A Stochastic Grammar of Images

A Draft

Barun Das
Roll number: 16MT10012

1 Introduction

The main aim of the paper was to develop a stochastic and context sensitive grammar of images. Grammars are mostly used in languages and are useful because we can generate a large set of configurations/instances (language) using a relatively smaller set of words using certain production rules. A similar tool was desired for images wherein we could generate several images using just a small set of primitives. The proposal grammar integrates three prominent representations in the literature: stochastic grammars for composition, Markov (or graphical) models for contexts, and sparse coding with primitives (wavelets). It also combines the structure-based and appearance based methods in the vision literature.

2 Objectives

- A common framework for visual knowledge representation and object categorization.
- Scalable and recursive top-down/bottom-up computation.
- Small sample learning and generalization.
- Mapping the visual vocabulary to fill the semantic gap.

3 The methodology in brief

Very simply put, the methodology employed involves creating a visual vocabulary of parts, objects and image primitives. These are represented as an And-Or graph (an And-Or tree augmented with horizontal edges representing contextual constraints). Here, the And nodes represent the decomposition of an object into its parts while an Or node acts as a switch between possible configurations.
The stochastic information is incorporated in two ways:

- Local probabilities at each Or node to account for the relative frequency of each alternative.
- Local energies associated with each horizontal link.

The And-Or graph is a useful representation framework for the vast amount of visual knowledge at all levels of abstraction because it allows reusability of common parts and thus can represent several different instances as well as intra-class variation. It is similar to a class in C++. A parse graph is an interpretation of a specific image. And-Or graphs contain all valid parse graphs and thus embed the whole image grammar.

3.1 Learning and Estimation with And-Or graphs

The learning algorithm starts with a SCFG (Markov tree) and a number of observed parse graphs for training D^{obs}. It first learns the SCFG model by counting the occurrence frequency at the Or-nodes. Then, by sampling this SCFG, it synthesizes a set of instances D^{syn}. The sampled instances in D^{syn} will have the proper components but often have wrong spatial relations among the parts as there are no relations specifed in SCFG. The algorithm chooses a relation that has the most different statistics (histogram) over some measurement between the sets D^{obs} and D^{syn}. The model is then learned to reproduce the observed statistics over the chosen relation. A new set of synthesized instances is sampled. This iterative process continues until no more significant differences are observed between the observed and synthesized sets.

3.2 Recursive top-down/bottom-up algorithm for Image Parsing

The learning algorithm is based on a bottom-up approach whereas for image parsing we use a top-down approach. We start with an image (terminal configuration) and an And-Or graph as inputs. This section briefly describes an inference algorithm for image parsing using the image grammar that has been developed.
Basically, for any generic And node A in the graph, there are two kinds of lists maintained:

- An **Open List** which stores the number of weighted particles (hypotheses) computed in the bottom-up process for the instances of A in the input image.
- A **Closed List** which stores instances of A accepted in the top-down process. These are nodes in the current parse graph.

There are two basic processes which maintain and compute both lists for each unit A

- The bottom-up process creates the particles in the Open lists.
- The top-down process validates the bottom-up hypotheses in all the Open lists, following the Bayesian posterior probability. It also needs to maintain the weights of the Open lists.

4 Current Issues to work on and possible applications

In this section, I outline the issues in the current model as well as areas where this model may be implemented to improve status quo.

1. A key issue is *scheduling* the top-down and bottom-up processes mentioned in the inference algorithm in the previous section. For some patterns, like human faces, it is more convenient to first detect the whole face and then locate its components, whereas in other cases, it is more effective to work bottom-up (detect components to infer the image). The optimal way to achieve this scheduling has been a long standing problem. Currently, greedy algorithms, data driven Markov chain Monte Carlo based algorithms and feed-forward neural networks are all being used for this problem, each with varying results and issues.
2. In the learning algorithm mentioned in the previous section, there are three phases involved in learning the probability model. Discovering and binding the vocabulary to the heirarchic And-Or tree automatically is a phase where no significant work has been done (although the authors state that all the three phases follow the same principle).
3. These graphs may be employed in systems where the computer has to make decisions based on recognition, such as autonomous vehicles or defence systems. Since And-Or graphs can be used as a template to not only identify a large set of intra- as well as inter-class objects, but can also explain them to the last pixel, it becomes simpler to understand the actions as predicted by the computer. This is especially useful in autonomous vehicles and defence systems where actions can have huge consequences.
4. The main objective is to map the visual vocabulary at all levels of abstraction, including dictionaries. Currently, this is done in a semi-automatic method, where human users guided by real life experience, psychology and vision tasks, define most of the structures, leaving the estimation of parameters and adaptation to computers. It would be useful if this process could be automated (it can theoretically be done by a common learning algorithm but without human intervention it would be difficult to account for the purpose of vision), or at the very least, if computers could find and pursue the addition of novel elements to their dictionaries.

5. We could attempt to build a system where visual structure may be learned from a purely textual information set, very much similar to how humans imagine things when they hear a description.

6. This process could be extended to other forms of media too, with video (non-static images) being the obvious next choice, but we could also try this in audio. We would need to define an adaptive stochastic And-Or graph to deal with this.