this lattice.

$$\max_{k^1,\ldots,k^T} \sum_{t=1}^{T} h(k^t, b^t_{j^t}) + \sum_{t=2}^{T} a(k^{t-1}, k^t) \tag{2}$$

The second component recognizes events with hidden Markov models (HMMs), by finding a maximum *a posteriori* probability (MAP) estimate of an event model given a track. This is computed as shown in Eq. 2, where k^t denotes the state for frame t, $h(k, b)$ denotes the log probability of generating a detection b conditioned on being in

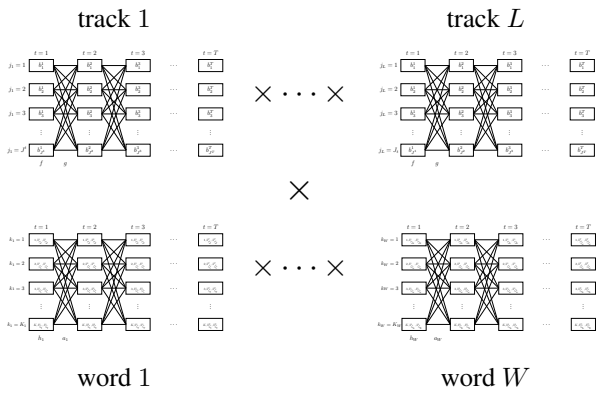


Figure 1. The cross-product lattice used by the sentence tracker, consisting of L tracking lattices and W event-model lattices.

state k, $a(k', k)$ denotes the log probability of transitioning from state k' to k, and $\hat{j}^t$ denotes the index of the detection produced by the tracker in frame t. This can also be computed in polynomial time using the Viterbi algorithm. Doing so induces a lattice, whose rows are indexed by k and whose columns are indexed by t.

The two components, detection-based tracking and event recognition, can be combined by combining the cost functions from Eq. 1 and Eq. 2 to yield a unified cost function

$$\max_{\substack{j^1,\ldots,j^T \\ k^1,\ldots,k^T}} \sum_{t=1}^{T} f(b^t_{j^t}) + \sum_{t=2}^{T} g(b^{t-1}_{j^{t-1}}, b^t_{j^t}) + \sum_{t=1}^{T} h(k^t, b^t_{j^t}) + \sum_{t=2}^{T} a(k^{t-1}, k^t)$$

that computes the joint MAP estimate of the best possible track and the best possible state sequence. This is done by replacing the $\hat{j}^t$ in Eq. 2 with j^t, allowing the joint maximization over detection and state sequences. This too can be computed in polynomial time with the Viterbi algorithm, finding the optimal path through a cross-product lattice where each node represents a detection paired with an event-model state. This formulation combines a single tracker lattice with a single event model, constraining the detection-based tracker to find a track that is not only temporally coherent but also satisfies the event model. This can be used to select that *ball* track from a video that contains multiple balls that exhibits the motion characteristics of an intransitive verb such as *bounce*.

One would expect that encoding the semantics of a complex sentence such as *The person to the right of the chair quickly carried the red object towards the trash can*, which involves nouns, adjectives, verbs, adverbs, and spatial-relation and motion prepositions, would provide substantially more mutual constraint on the *collection* of tracks for the participants than a single intransitive verb would constrain a single track. We thus extend the approach described above by incorporating a complex multi-argument predicate that represents the semantics of an entire sentence instead of one that only represents the semantics of a single intransitive verb. This involves formulating the semantics of other parts of speech, in addition to intransitive verbs, also as HMMs. We then construct a large cross-product lattice, illustrated in Fig. 1, to support L tracks and W words. Each node in this cross-product lattice represents L detections and the states for W words. To support L tracks, we subindex each detection index j as j_l for track l. Similarly, to support W words, we subindex each state index k as k_w for word w and the HMM parameters h and a for word w as h_w and a_w. The argument-to-track mappings θ^1_w and θ^2_w specify the tracks that fill arguments 1 and 2 (where necessary) of word w respectively. We then seek a path through this cross-product lattice that optimizes

$$\max_{\substack{j^1_1,\ldots,j^T_1 \\ j^1_L,\ldots,j^T_L \\ k^1_1,\ldots,k^T_1 \\ k^1_W,\ldots,k^T_W}} \sum_{l=1}^{L}\sum_{t=1}^{T} f(b^t_{j^t_l}) + \sum_{t=2}^{T} g(b^{t-1}_{j^{t-1}_l}, b^t_{j^t_l}) + \sum_{w=1}^{W}\sum_{t=1}^{T} h_w(k^t_w, b^t_{j^t_{\theta^1_w}}, b^t_{j^t_{\theta^2_w}}) + \sum_{t=2}^{T} a_w(k^{t-1}_w, k^t_w)$$

This can also be computed in polynomial time using the Viterbi algorithm. This describes a method by which the function $\mathcal{S}(D, \Phi) \mapsto (\tau, Z)$, discussed earlier, can be computed, where D is the collection of detections b^t_j and Z is the collection of tracks j^t_l.

