Recognition of Visual Activities and Interactions by Stochastic Parsing

Yuri Ivanov and Aaron Bobick

presented by David Ross
March 24, 2005

Motivation

- recognize complex action sequences in video
- sequences are "structurally defined relationships of primitives"
- two level approach: recognize...
  - low level primitives using statistical detection
  - actions (configurations of primitives) using stochastic context free grammars (SCFG)

Context-Free Grammars

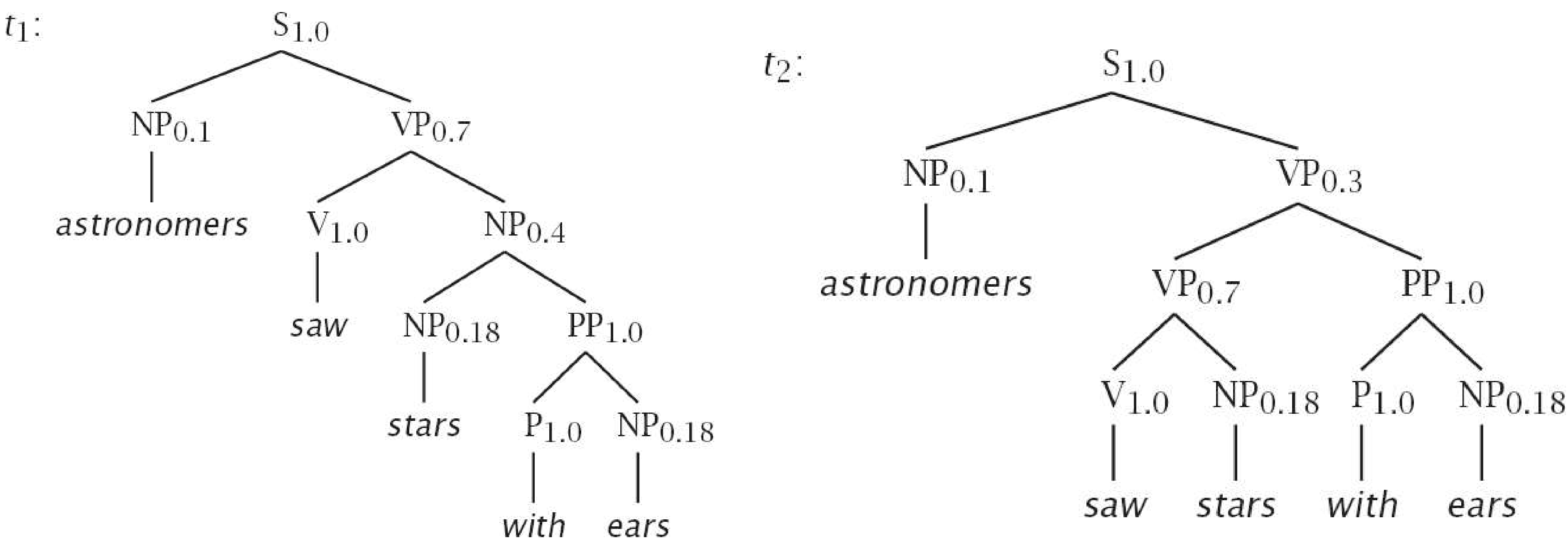
- start symbol, non-terminals, terminals, rules
- context-free: LHS of rule is single non-terminal

S → NP VP
PP → P NP
VP → V NP
VP → VP PP
P → *with*
V → *saw*

NP → NP PP
NP → *astronomers*
NP → *ears*
NP → *saw*
NP → *stars*
NP → *telescopes*

Alternative Parses

- “astronomers saw stars with ears”



Stochastic/Probabilistic CFGs

- add probabilities to each rule
- probability of a parse tree is product of rule probabilities

