From Fu to Zhu: Stochastic Image Grammars and their Siblings for Activity Recognition in Videos

Rama Chellappa

University of Maryland, College Park, MD

Outline of the Talk

- Early efforts in stochastic grammars for shape analysis
- Activity recognition in videos
- Context-free grammars for activity recognition
- Stochastic Petri nets
- Probabilistic language for activity recognition
- What happened to ontologies for activity recognition?
- Some observations
  - Why stochastic grammars are in boutique stores and SVMs are in 7-11!
  - Lack of results on bounds on probability of error

Thanks to

Early years (70’s and 80’s)

Dr. R. Narasimhan, TIFR, Labeling schemata and syntactic descriptions of pictures, Information and Control, Vol. 7, pp. 151-179, June 1964, while at University of Illinois.

Prof. K.S. Fu at Purdue University

Got out of statistical pattern recognition in the late sixties and vigorously pursued syntactic pattern recognition. Initiated the development of stochastic image grammars in the late seventies, and designed various stochastic inference rules for syntactic pattern recognition. Mostly dealt with 2-D shape and texture analysis.

Grammatical inference Fu and Booth, IEEE Trans. on SMC 1975.

The field almost died when Prof. Fu passed away.

Prof. A. Rosenfeld at University of Maryland (IEEE Symposium on Decision and Control, 1970)

Abstract: Picture grammars: In recent years there has been considerable interest in applying the methods of mathematical linguistics to picture generation and description [1]. In this approach, pictures are regarded as combinations of subpictures, which are in turn built up out of still smaller parts, in analogy with the way that sentences can be broken down into phrases and words. Conventionally, however, mathematical linguistics deals with strings (of words, etc.), whereas pictures do not usually have natural representations as strings of subpictures. This suggests that it would be desirable to generalize the tools of mathematical linguistics so as to allow combining parts into wholes by methods more general than string concatenation.

Syntactic pattern recognition – Early years

Grammars were first used in image/scene modeling by K.S Fu’s school in the 1970-80s (Purdue Univ).

SCFGs were illustrated in “block world” and string/web grammars are used instead.

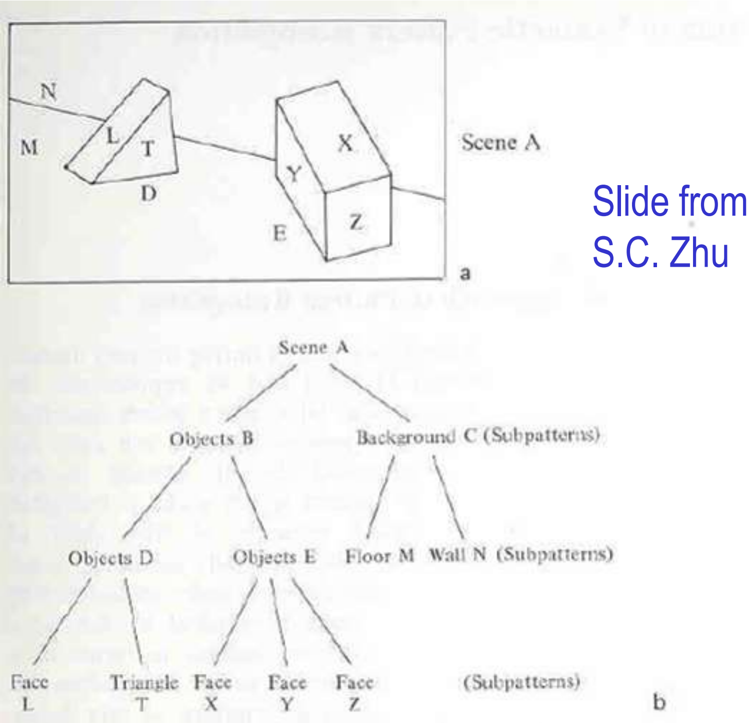
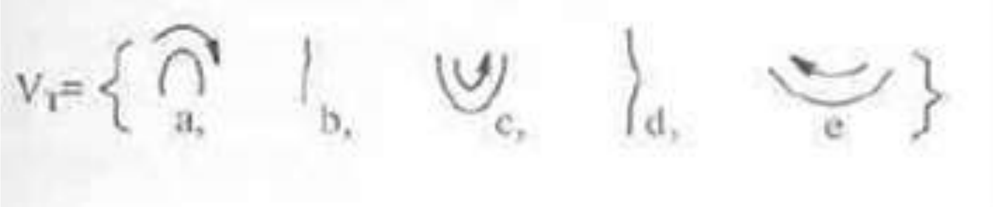


Fig. 1.1a and b. The pictorial pattern *A* and its hierarchical structural descriptions

Slide from S.C. Zhu



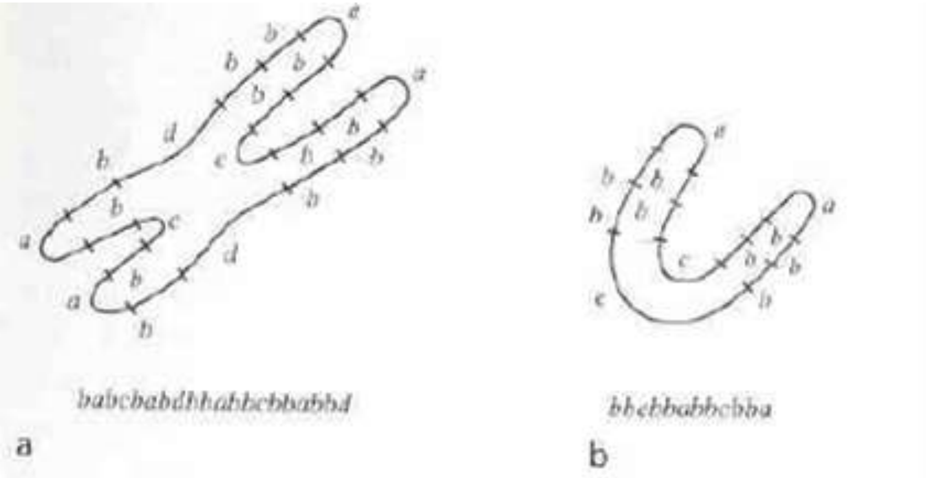


Fig. 1.7. (a) Submedian chromosome; (b) Telocentric chromosome

Other works

- Gonzalez and Thomason, Syntactic pattern recognition, 1978.
- Regular grammars for 2-D shape analysis
  - Ali and Pavlidis, PAMI 1979
- Azriel Rosenfeld, Picture languages, 1979.
  - Wrote this book while traveling
- Syntactic and structural methods for pattern recognition Bunke and Thomason, 1990.
- Syntactic methods for time varying imagery – Fan and Fu, 1979, Fu and Fan, IEEE Trans. SMC, 1986.