3. Natural-Language Semantics

The sentence tracker uniformly represents the semantics of words in all parts of speech, namely nouns, adjectives, verbs, adverbs, and prepositions (both those that describe spatial relations and those that describe motion), as HMMs. Finite state recognizers (FSMs) are a special case of HMMs where the transition matrices a and the output models h are 0/1. Here, we formulate the semantics of a small fragment of English consisting of 17 lexical items (5 nouns, 2 adjectives, 4 verbs, 2 adverbs, 2 spatial-relation prepositions, and 2 motion prepositions), by hand, as FSMs. We do so to focus on what once can do with this approach, namely take sentences as input and focus the attention of a tracker, take video as input and produce sentential descriptions as output, and perform content-based video retrieval given a sentential input query, as discussed in Section 4. It is particularly enlightening that the FSMs we use are perspicuous and clearly encode pretheoretic human intuitions about the semantics of these words. But nothing turns on the use of hand-coded FSMs. Our framework, as described above, supports HMMs. A companion submission describes a method by which one can automatically learn such HMMs for the lexicon, grammar, and corpus discussed in this paper.

Nouns (*e.g. person*) may be represented by constructing static FSMs over discrete features, such as detector class. Adjectives (*e.g. red*, *tall*, and *big*) may be represented as

S → NP VP
NP → D [A] N [PP]
D → *an* | *the*
A → *blue* | *red*
N → *person* | *backpack* | *trash can* | *chair* | *object*
PP → P NP
P → *to the left of* | *to the right of*
VP → V NP [ADV] [PPM]
V → *picked up* | *put down* | *carried* | *approached*
ADV → *quickly* | *slowly*
PPM → PM NP
PM → *towards* | *away from*

(a)

to the left of = (agent patient) (referent)
to the right of = (agent patient) (referent)
picked up = (agent) (patient)
put down = (agent) (patient)
carried = (agent) (patient)
approached = (agent) (goal)
towards = (agent patient) (goal)
away from = (agent patient) (source)
other = (agent patient referent goal source)

(b)

1a. *The backpack approached the trash can.*
b. *The chair approached the trash can.*
2a. *The red object approached the chair.*
b. *The blue object approached the chair.*
3a. *The person to the left of the trash can put down an object.*
b. *The person to the right of the trash can put down an object.*
4a. *The person put down the trash can.*
b. *The person put down the backpack.*
5a. *The person carried the red object.*
b. *The person carried the blue object.*
6a. *The person picked up an object to the left of the trash can.*
b. *The person picked up an object to the right of the trash can.*
7a. *The person picked up an object.*
b. *The person put down an object.*
8a. *The person picked up an object quickly.*
b. *The person picked up an object slowly.*
9a. *The person carried an object towards the trash can.*
b. *The person carried an object away from the trash can.*
10. *The backpack approached the chair.*
11. *The red object approached the trash can.*
12. *The person put down the chair.*

(c)

Table 1. (a) The grammar for our lexicon of 19 lexical entries (2 determiners, 2 adjectives, 5 nouns, 2 spatial relations, 4 verbs, 2 adverbs, and 2 motion prepositions). Note that the grammar allows for infinite recursion in the noun phrase. (b) The theta grid, specifying the number of arguments and roles such arguments refer to. (c) A selection of sentences drawn from the grammar based on which we collected multiple videos for our corpus.