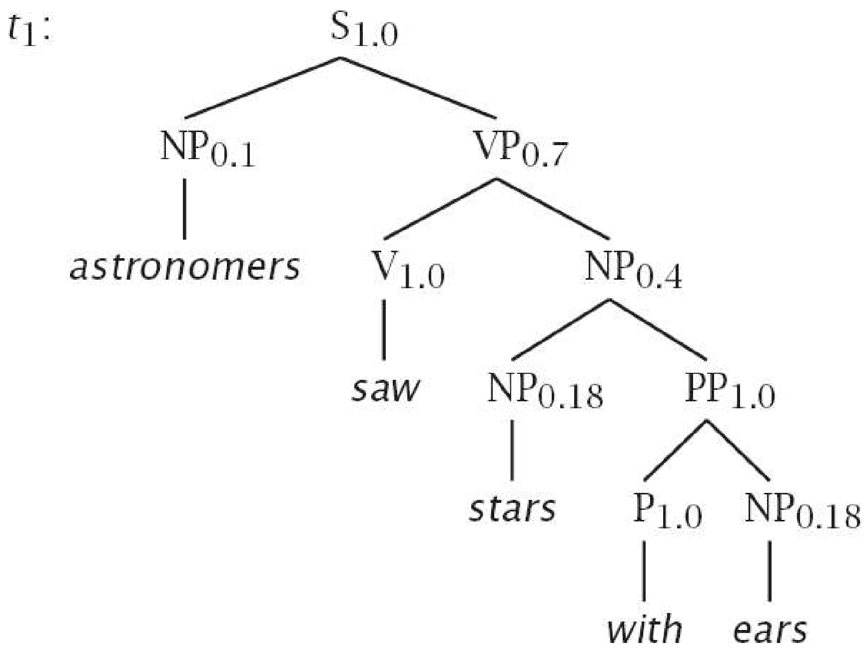
S → NP VP	1.0	NP → NP PP	0.4
PP → P NP	1.0	NP → *astronomers*	0.1
VP → V NP	0.7	NP → *ears*	0.18
VP → VP PP	0.3	NP → *saw*	0.04
P → *with*	1.0	NP → *stars*	0.18
V → *saw*	1.0	NP → *telescopes*	0.1

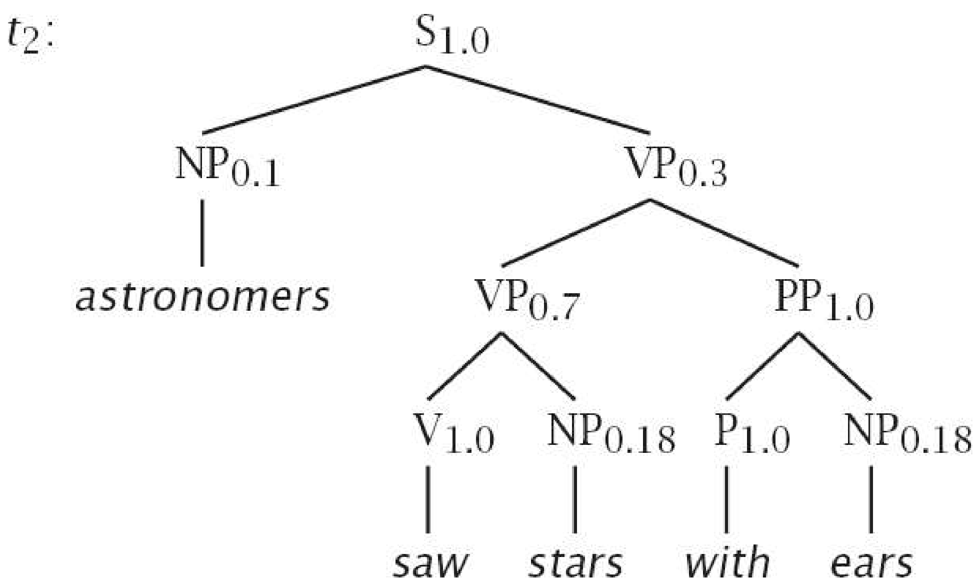
Alternative Parses

- compare probabilities

$P(t_1) = 9.072e\text{-}4$

$P(t_2) = 6.804e\text{-}4$





Parsing Algorithm

- begin with start symbol, S
- predict – expand all non-terminals, via a left-most parse
- scan – find partial parses that match the current input symbol
- complete – when a parse subtree is complete, work to the next non-terminal; advance to next input symbol

Parsing Example

S	$\rightarrow$	NP VP	Det	$\rightarrow$	a
NP	$\rightarrow$	Det N	N	$\rightarrow$	circle | square | triangle
VP	$\rightarrow$	VT NP	VT	$\rightarrow$	touches
VP	$\rightarrow$	VI PP	VI	$\rightarrow$	is
PP	$\rightarrow$	P NP	P	$\rightarrow$	above | below

