## Grammars (C2)

<table>
<thead>
<tr>
<th>Grammar</th>
<th>Languages</th>
<th>Automaton</th>
<th>Production rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type-0</td>
<td>Recursively enumerable</td>
<td>Turing machine</td>
<td>$\alpha \rightarrow \beta$ (no restrictions)</td>
</tr>
<tr>
<td>Type-1</td>
<td>Context-sensitive</td>
<td>Linear-bounded non-deterministic Turing machine</td>
<td>$\alpha A\beta \rightarrow \alpha \gamma \beta$</td>
</tr>
<tr>
<td>Type-2</td>
<td>Context-free</td>
<td>Non-deterministic pushdown automaton</td>
<td>$A \rightarrow \gamma$</td>
</tr>
<tr>
<td>Type-3</td>
<td>Regular</td>
<td>Finite state automaton</td>
<td>$A \rightarrow a$ and $A \rightarrow aB$</td>
</tr>
</tbody>
</table>

- Stochastic Grammars (2.4) (Bayes Net w/ dynamic structure)
- Polytime for Types 2-3*
- Wikipedia says most natural languages generated by Type-1
- Intermediate between 1 and 2: Tree-adjoining/Attribute Grammars
- Type-0 includes innate universal grammar shared by all humans*
Why Grammars?

- Composition and Reusability
- Productivity

(fewer training examples, represent large intra-class variation)
Image Grammar Challenges (2.3 + 2.6)

Ambiguity (2.3)

No LR Order

Continuous Image Scaling

Continuous Spectrum of Texture/Clutter

images  sketches  primitives
Previous Work on Image Grammars (2.7)

- Syntactic Pattern Recognition
- Medial Axis, Shock Graphs
- Pattern Theory
- Sparse Coding
And-Or Graph (6.2)

- And-Or Tree (PCFG):
  - (Leaf Node) := a
  - (Or Node) := A (label)
  - (And Node) := γ (template)

- And-Or Graph (Eq 54-55):
  - Horizontal Or Edges (pot.)
  - *Other Relations
  - *Sharing Nodes
Learning (C7)

- 1. No horizontal edges, standard PCFG learning
- 2. Introduce edge w/ highest information gain (67)
- 3. Sample to approximate expectations (63,65)
- 4. Move down gradient

Repeat 2-4
Parsing (decoding)

- Recursive top-down / bottom-up algorithm:
  - Bottom-up:
    - Feature Detection
    - Composition Binding
  - Top-down
    - Compute Hypothesis Posterior
    - Update Hypothesis
• Vocabulary (C3):
  • Image Primitives
  • Basic Geometric Groupings
  • Parts and Objects

• Relations (C4):
  • Connections between primitives
  • Joints and Junctions
  • Object Interactions and Semantics
Firstly, to represent intra-category variation, the grammar can create a large number of configurations from a relatively much smaller vocabulary. The Andor graph acts like a reconfigurable template and assembles novel configurations on the fly to interpret novel instances unseen before.

Secondly, to scale up to hundreds of categories, the Andor graph is recursively designed. Thus, one can integrate without much overhead all categories into one big Andor graph. The learning and inference algorithms are designed recursively as well. This permits large-scale parallel computing.

There are two open issues for further study: learning and dis-covering the Andor graph. As it was proposed in a series of recent works [87; 52; 58; 87], the objective is to map the visual vocabulary including dictionaries at all levels of abstraction and all visual aspects. This task can be formulated in theory under a common learning principle: to put the dictionary $\Delta$ into the maximum likelihood learning process. The various information criteria such as the binding strength, mutual information, minimax entropy will come naturally out of this learning process. However, the ultimate visual vocabulary is unlikely to be learned fully automatically from statistical principles as the determination of the vocabulary must take the purposes of vision into account. This argues for a semi-automatic method which is being carried out at the Lotus Hill Institute: human users guided by real-life experiences in psychology and vision tasks define most of the structures and leaving the

*learn graph structure, learn vocabulary, share re-usable parts
Test image

Top objects

Top object under Markov distribution

Top object under content-sensitive distribution