static FSMs that describe select properties of the detections for a single participant, such as color, shape, or size, independent of other features of the overall event. Intransitive verbs (*e.g*. *bounce*) may be represented as FSMs that describe the changing motion characteristics of a single participant, such as *moving downward* followed by *moving upward*. Transitive verbs (*e.g*. *approach*) may be represented as FSMs that describe the changing relative motion characteristics of two participants, such as *moving closer*. Adverbs (*e.g*. *slowly* and *quickly*) may be represented by FSMs that describe the velocity of a single participant, independent of the direction of motion. Spatial-relation prepositions (*e.g*. *to the left of*) may be represented as static FSMs that describe the relative position of two participants. Motion prepositions (*e.g*. *towards* and *away from*) may be represented as FSMs that describe the changing relative position of two participants. As is often the case, even simple static properties, such as detector class, object color, shape, and size, spatial relations, and direction of motion, might hold only for a portion of an event. We handle such temporal uncertainty by incorporating garbage states into the FSMs that always accept and do not affect the scores computed. This also allows for alignment between multiple words in a temporal interval during a longer aggregate event. We formulate the FSMs for specifying the word meanings as regular expressions over predicates computed from detections. The particular set of regular expressions and associated predicates that are used in the experiments are given in Table 2. The predicates are formulated around a number of primitive functions. The function *avgFlow*(b) computes a vector that represents the average optical flow inside the detection b. The functions $x(b)$, *model*(b), and *hue*(b) return the x-coordinate of the center of b, its object class, and the average hue of the pixels inside b respectively. The function *fwdProj*(b) displaces b by the average optical flow inside b. The functions $\angle$ and *angleSep* determine the angular component of a given vector and angular distance between two angular arguments respectively. The function *normal* computes a normal unit vector for a given vector. The argument v to NOJITTER denotes a specified direction represented as a 2D unit vector in that direction. Regular expressions are formulated around predicates as atoms. A given regular expression must be formed solely from output models of the same arity and denotes an FSM with a $-\infty$/0 transition matrix. We use a new regular-expression operator, $R^{[n,]} \triangleq (\text{R } [\text{TRUE}])^{\{n,\}}$ to indicate that R must be repeated at least n times but can optionally have a single frame of noise between each repetition. This allows for some flexibility in the models.

A sentence may describe an activity involving multiple tracks, where different (collections of) tracks fill the arguments of different words. This gives rise to the requirement of compositional semantics: dealing with the mappings from arguments to tracks. Given a sentence, say *The person to the right of the chair picked up the backpack*, argument-to-track assignment is a function $\mathcal{T}(\Lambda, \Gamma, \Psi) \mapsto (\Phi)$, that takes, as input, a sentence Λ and a grammar Γ, along with a specification of the argument arity and role types Ψ for the words in the lexicon and produces a formula Φ that specifies which tracks fill which arguments of which predicate instances for the words in the sentence. Such a function, applied to our example sentence with the grammar Γ as