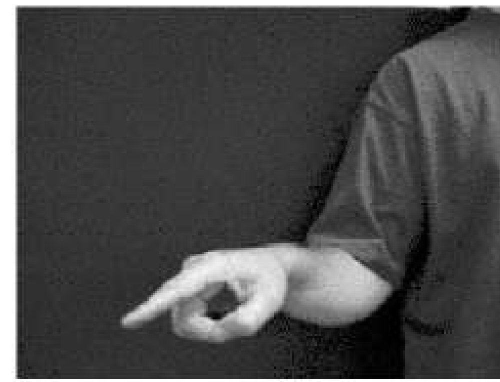
(b)

	a	circle	touches	a	square
$_0$ $\rightarrow$.*S*	*scanned*	*scanned*	*scanned*	*scanned*	*scanned*
predicted	$_0$Det $\rightarrow$ a.	$_1$N $\rightarrow$ circle.	$_2$VT $\rightarrow$ touches.	$_3$Det $\rightarrow$ a.	$_4$N $\rightarrow$ triangle.
$_0$S $\rightarrow$ .NP VP	*completed*	*completed*	*completed*	*completed*	*completed*
$_0$NP $\rightarrow$ .Det N	$_0$NP $\rightarrow$ Det.N	$_0$NP $\rightarrow$ Det N.	$_2$VP $\rightarrow$ VT.NP	$_3$NP $\rightarrow$ Det.N	$_4$NP $\rightarrow$ Det N.
$_0$Det $\rightarrow$ .a	*predicted*	$_0$S $\rightarrow$ NP.VP	*predicted*	*predicted*	$_3$VP $\rightarrow$ VT NP.
	$_1$N $\rightarrow$ .circle	*predicted*	$_3$NP $\rightarrow$ .Det N	$_5$N $\rightarrow$ .circle	$_0$S $\rightarrow$ NP VP.
	$_1$N $\rightarrow$ .square	$_2$VP $\rightarrow$ .VT NP	$_3$Det $\rightarrow$ .a	$_4$N $\rightarrow$ .square	$_0$ $\rightarrow$ S.
	$_1$N $\rightarrow$ .triangle	$_2$VP $\rightarrow$ .VI PP		$_4$N $\rightarrow$ .triangle	
		$_2$VT $\rightarrow$ .touches			
		$_2$VI $\rightarrow$ .is			
State set 0	1	2	3	4	5

Example: Structured Gesture

(a)

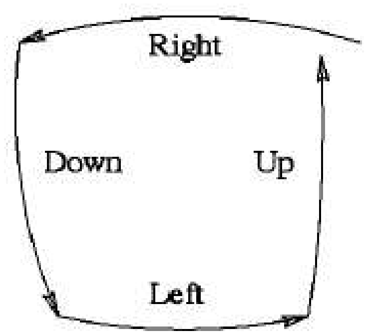
(b)



(c)

(d)

(e)



(f)

Structured Gesture: SCFG

$$
\begin{array}{llll}
G_{square}: & & & \\
\texttt{SQUARE} & \rightarrow & \texttt{RH} & [0.5] \\
 & | & \texttt{LH} & [0.5] \\
\texttt{RH} & \rightarrow & \texttt{TOP up-down BOT down-up} & [1.0] \\
\texttt{LH} & \rightarrow & \texttt{BOT down-up TOP up-down} & [1.0] \\
\texttt{TOP} & \rightarrow & \texttt{left-right} & [0.5] \\
 & | & \texttt{right-left} & [0.5] \\
\texttt{BOT} & \rightarrow & \texttt{right-left} & [0.5] \\
 & | & \texttt{left-right} & [0.5]
\end{array}
$$

- notice any problems?

Recognizing Gesture Primitives

- Hidden Markov Model (HMM)
  - state-space model, with discrete state variables
  - must provide a mapping from states to observations
  - can be fit to training data using EM, providing a density model
  - Viterbi (aka dynamic programming, max-product)
- fit one HMM to each primitive
- use them backwards from current time... each outputs probability and starting time