Activity recognition in video: Motivation

- Airport Surveillance
  - Identify suspicious activities in baggage areas, tarmac areas, and terminals
- Bank Surveillance
  - Identify suspicious activities at ATMs and in lobby areas
- Military site security
  - Identify suspicious activities at or near military sites
- Building security
  - Identify suspicious vehicles
- Retail stores
  - Understanding customer’s shopping preferences
  - Catching shoplifters

Example: Bank surveillance

Sequencing
Multi-threading
Synchronization

Example 2: Surveillance

Synchronization
Concurrence
Multiple instances

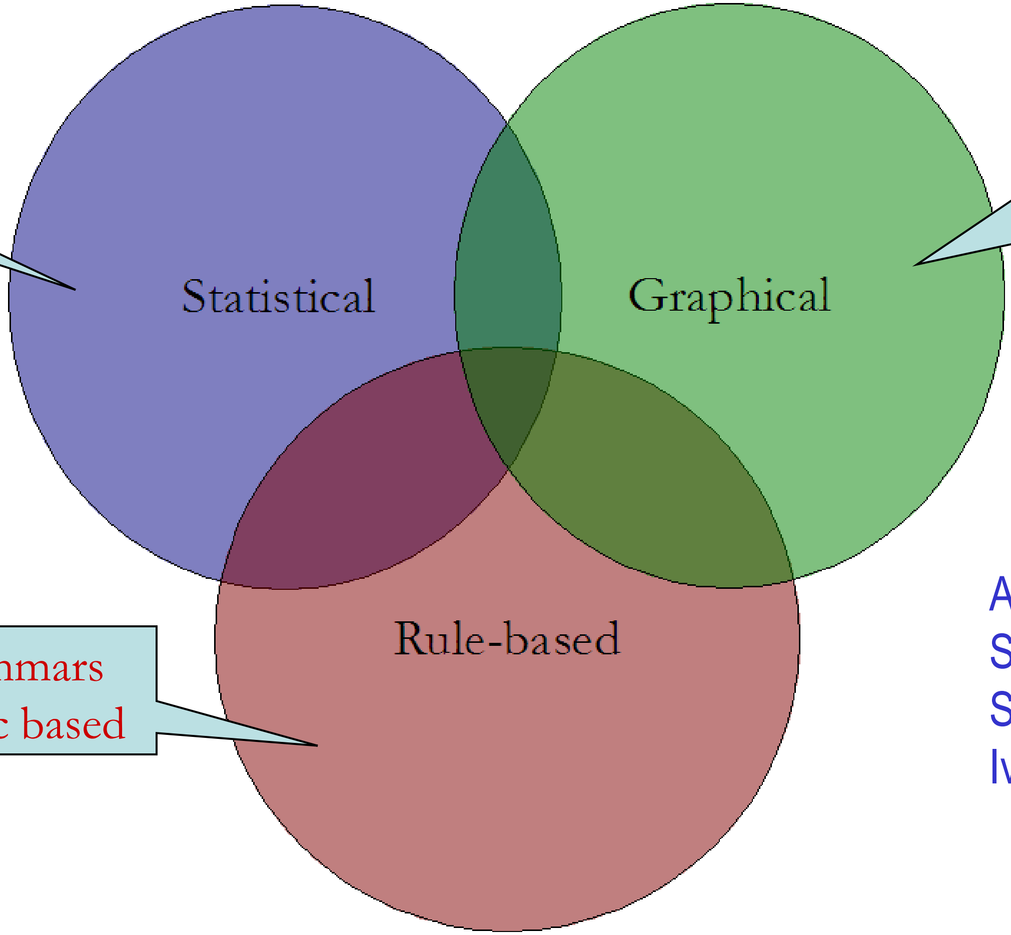
Activity recognition - Challenges

- Complex multi-agent interactions.
- Huge variations of the same activity.
  - Atleast 10 variations of transferring an object
- Notions of ‘mean’ and ‘variance’ of an activity ill-defined.
- Semantics of activities may not conform to strict statistical or structural constraints.
- Limited view and rate invariances are desired.
- Multiple events produce similar videos (identifiability issues)

Major approaches for activity modeling

Pentland

- HMM’s
- SLDS



- Markov Chains
- FSM
- DBNs

- Grammars
- Logic based

Aho and Petersen, 1972
Stolcke, 1994
Sanfeliu, 1996
Ivanov and Bobick, 1997

Shet and Davis,
IJCV Spl. issue

Event Modeling Based on Syntactic Representations

- Overview
  - Representation using attribute grammar
    - Extension of attribute grammar
  - Recognition by online parsing
  - Results
    - Specific event recognition
    - Anomaly detection
- Attribute grammar
  - Grammar: strings naturally correspond to sequences, compact
  - Attributes: describe general features, allow events with multiple objects & concurrent events
- Woo and Chellappa, 2007.