Constants	Simple Predicates	Complex Predicates
$\textsc{xBoundary} \triangleq 300\text{px}$	$\textsc{noJitter}(b, v) \triangleq \|\textit{avgFlow}(b) \cdot v\| \leq \Delta\textsc{jump}$	$\textsc{stationaryClose}(b_1, b_2) \triangleq \textsc{stationary}(b_1) \wedge \textsc{stationary}(b_2) \wedge \neg\textsc{alike}(b_1, b_2) \wedge \textsc{close}(b_1, b_2)$
$\textsc{nextTo} \triangleq 50\text{px}$	$\textsc{alike}(b_1, b_2) \triangleq \textit{model}(b_1) = \textit{model}(b_2)$	$\textsc{stationaryFar}(b_1, b_2) \triangleq \textsc{stationary}(b_1) \wedge \textsc{stationary}(b_2) \wedge \neg\textsc{alike}(b_1, b_2) \wedge \textsc{far}(b_1, b_2)$
$\Delta\textsc{static} \triangleq 6\text{px}$	$\textsc{far}(b_1, b_2) \triangleq \|x(b_1) - x(b_2)\| \geq \textsc{xBoundary}$	$\textsc{closer}(b_1, b_2) \triangleq \|x(b_1) - x(b_2)\| > \|x(\textit{fwdProj}(b_1)) - x(b_2)\| + \Delta\textsc{closing}$
$\Delta\textsc{jump} \triangleq 30\text{px}$	$\textsc{close}(b_1, b_2) \triangleq \|x(b_1) - x(b_2)\| < \textsc{xBoundary}$	$\textsc{farther}(b_1, b_2) \triangleq \|x(b_1) - x(b_2)\| < \|x(\textit{fwdProj}(b_1)) - x(b_2)\| + \Delta\textsc{closing}$
$\Delta\textsc{quick} \triangleq 80\text{px}$	$\textsc{left}(b_1, b_2) \triangleq 0 < x(b_2) - x(b_1) \leq \textsc{nextTo}$	$\textsc{moveCloser}(b_1, b_2) \triangleq \textsc{noJitter}(b_1, (0, 1)) \wedge \textsc{noJitter}(b_2, (0, 1)) \wedge \textsc{closer}(b_1, b_2)$
$\Delta\textsc{slow} \triangleq 30\text{px}$	$\textsc{right}(b_1, b_2) \triangleq 0 < x(b_1) - x(b_2) \leq \textsc{nextTo}$	$\textsc{moveFarther}(b_1, b_2) \triangleq \textsc{noJitter}(b_1, (0, 1)) \wedge \textsc{noJitter}(b_2, (0, 1)) \wedge \textsc{farther}(b_1, b_2)$
$\Delta\textsc{closing} \triangleq 10\text{px}$	$\textsc{hasColour}(b, \text{hue}) \triangleq \textit{angleSep}(\textit{hue}(b), \text{hue}) \leq \Delta\textsc{hue}$	$\textsc{alongDir}(b, v) \triangleq \textit{angleSep}(\angle\textit{avgFlow}(b), \angle v) < \Delta\textsc{direction} \wedge \neg\textsc{stationary}(b)$
$\Delta\textsc{direction} \triangleq 30°$	$\textsc{stationary}(b) \triangleq \|\textit{avgFlow}(b)\| \leq \Delta\textsc{static}$	$\textsc{movingDir}(b, v) \triangleq \textsc{alongDir}(b, v) \wedge \textsc{noJitter}(b, \textit{normal}(v))$
$\Delta\textsc{hue} \triangleq 30°$	$\textsc{quick}(b) \triangleq \|\textit{avgFlow}(b)\| \geq \Delta\textsc{quick}$	$\textsc{approaching}(b_1, b_2) \triangleq \neg\textsc{alike}(b_1, b_2) \wedge \textsc{stationary}(b_2) \wedge \textsc{moveCloser}(b_1, b_2)$
	$\textsc{slow}(b) \triangleq \|\textit{avgFlow}(b)\| \leq \Delta\textsc{slow}$	$\textsc{departing}(b_1, b_2) \triangleq \neg\textsc{alike}(b_1, b_2) \wedge \textsc{stationary}(b_2) \wedge \textsc{moveFarther}(b_1, b_2)$
	$\textsc{isPerson}(b) \triangleq \textit{model}(b) = \textbf{person}$	$\textsc{pickingUp}(b_1, b_2) \triangleq \neg\textsc{alike}(b_1, b_2) \wedge \textsc{stationary}(b_1) \wedge \textsc{movingDir}(b_2, (0, 1))$
	$\textsc{isBackpack}(b) \triangleq \textit{model}(b) = \textbf{backpack}$	$\textsc{puttingDown}(b_1, b_2) \triangleq \neg\textsc{alike}(b_1, b_2) \wedge \textsc{stationary}(b_1) \wedge \textsc{movingDir}(b_2, (0, -1))$
	$\textsc{isTrashcan}(b) \triangleq \textit{model}(b) = \textbf{trashcan}$	$\textsc{carry}(b_1, b_2, v) \triangleq \textsc{movingDir}(b_1, v) \wedge \textsc{movingDir}(b_2, v)$
	$\textsc{isChair}(b) \triangleq \textit{model}(b) = \textbf{chair}$	$\textsc{carrying}(b_1, b_2) \triangleq \textsc{carry}(b_1, b_2, (0, 1)) \vee \textsc{carry}(b_1, b_2, (0, -1))$
	$\textsc{isBlue}(b) \triangleq \textsc{hasColour}(b, 225°)$	
	$\textsc{isRed}(b) \triangleq \textsc{hasColour}(b, 0°)$	

Regular Expressions

$\textsc{person} \triangleq \textsc{isPerson}^+$	$\textsc{blue} \triangleq \textsc{isBlue}^+$	$\textsc{pickedUp} \triangleq \textsc{stationaryClose}^+ \ \textsc{pickingUp}^{[3,]} \ \textsc{stationaryClose}^+$
$\textsc{backpack} \triangleq \textsc{isBackpack}^+$	$\textsc{red} \triangleq \textsc{isRed}^+$	$\textsc{putDown} \triangleq \textsc{stationaryClose}^+ \ \textsc{puttingDown}^{[3,]} \ \textsc{stationaryClose}^+$
$\textsc{trashcan} \triangleq \textsc{isTrashcan}^+$	$\textsc{quickly} \triangleq \textsc{true}^+ \ \textsc{quick}^{[3,]} \ \textsc{true}^+$	$\textsc{carried} \triangleq \textsc{stationaryClose}^+ \ \textsc{carrying}^{[3,]} \ \textsc{stationaryClose}^+$
$\textsc{chair} \triangleq \textsc{isChair}^+$	$\textsc{slowly} \triangleq \textsc{true}^+ \ \textsc{slow}^{[3,]} \ \textsc{true}^+$	$\textsc{approached} \triangleq \textsc{stationaryFar}^+ \ \textsc{approaching}^{[3,]} \ \textsc{stationaryClose}^+$
$\textsc{object} \triangleq (\textsc{isBackpack} \mid \textsc{isTrashcan} \mid \textsc{isChair})^+$	$\textsc{toTheLeftOf} \triangleq \textsc{left}^+$	$\textsc{towards} \triangleq \textsc{stationaryFar}^+ \ \textsc{approaching}^{[3,]} \ \textsc{stationaryClose}^+$
	$\textsc{toTheRightOf} \triangleq \textsc{right}^+$	$\textsc{awayFrom} \triangleq \textsc{stationaryClose}^+ \ \textsc{departing}^{[3,]} \ \textsc{stationaryFar}^+$