Output of Primitive Detector

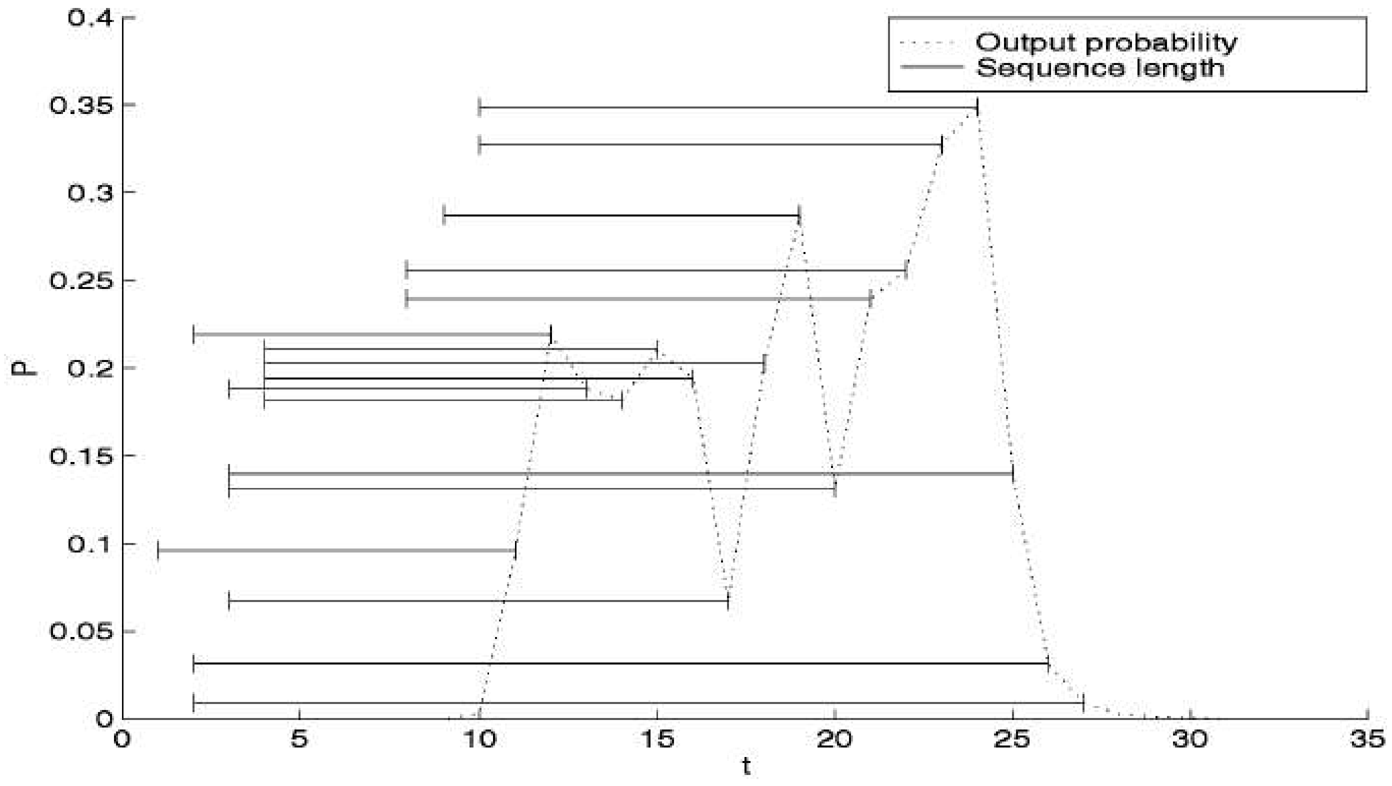
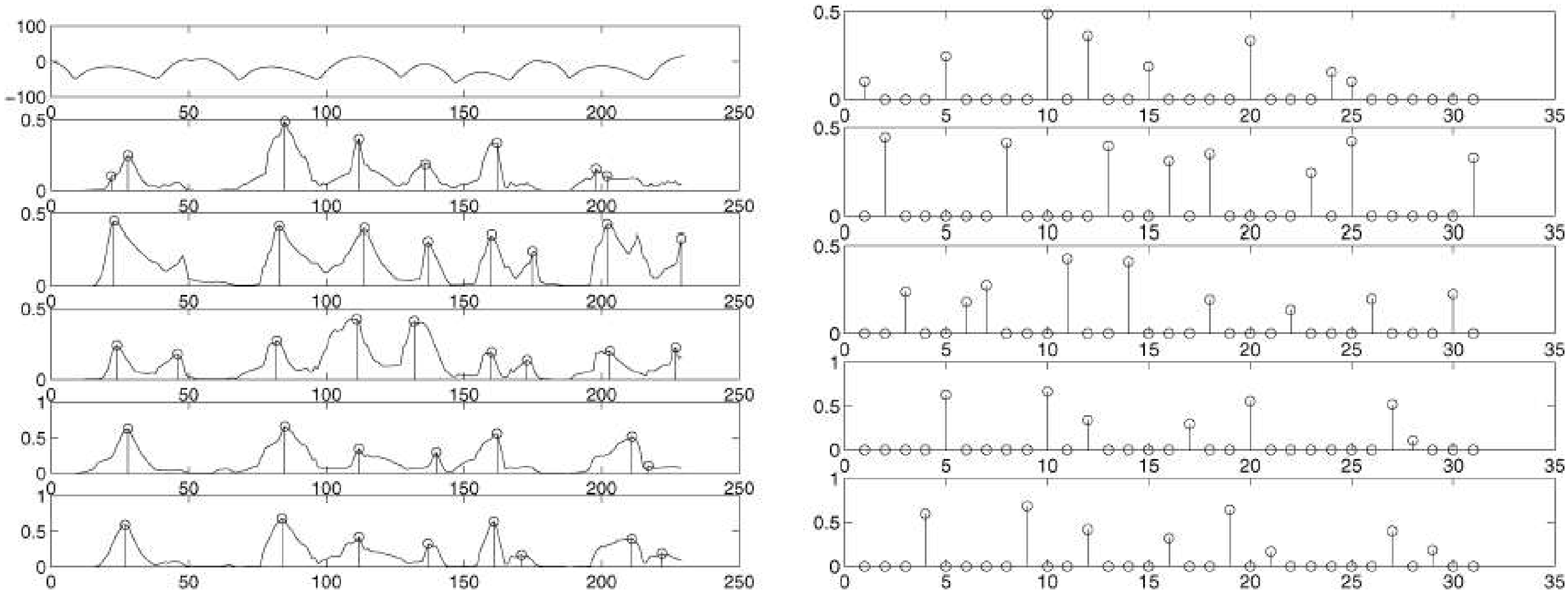