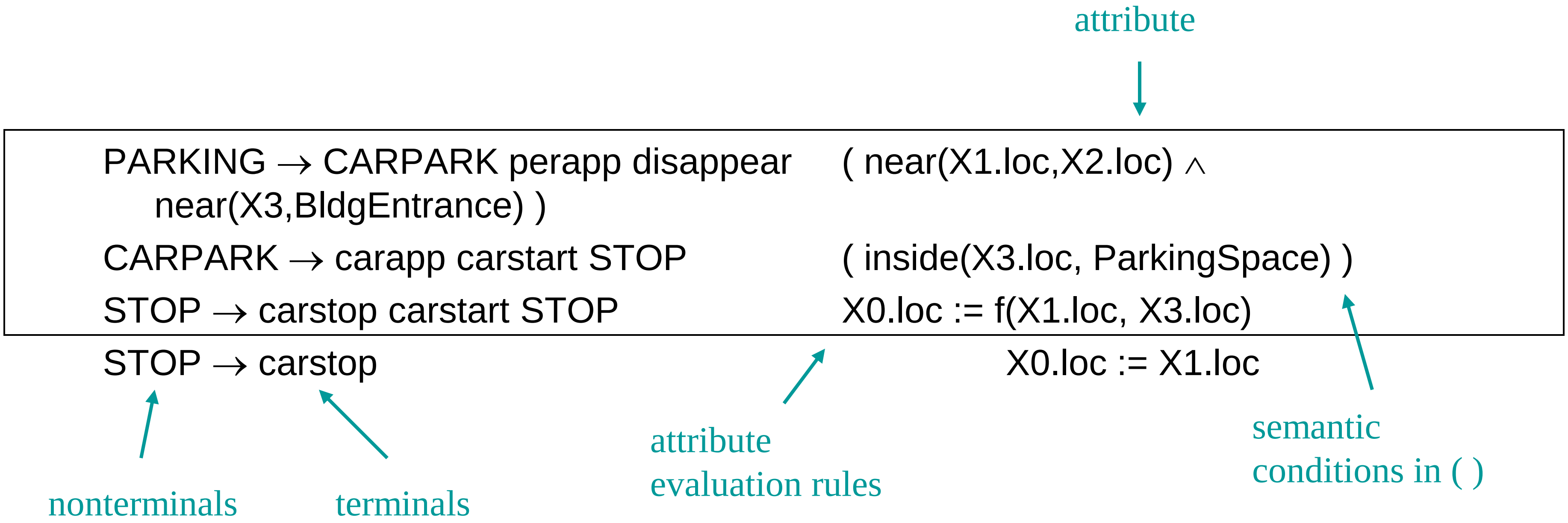
Primitives

- Input symbols correspond to “primitive events”
  - e.g., *stop*, *disappear*
- Attributes
  - Additional features associated with primitive event
  - Features that cannot be represented by (finite) input symbols
  - e.g., *location*, object *id*

Attribute grammar

- Definition $AG = (G, SD, AD, R, C)$
  - $G = (V_N, V_T, P, S)$: Context free grammar. Defines syntax.
  - *SD* : Semantic domain. Set of attribute types and functions.
  - *AD* : Set of ***attributes*** associated with each symbol in *P*
    - Attributes are interpreted as “semantics”
  - *R* : Set of ***attribute evaluation rules***
    - Functions that determines attribute values
  - *C* : Set of ***semantic conditions*** (predicates) on attributes
    - For each production
    - The production can be used only if the conditions are satisfied

Example



attribute

PARKING → CARPARK perapp disappear (near(X1.loc,X2.loc) ∧ near(X3,BldgEntrance))

CARPARK → carapp carstart STOP (inside(X3.loc, ParkingSpace))

STOP → carstop carstart STOP X0.loc := f(X1.loc, X3.loc)

STOP → carstop X0.loc := X1.loc

nonterminals

terminals

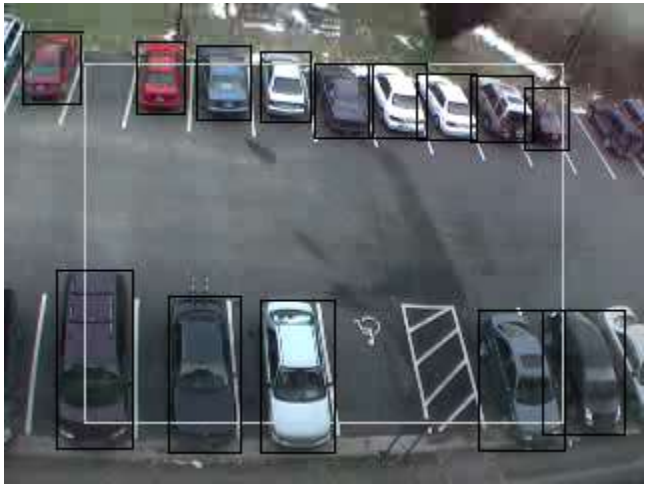
attribute evaluation rules

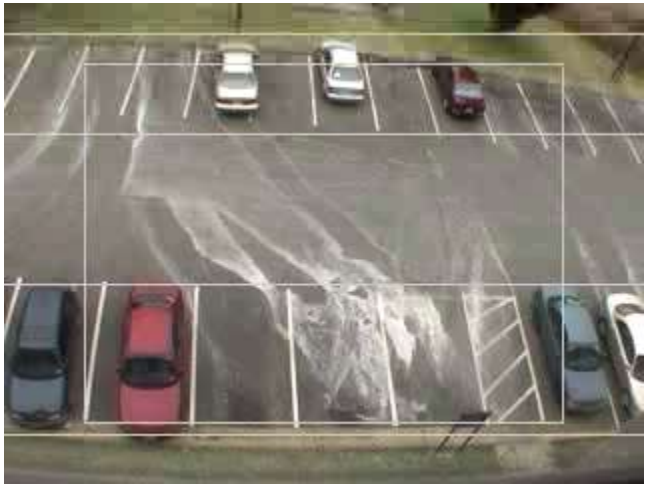
semantic conditions in ( )

Recognition

- Parsing based on Earley's algorithm (1970)
- Error correcting grammar for handling tracking error
- Two types of applications
  - Recognize specific events: Ignore negative patterns
  - Detect anomalies: Classify into positive & negative patterns
- Event recognition system
  - Track moving objects, classify into person / vehicle, generate primitive events
  - Positive recognition
    - Successfully parsed (syntactic) with high confidence (in attribute condition)
  - Anomaly detection
    - Parse failed or parsed with low confidence