Table 2. The finite-state recognizers corresponding to the lexicon in Table 1(a).

specified in Table 1(a) and *theta grid* Ψ, as specified in Table 1(b), would produce the following formula.

$$\textsc{person}(P) \wedge \textsc{toTheRightOf}(P, Q) \wedge \textsc{chair}(Q) \wedge \textsc{pickedUp}(P, R) \wedge \textsc{backpack}(R)$$

To do so, we first construct a parse tree of the sentence Λ given the grammar Γ, using a recursive-descent parser, producing a parse tree.Such a parse tree encodes in its structure, the dependency relationships between different parts of speech as specified by the grammar. For each word, we then determine from the parse tree, which words in the sentence are determined to be its *dependents* in the sense of *government*, and how many such *dependents* exist, from the theta grid specified in Table 1(b). For example, the dependents of *to the right of* are determined to be *person* and *chair*, filling its first and second arguments respectively. Moreover, we determine a consistent assignment of roles, one of agent, patient, source, goal, and referent, for each participant track that fills the word arguments, from the allowed roles specified for that word and argument in the theta grid. Here, P, Q, and R are participants that play the agent, referent, and patient roles respectively.

4. Experimental Evaluation

The sentence tracker supports three distinct capabilities. It can take sentences as input and focus the attention of a tracker, it can take video as input and produce sentential descriptions as output, and it can perform content-based video retrieval given a sentential input query. To evaluate these, we filmed a corpus of 94 short videos, of varying length, in 3 different outdoor environments. The camera was moved for each video so that the varying background precluded unanticipated confounds. These videos, filmed with a variety of actors, each depicted one or more of the 21 sentences from Table 1(c). The depiction, from video to video, varied in scene layout and the actor(s) performing the event. The corpus was carefully constructed in a number of ways. First, many videos depict more than one sentence. In particular, many videos depict simultaneous distinct events. Second, each sentence is depicted by multiple videos. Third the corpus was constructed with minimal pairs: pairs of videos whose depicted sentences differ in exactly one word. These minimal pairs are indicated as the 'a' and 'b' variants of sentences 1–9 in Table 1(c). That varying word was carefully chosen to span all parts of speech and all sentential positions: sentence 1 varies subject noun, sentence 2 varies subject adjective, sentence 3 varies subject preposition, sentence 4 varies object noun, sentence 5 varies object adjective, sentence 6 varies object preposition, sentence 7 varies verb, sentence 8 varies adverb, and sentence 9 varies motion preposition. We filmed our own corpus as we are unaware of any existing corpora that exhibit the above properties. We annotated each of the 94 clips with ground truth judgments for each of the 21 sentences, indicating whether the given clip depicted the given sentence. This set of 1974 judgments was used for the following analyses.

4.1. Focus of Attention

Tracking is traditionally performed using cues from motion, object detection, or manual initialization on an object of interest. However, in the case of a cluttered scene involving multiple activities occurring simultaneously, there can be many moving objects, many instances of the same object class, and perhaps even multiple simultaneously occurring instances of the same event class. This presents a significant

obstacle to the efficacy of existing methods in such scenarios. To alleviate this problem, one can decide which objects to track based on which ones participate in a target event.

The sentence tracker can focus its attention on just those objects that participate in an event specified by a sentential description. Such a description can differentiate between different simultaneous events taking place between many moving objects in the scene using descriptions constructed out of a variety of parts of speech: nouns to specify object class, adjectives to specify object properties, verbs to specify events, adverbs to specify motion properties, and prepositions to specify (changing) spatial relations between objects. Furthermore, such a sentential description can even differentiate which objects to track based on the role that they play in an event: agent, patient, source, goal, or referent. Fig. 2 demonstrates this ability: different tracks are produced for the same video that depicts multiple simultaneous events when focused with different sentences.

We further evaluated this ability on all 9 minimal pairs, collectively applied to all 24 suitable videos in our corpus. For 21 of these, both sentences in the minimal pair yielded tracks deemed to be correct depictions. We include example videos for all 9 minimal pairs in the supplementary material.