Fig. 2. Output of a component model. Here, at every sample, the activity primitive, modeled by the HMM, outputs a model likelihood. Each point of the probability plot is the normalized maximum likelihood of the HMM

Produce Discrete Events



- threshold HMM probability, search for local maxima, discretize, discard times lacking detections

Problems Producing Discrete Symbols

1) uncertainty in the observation (substitution errors)

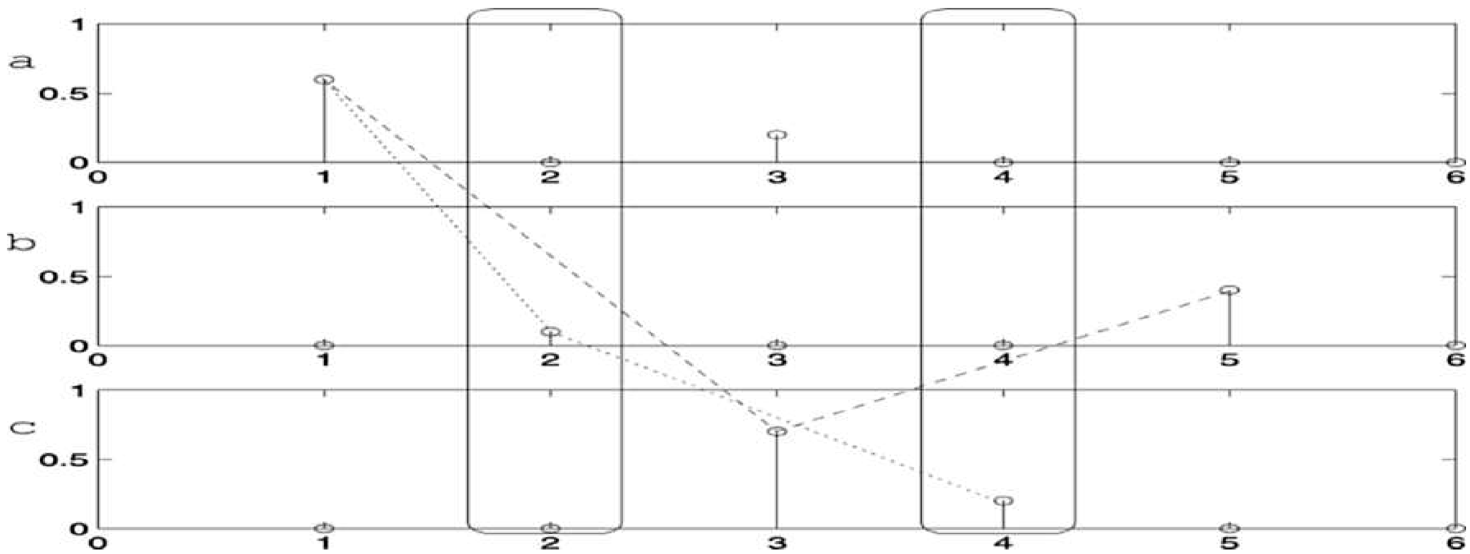
"a game of cat and {mouse,house}"