Specific Event Recognition: Results



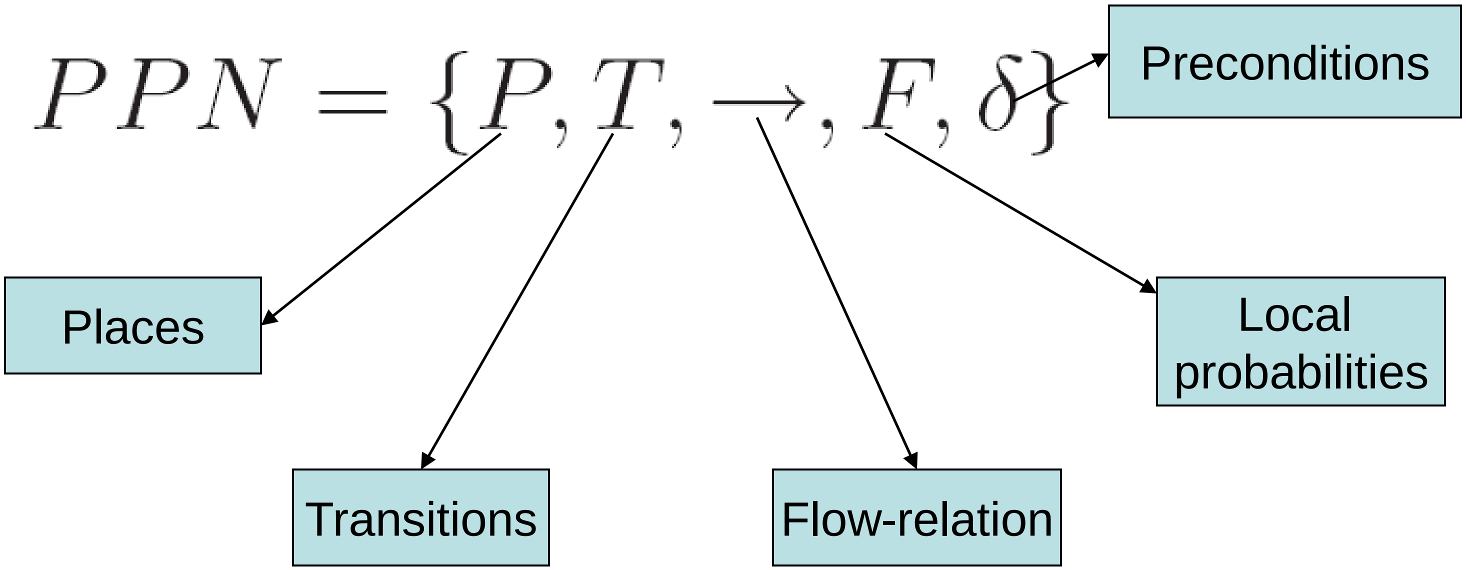


Anomaly detection: Results

Probabilistic Petri-nets

- PPN is a symbolic network: Rich and expressive models can be designed.
- Encode semantic rules of activity unfolding.
- Probabilistic extension allows for robust inference.
- M. Albanese, R. Chellappa, V. Moscato, A. Picariello, V.S. Subrahmanian, P. Turaga and O. Udrea, IEEE Trans. on Multimedia, Dec. 2008.