4.2. Generation

Much of the prior work on generating sentences to describe images [4, 7, 8, 12, 13, 18] and video [1, 6, 9, 10, 16] uses special-purpose natural-language-generation methods. We can instead use the ability of the sentence tracker to score a sentence paired with a video as a general-purpose natural-language generator by searching for the highest-scoring sentence for a given video. However, this has a problem. Since h and a are $\log$ probabilities, g is a negative Euclidean distance, and we constrain f to be negative, scores decrease with longer word strings and greater numbers of tracks that result from longer word strings. So we don't actually search for the highest-scoring sentence, which would bias the process towards short sentences. Instead we seek complex sentences that are true of the video as they are more informative.

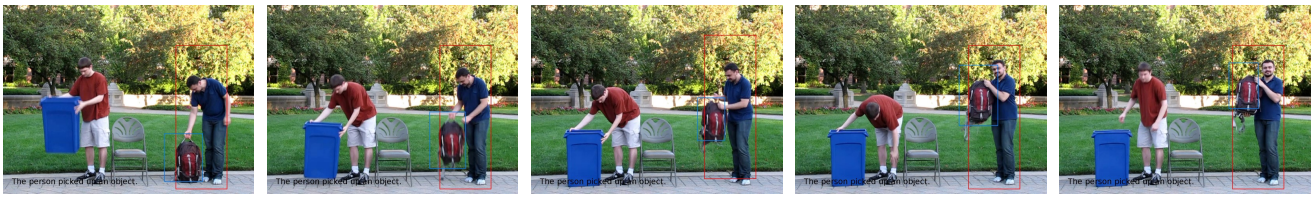
Nominally, this search process would be intractable since the space of possible sentences can be huge and even infinite. However, we can use beam search to get an approximate answer. This is possible because the sentence tracker can score any collection of words, not just complete phrases or sentences. We can select the k top-scoring single-word strings and then repeatedly extend the k top-scoring n-word strings, by one word, to select the k top-scoring $n+1$-word strings, subject to the constraint that these $n+1$-word strings can be extended to grammatical sentences by insertion of additional words. Thus we terminate the search process when the *contraction threshold*, the ratio between the score of an expanded string and the score of the string it expanded from, exceeds a specified value and the string being expanded is a complete sentence. This contraction threshold controls complexity of the generated sentence.

When restricted to FSMs, h and a will be 0/1 which become $-\infty$/0 in $\log$ space. Thus increase in the number of words can only decrease a score to $-\infty$, meaning that a string of words is no-longer true of a video. Since we seek true sentences, we terminate the above beam search process before the score goes to $-\infty$. In this case, there is no approximation: a beam search maintaining all n-word strings with finite score yields the highest-scoring sentence before the contraction threshold is met.

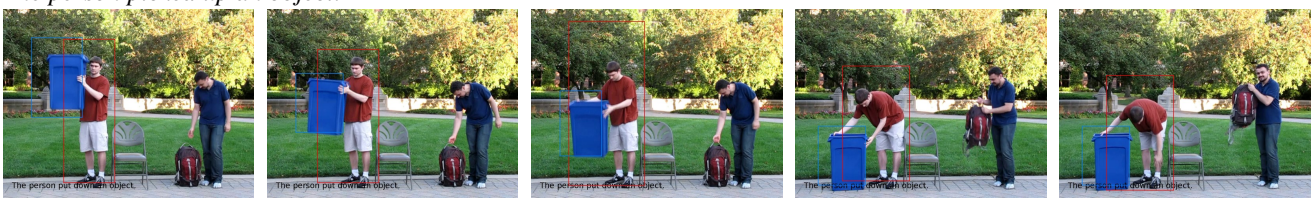
To evaluate this approach, we searched the space of sentences in the grammar in Table 1(a) to find the best true sentence for each of the 94 videos in our corpus. Note that the grammar generates an infinite number of sentences due to recursion in NP. Even restricting the grammar to eliminate NP recursion yields a space of 147,123,874,800 sentences. Despite not restricting the grammar in this fashion, we are able to effectively find good descriptions of the videos. We computed the accuracy of the sentence tracker in generating descriptions for all 94 videos in our corpus for multiple contraction thresholds. Accuracy was computed as the percentage of the 94 videos for which the sentence tracker produced descriptions that were deemed to be true. Contraction thresholds of 0.95, 0.90, and 0.85 yielded accuracies of 63.82%, 69.14%, and 64.89% respectively. We demonstrate examples of this approach in Fig. 3. The supplementary material contains additional examples.