2) spurious detections (insertion errors)

"a game of 9 cat and mouse"

3) ensuring events don't overlap (temporal consistency)

1. Uncertainty in Observations



- each detection has a probability
- "multivalued string"
- when parsing, multiply parse tree by probability of symbol

2. Dealing with Insertion Errors

- sources of insertions
  - noise in component detectors
  - other actions going on in the video
- robustify grammar

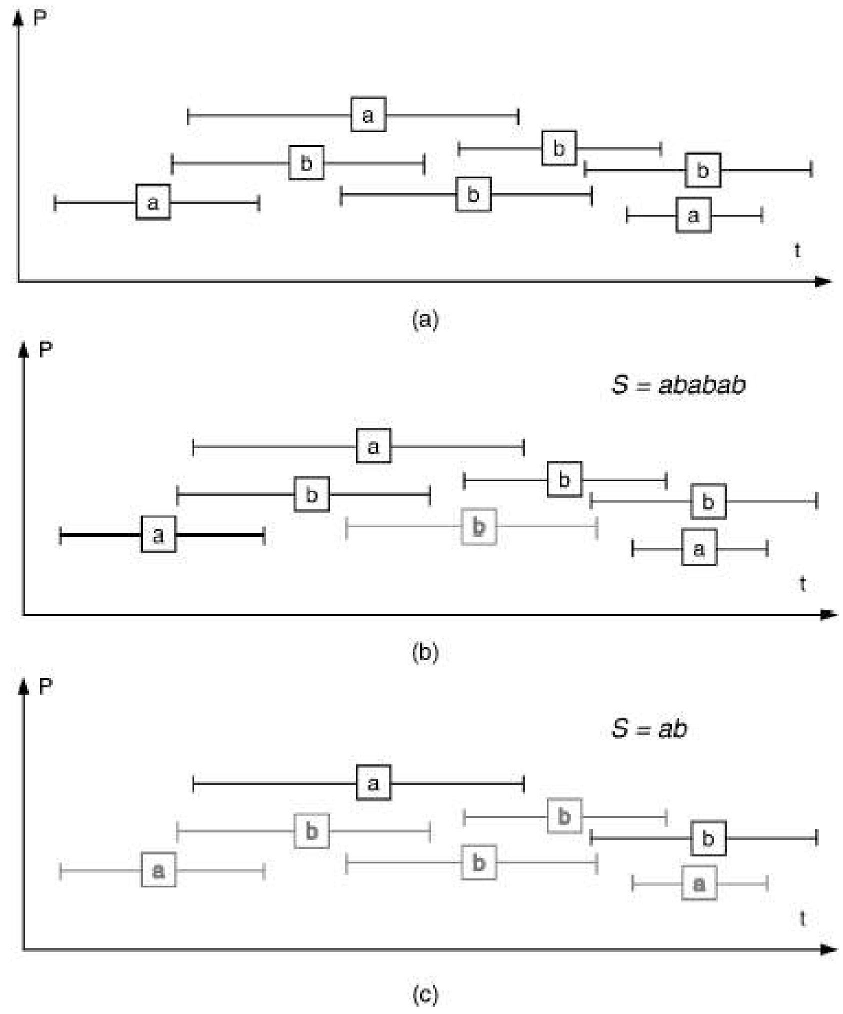
A: b C

A: B C
B: b | b SKIP | SKIP b
SKIP: a | b | c | ...