Probabilistic Petri-net as a tuple



$$PPN = \{P, T, \rightarrow, F, \delta\}$$

Preconditions

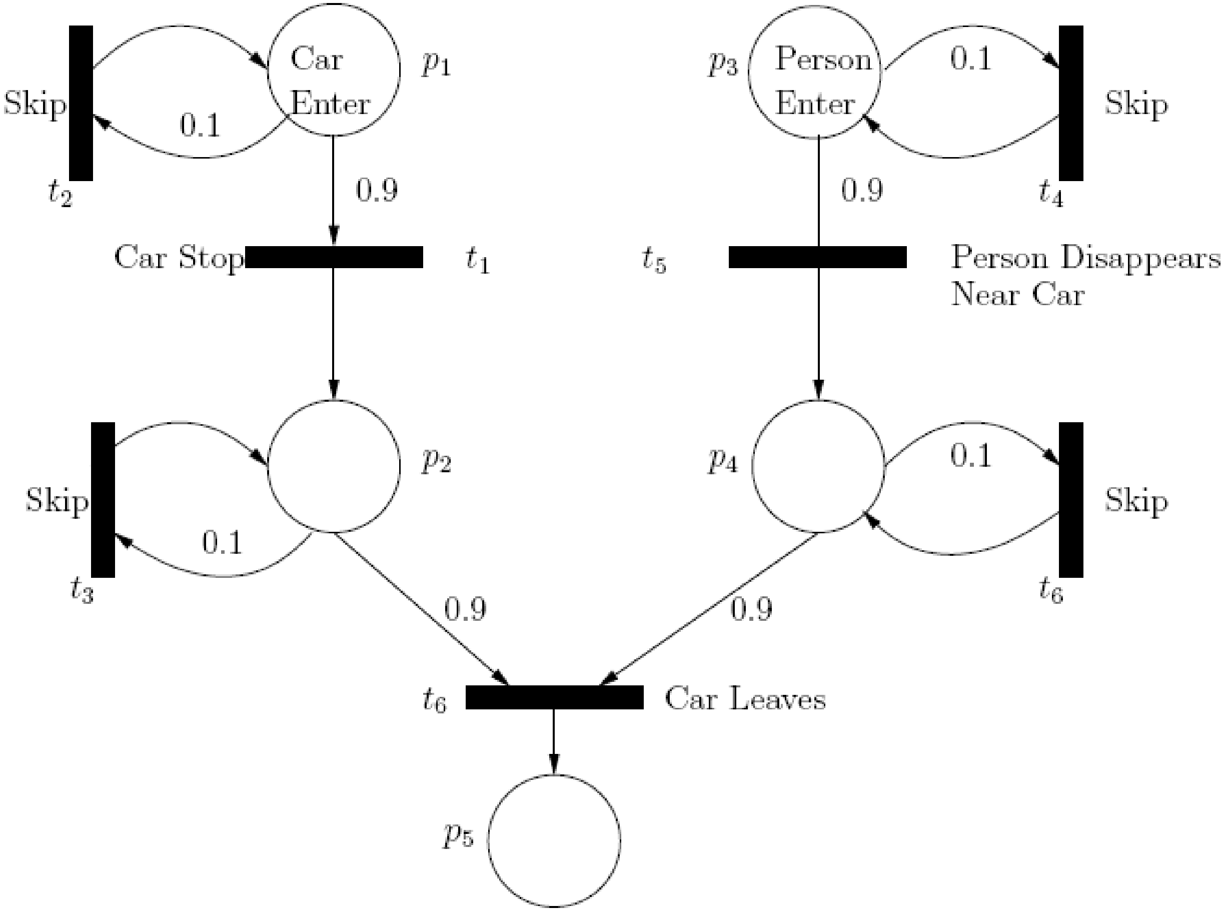
Places

Local probabilities

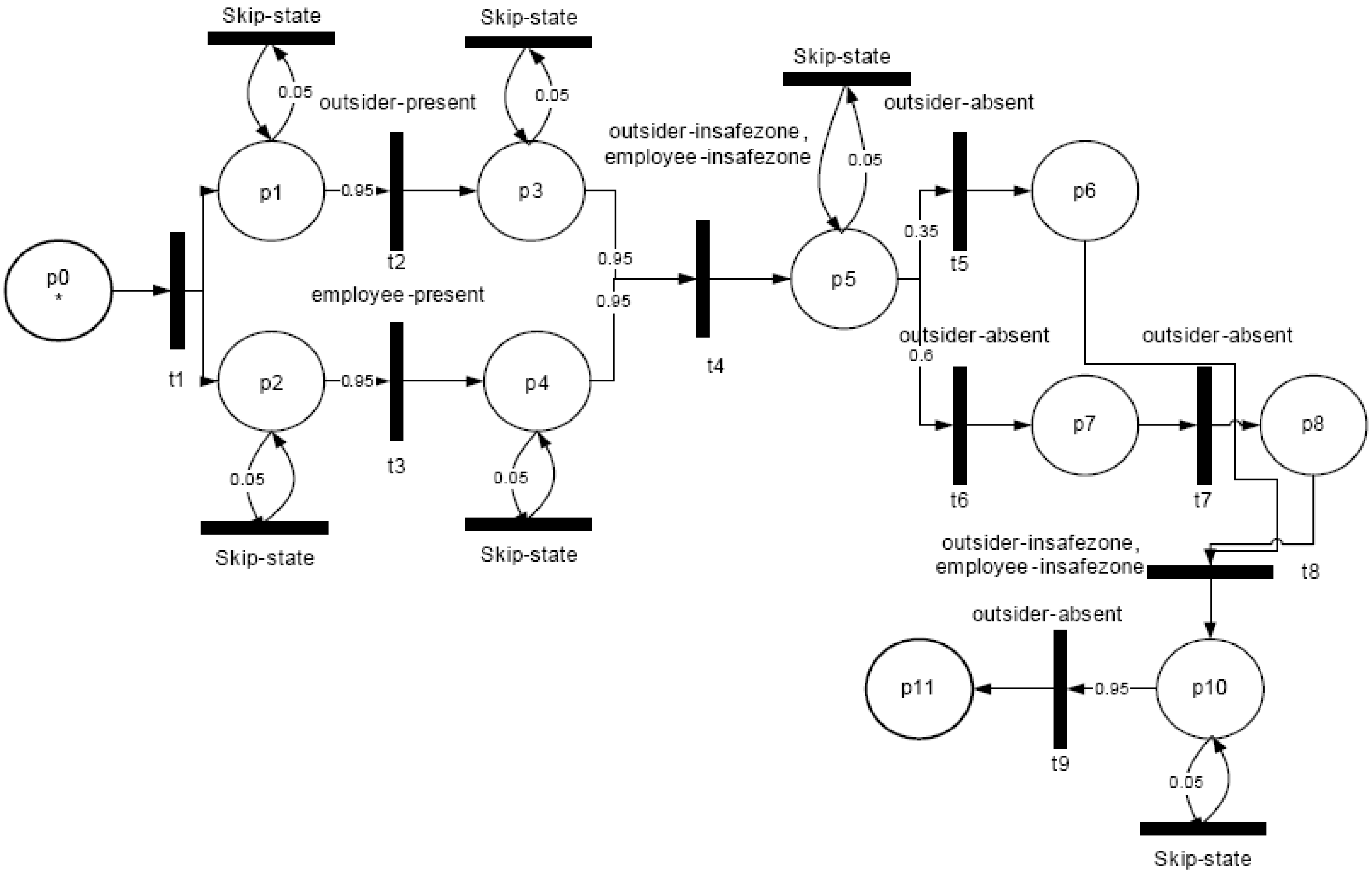
Transitions

Flow-relation

Example: Car pickup model



Bank robbery model



Petri-net parsing

- Parsing involves ‘firing’ enabled transitions
- Transitions are enabled if all its input places have a token and its associated pre-condition is satisfied.
- Need to take care of probability computations during firing.
- Parsing can be illustrated graphically using tokens.
- After firing, tokens are removed from input places and placed in output places.
- Two modes of operation
  - What are the minimal sub-videos in which a given

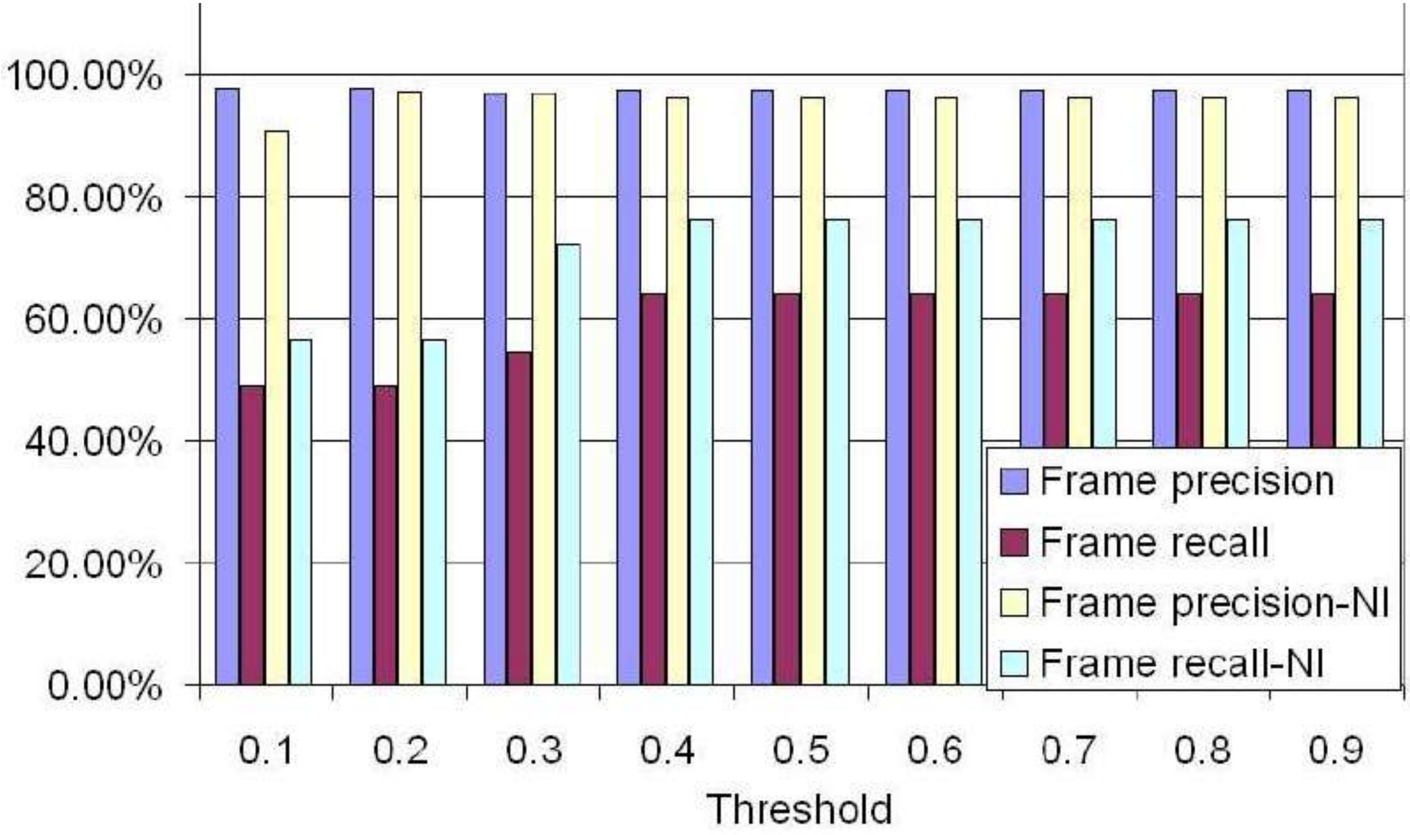
activity is identified with a probability above a certain threshold?