4.3. Retrieval

The availability of vast video corpora, such as on YouTube, has created a rapidly growing demand for content-based video search and retrieval. The existing systems, however, only provide a means to search via human-provided captions. The inefficacy of such an approach is evident. Attempting to search for even simple queries such as *pick up* or *put down* yields surprisingly poor results, let alone searching for more complex queries such as *person approached horse*. Furthermore, prior work on content-based video-retrieval systems like Sivic and Zisserman [14] search only for objects and like Laptev *et al*. [11] search only for events. Even combining such to support conjunctive queries for videos with specified collections of objects jointly with a specified event, would not effectively rule out videos where the specified objects did not play a role in the event or played different roles in the event. For example, it could not rule out a video depicting a person jumping next to a stationary ball for a query *ball bounce* or distinguish between the queries *person approached horse* and *horse approached person*. The sentence tracker exhibits the ability to serve as the basis of a much better video search and retrieval tool, one that performs content-based search with complex sentential queries to find precise semantically relevant clips, as demonstrated in Fig. 4.

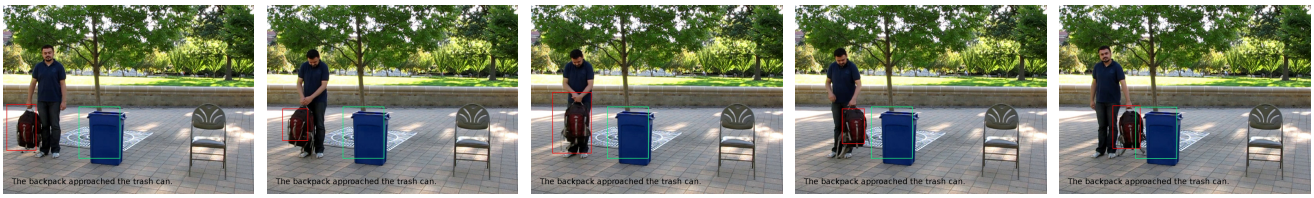


The person picked up an object.

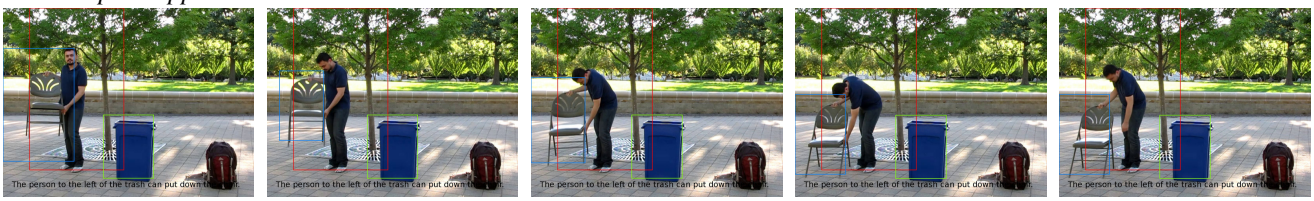


The person put down an object.

Figure 2. Sentence-guided focus of attention: different sets of tracks for the same video produced under guidance of different sentences.



The backpack approached the trash can.



The person to the left of the trash can put down the chair.

Figure 3. Generation of sentential descriptions: constructing the highest-scoring sentence for each video that is generated by the grammar in Table 1(a), by means of a beam search.

To evaluate this approach, we scored every video in our corpus against every sentence in Table 1(c), rank ordering the videos for each sentence, yielding the following statistics over the 1974 scores.

chance that a random video depicts a given sentence	*13.12%*
top-scoring video depicts the given sentence	*85.71%*
at least 1 of the top 3 scoring videos depicts the given sentence	*100.00%*

The judgment of whether a video depicted a given sentence was made using our annotation. We conducted an additional evaluation with this annotation. One can threshold the sentence-tracker score to yield a binary predicate on video-sentence pairs. We performed 4-fold cross validation on our corpus, selecting the threshold for each fold that maximized accuracy of this predicate, relative to the annotation, on 75% of the videos and evaluating the accuracy with this selected threshold on the remaining 25%. This yielded an average accuracy of 91.74%.

5. Conclusion

We have presented a novel framework that utilizes the compositional structure of events and the compositional structure of language to drive a semantically meaningful and targeted approach towards activity recognition. This multimodal framework integrates low-level visual components, such as object detectors, with high-level semantic information in the form of sentential descriptions in natural language. This is facilitated by the shared structure of detection-based tracking, which incorporates the low-level object-detector components, and of finite-state recognizers, which incorporate the semantics of the words in a lexicon.