- assign low probability to SKIP

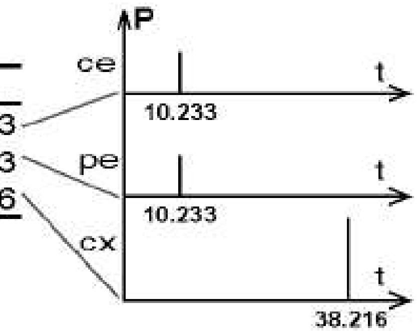
3. Temporal Consistency

- terminals should be non-overlapping
- when parsing, multiply prob. by a compatibility function f(d) ε [0,1]



Example: Surveillance

Event	Likelihood	x	y	dx	dy	time
car-enter	0.5	0.454	1	-0.01	0.05	10.233
person-enter	0.5	0.454	1	-0.01	0.05	10.233
car-exit	1	1	0.784	0.1	0.1	38.216



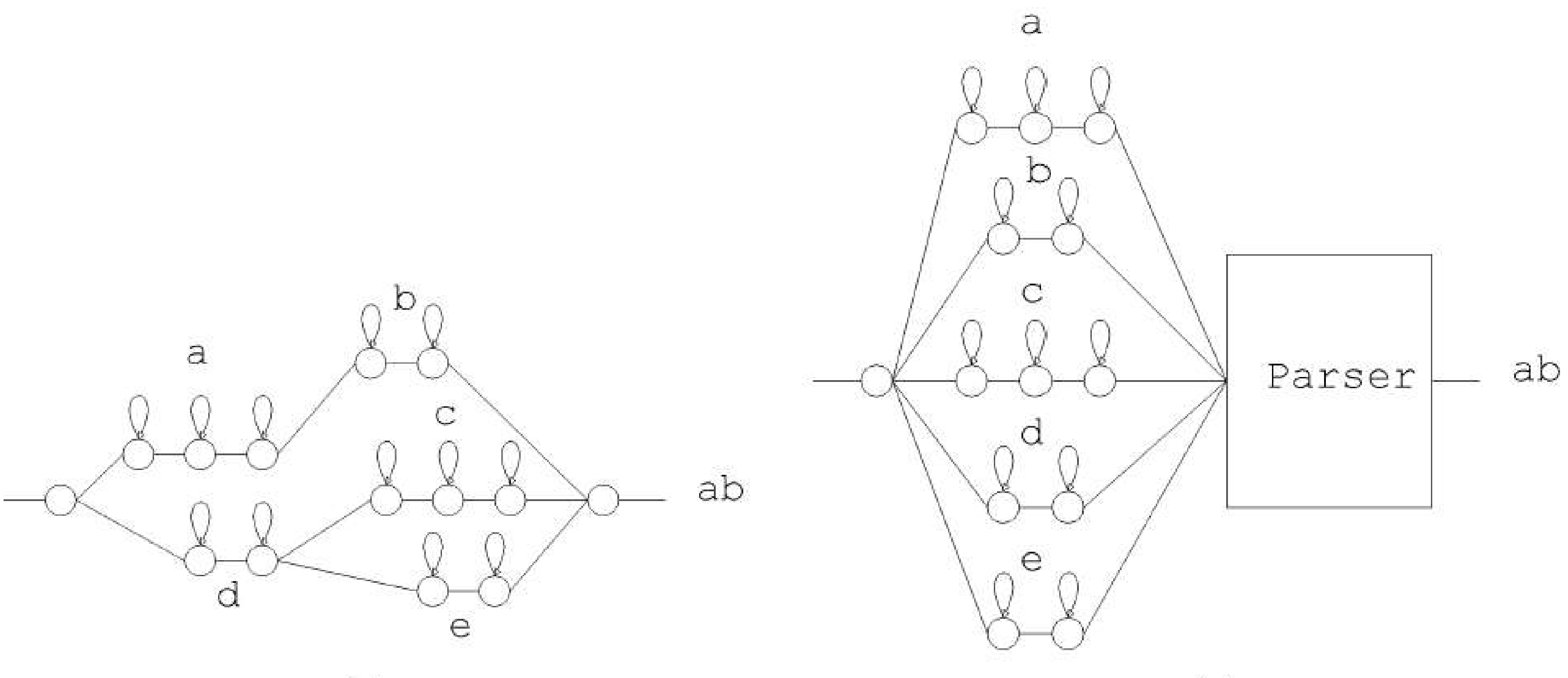
- primitive detection:
  - track moving blobs
  - label as car or person (probabilistically)
  - from tracks, generate discrete events using rules (6.1.2)
  - {person,car} + {enter, found, exit, lost, stopped}

Rule Probabilities

- not learned – set to uniform
- might be hard to set manually
- can estimate from data using EM
- square example: grammar not even ambiguous
- only important probability: SKIP rule (probability of insertion errors)

SCFG vs. HMM

- advantages to using SCFG over HMM for complex action recognition?



(a) (b)