  - For a given video, which activity from a given set occurred with the highest probability

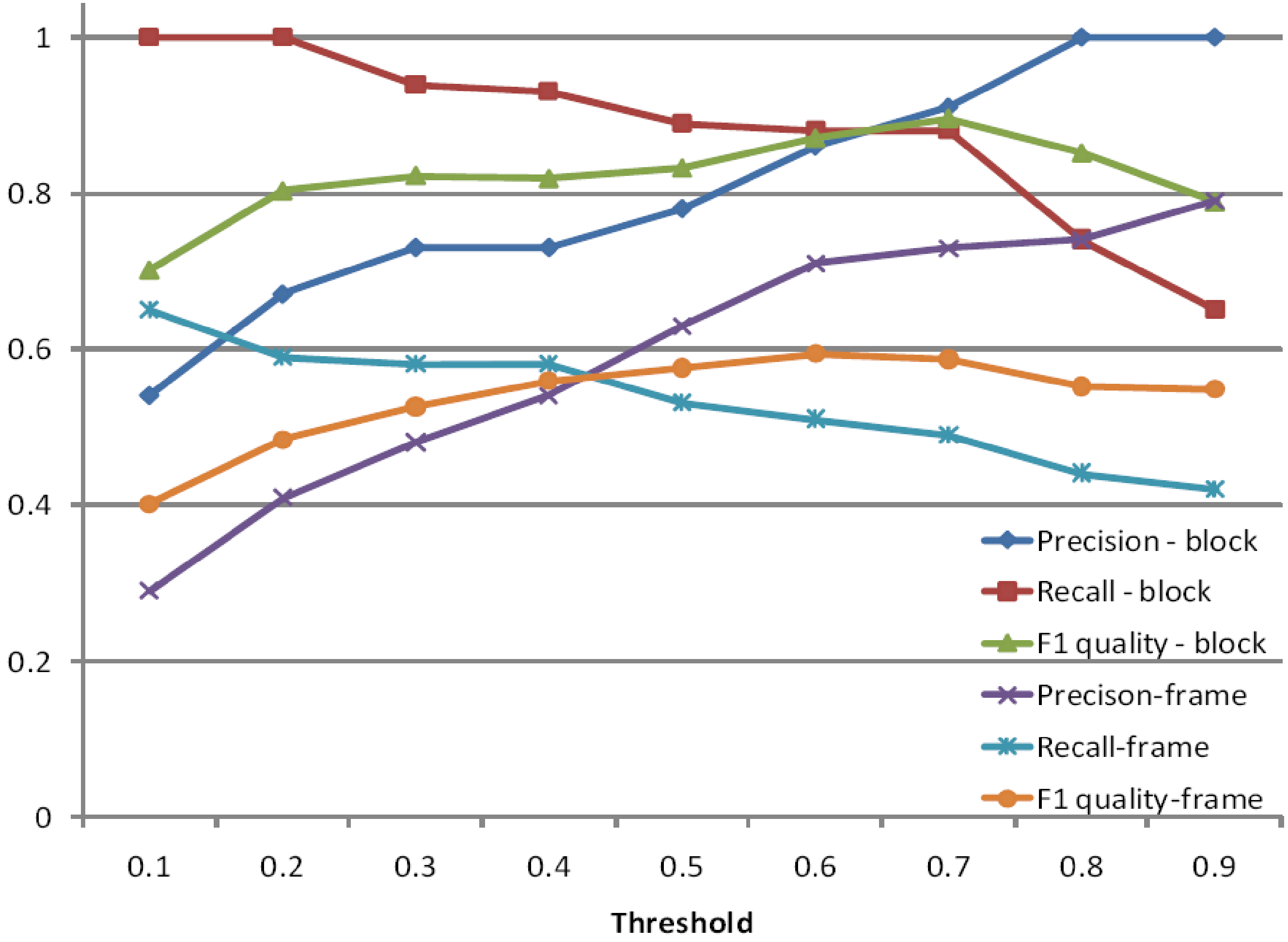
Activities considered

- a1) regular customer-bank employee interaction;
- a2) outsider enters the safe;
- a3) a bank robbery attempt—the suspected assailant does
- not make a getaway;
- a4) A successful bank robbery;
- a5) An employee accessing the safe on behalf of a customer.

Precision and recall for robbery detection in bank videos



Airport surveillance



PADS approach

- Using a first order logical formulas with frame variables to define activities of interest
  - PADL: Probabilistic Activity Description Language
- Allowing image processing algorithms to return probabilistic info about frame content (labeling)
- Defining the probability that the labeling of a subvideo satisfies an activity formula

M. Albanese, R. Chellappa, N. Cuntoor, V. Moscato,
A. Picariello, V.S. Subrahmanian, and O. Udrea,
IEEE Trans. PAMI, Dec. 2010.