We demonstrated the utility and expressiveness of our framework by performing three separate tasks on our corpus, requiring no training or annotation, simply by leveraging our framework in different manners. The first, sentence-guided focus of attention, showcases the ability to focus the attention of a tracker on the activity described in a sentence, indicating the capability to identify such subtle distinctions as between *The person picked up the chair to the left of the trash can* and *The person picked up the chair to the right of the trash can*. The second, generation of sentential description of video, showcases the ability to produce a complex description of a video, involving multiple parts of speech,

The person carried an object away from the trash can.

The person picked up an object to the left of the trash can.

Figure 4. Sentential-query-based video search: returning the best-scoring video, in a corpus of 94 videos, for a given sentence.

by performing an efficient search for the best description though the space of all possible descriptions. The final task, query-based video search, showcases the ability to perform content-based video search and retrieval, allowing for such distinctions as between *The person approached the trash can* and *The trash can approached the person*.

Acknowledgments

This research was sponsored by the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-10-2-0060. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either express or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes, notwithstanding any copyright notation herein.

References

[1] A. Barbu, A. Bridge, Z. Burchill, D. Coroian, S. Dickinson, S. Fidler, A. Michaux, S. Mussman, N. Siddharth, D. Salvi, L. Schmidt, J. Shangguan, J. M. Siskind, J. Waggoner, S. Wang, J. Wei, Y. Yin, and Z. Zhang. Video in sentences out. In *Proceedings of the Twenty-Eighth Conference on Uncertainty in Artificial Intelligence*, pages 102–112, Aug. 2012. 6

[2] A. Barbu, N. Siddharth, A. Michaux, and J. M. Siskind. Simultaneous object detection, tracking, and event recognition. *Advances in Cognitive Systems*, 2:203–220, Dec. 2012. 1, 2

[3] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes (VOC) challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010. 1

[4] A. Farhadi, M. Hejrati, M. Sadeghi, P. Young, C. Rashtchian, J. Hockenmaier, and D. Forsyth. Every picture tells a story: Generating sentences from images. In *Proceedings of the European Conference on Computer Vision*, pages 15–29, 2010. 6

[5] P. F. Felzenszwalb, R. B. Girshick, and D. McAllester. Cascade object detection with deformable part models. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2241–2248, 2010. 2

[6] C. Fernández Tena, P. Baiget, X. Roca, and J. Gonzàlez. Natural language descriptions of human behavior from video sequences. In *Advances in Artificial Intelligence*, pages 279–292, 2007. 6

[7] A. Gupta, Y. Verma, and C. Jawahar. Choosing linguistics over vision to describe images. In *Proceedings of the Twenty-Sixth National Conference on Artificial Intelligence*, 2012. 6

[8] L. Jie, B. Caputo, and V. Ferrari. Who's doing what: Joint modeling of names and verbs for simultaneous face and pose annotation. In *Proceedings of the Neural Information Processing Systems Conference*, 2009. 6

[9] M. U. G. Khan and Y. Gotoh. Describing video contents in natural language. In *Proceedings of the Workshop on Innovative Hybrid Approaches to the Processing of Textual Data*, pages 27–35, 2012. 6

[10] A. Kojima, T. Tamura, and K. Fukunaga. Natural language description of human activities from video images based on concept hierarchy of actions. *International Journal of Computer Vision*, 50(2):171–184, Nov. 2002. 6

[11] I. Laptev, M. Marszałek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2008. 6

[12] P. Li and J. Ma. What is happening in a still picture? In *First Asian Conference on Pattern Recognition*, pages 32–36, Nov. 2011. 6

[13] M. Mitchell, J. Dodge, A. Goyal, K. Yamaguchi, K. Stratos, X. Han, A. Mensch, A. C. Berg, T. L. Berg, and H. Daumé III. Midge: Generating image descriptions from computer vision detections. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, 2012. 6

[14] J. Sivic and A. Zisserman. Video Google: a text retrieval approach to object matching in videos. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 1470–1477, 2003. 6

[15] A. J. Viterbi. Convolutional codes and their performance in

communication systems. *IEEE Transactions on Communication*, 19:751–772, Oct. 1971. 2

[16] Z. Wang, G. Guan, Y. Qiu, L. Zhuo, and D. Feng. Semantic context based refinement for news video annotation. *Multimedia Tools and Applications*, pages 1–21, 2012. 6

[17] J. K. Wolf, A. M. Viterbi, and G. S. Dixon. Finding the best set of K paths through a trellis with application to multitarget tracking. *IEEE Transactions on Aerospace and Electronic Systems*, 25(2):287–296, Mar. 1989. 1

[18] Y. Yang, C. L. Teo, H. Daumé III, and Y. Aloimonos. Corpus-guided sentence generation of natural images. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 444–454, 2011. 6