PADL
Probabilistic activity description language

- PADL is a logical language, with a set of constant, function, and variable symbols, and a set of boolean predicate symbols
  - PADL also has a set of *probabilistic predicate symbols*
- Examples of boolean predicates
  - *motion(O, F)* is true if object *O* is moving in frame *F* of the video
  - *occl(O_1, O_2, F)* is true if object *O_1* is occluded by *O_2* or vice-versa in frame F
- Examples of probabilistic predicates
  - eq(*O_1*, *O_2*) returns the probability that *O_1* and *O_2* are the same object
- Activities are described in PADL as first order logic formulas (*activity formulas*)

Package transfer

- pwff$_1$

$$\exists(P, P', pkg, t_1, t_2)\ t_2 > t_1 \wedge \text{haspkg}(P, pkg, t_1) \wedge$$
$$\text{in}(P, t_1) \wedge \text{haspkg}(P', pkg, t_2) \wedge \text{in}(P', t_2).$$

- According to **pwff$_1$**a package exchange occurs when a person ***P*** has a package at some time ***t_1***, and a person ***P'*** has the package at a later time ***t_2***
- **pwff$_1$** represents the simplest possible definition of package transfer
  - 10 different ways to characterize a package transfer!
  - More detailed descriptions can be provided using PADL
  - See PAMI paper for more details on activity recognition algorithms

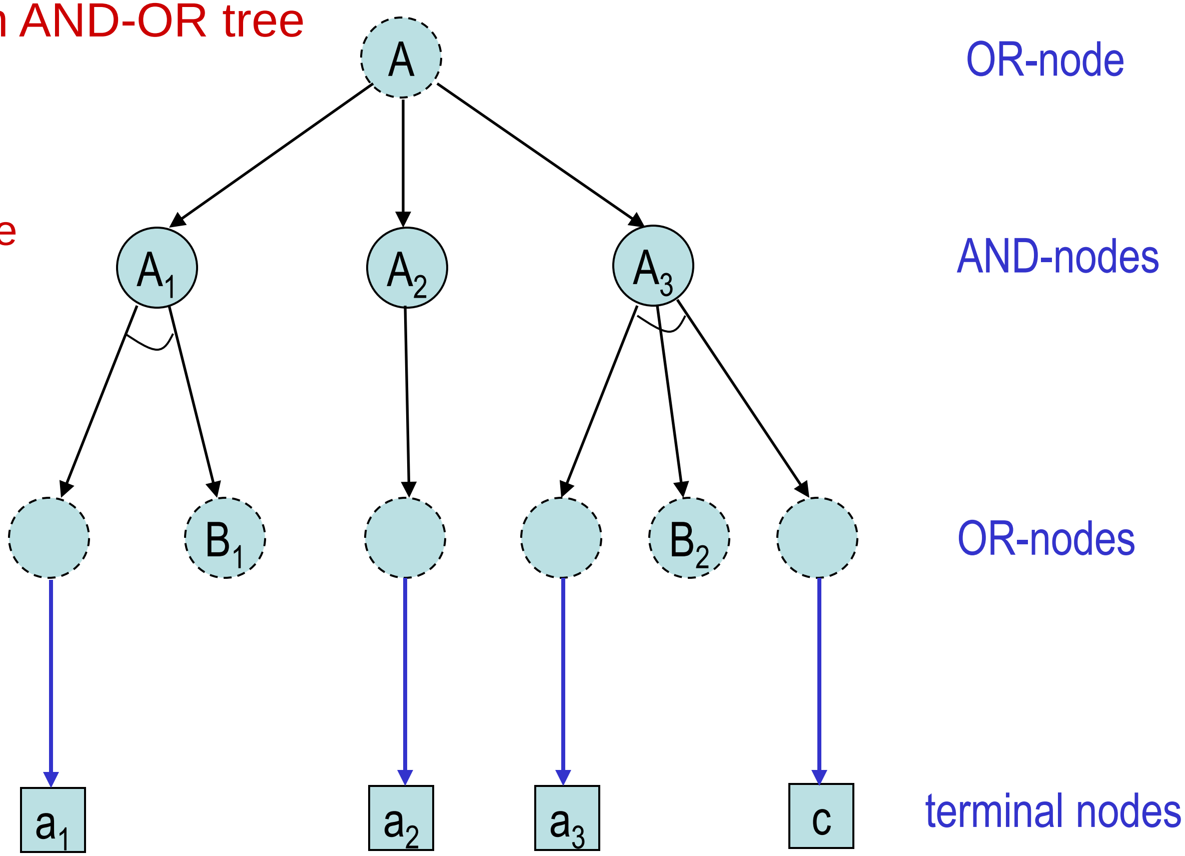
Embodying SCFG grammars in AND-OR Trees

In a grammar, each non-terminal node has a number of alternative ways for expanding, and thus
can be represented by an AND-OR tree

A ::= aB | a | aBc

When the grammar does not have infinite recursion, then the whole grammar can always be represented by an AND-OR tree

From Zhu



AND-OR graph

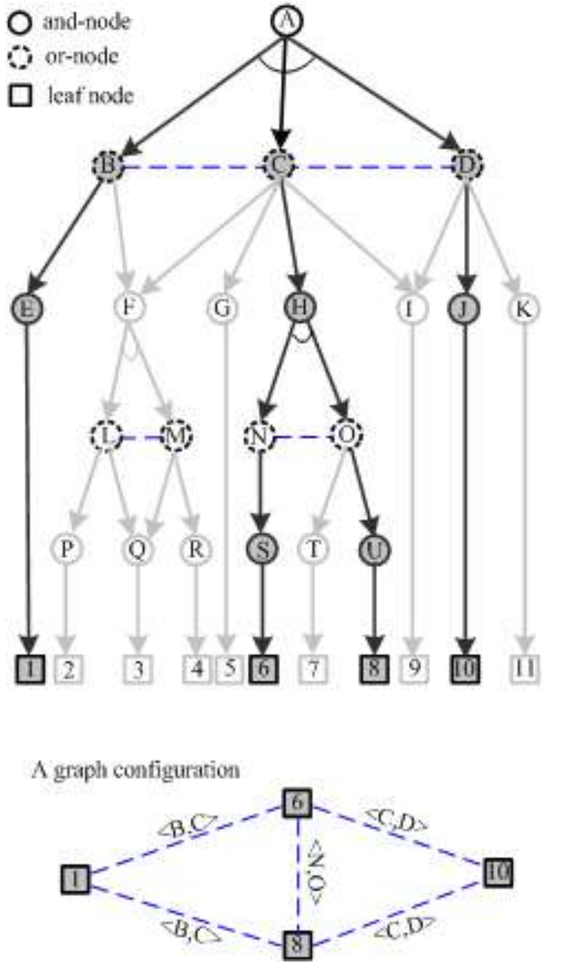
AND-OR graph represents the whole “grammar”.

A specific *parsing graph* models the hierarchic structure of a specific object instance.

A *configuration* is a flat graph generated by the parsing graph

The AND-OR graph was used in heuristic AI search (Pearl, 1984).
Like the 12-counterfeit coin problem.

It was not used for modeling, but for problem solving in a divide-and-conquer strategy. Pearl didn’t use the horizontal links for context.



From Zhu

Chen, Xu, Liu, and Zhu, CVPR, 2005

Defining Probability model on the AND-OR graph

Denote:

G ---- a parse graph,

U(G) ---- the set of OR-nodes in G,

V(G) ---- the set of the And-nodes + leaf nodes in G

R(G) ---- the set of relational links between nodes in G.

The probability model is defined as

$$p(G;\Delta, R, \theta) = \frac{1}{Z}\exp\{-\sum_{u \in U(G)} \lambda(u) - \sum_{v \in V(G)} \varphi(v) - \sum_{r_{ij} \in R(G)} \psi(r_{ij})\}$$

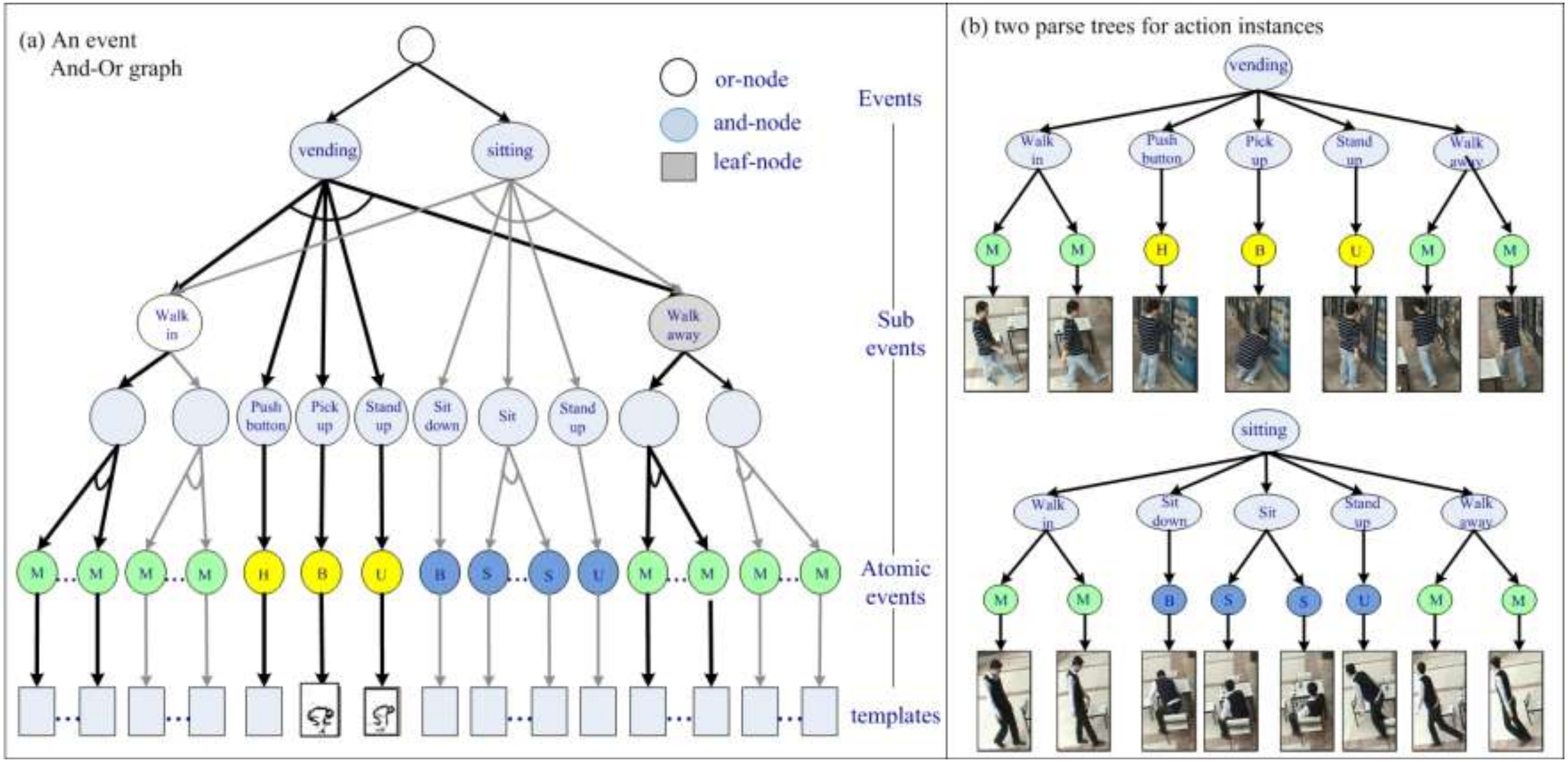
From Zhu

The first term alone stands for a SCFG.

The second and third terms are Markov potentials.

For a context sensitive attribute grammar: the hard relations / constraints affect the frequency at the or-nodes.

Temporal AoG for action / events



Ref. M. Pei and S.C. Zhu, “Parsing Video Events with Goal inference and Intent Prediction,” ICCV, 2011.

Some observations

- Activity recognition is an inference problem
  - Models, inputs (features) and inference algorithm
- Stochastic grammars have potential but are hard to transition.
- Needs to be validated on large VACE, Mind’s Eye, ALADDIN data sets.
- Should develop methods that can incorporate limited view and rate invariances.
- Develop performance bounds on recognition performance.
- More importantly, populate OpenCV with free codes and software!