Hierarchical Cascade of Classifiers for Efficient Poselet Evaluation

Bo Chen[1]
bchen3@caltech.edu
Pietro Perona[1]
perona@caltech.edu
Lubomir Bourdev[2]
lubomir@fb.com

[1] Computation and Neural Systems
California Institute of Technology
California, USA

[2] Facebook AI Research,
Menlo Park, California, USA

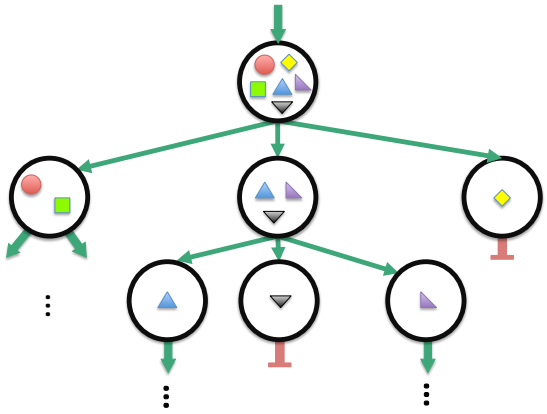

Figure 1: Cascade hierarchy. Each node is a classifier trained to let through examples of a set of parts (represented as shapes within the node) while filtering out the background class. The set of parts at the node is partitioned and each child is responsible for a subset of them. When an example passes a node classifier, it is evaluated by all of its children.

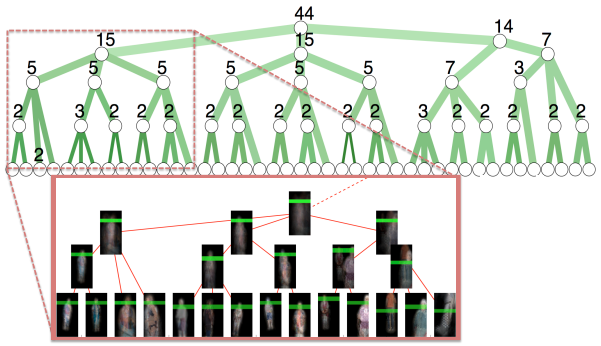

Figure 2: A classification tree generated by our algorithm for classifying 44 poselets at 90% target detection rate. The thickness of the edges denotes the retention rate of the classifiers. The number of poselet types classified by each node is indicated. **Left corner:** A zoom on part of the tree. At each node we show the average mask over all classifiers captured by the node, along with the horizontal stripe that was used to classify the node.

Poselets [1] have been used in a variety of computer vision tasks, such as detection, segmentation, action classification, pose estimation and action recognition, often achieving state-of-the-art performance. Poselets are part classifiers trained to detect part of a human pose under a given viewpoint. Examples of poselet classifiers are a frontal face, a part of a face and left shoulder, or a hand next to a hip in a side view. Poselet evaluation, however, is computationally intensive as it involves running thousands of scanning window classifiers to detect hundreds of poselet types. We present an algorithm for training a hierarchical cascade of part-based detectors and apply it to speed up poselet evaluation. Our cascade hierarchy leverages common components shared across poselets. We generate a family of cascade hierarchies, including trees that grow logarithmically on the number of poselet classifiers.

Example of our cascade and evaluation algorithm is shown on Figure 1. At each node we train a classifier designed to distinguish between a subset of the parts and the background class. We compute two values at the node: the detection rate (the fraction of positive examples that the node classifier passes) and the retention rate (the fraction of examples the node classifier passes). The detection rate of the cascade is the product of detection rates of the chain of nodes from the root to the leaves, and the computational cost is inversely proportional to the retention rate. Our algorithm finds the cascade structure that minimizes the computational cost while preserving a given target detection rate. Since the space of all possible trees is intractably large, we restrict it using a few simplifying assumptions: (1) the detection rate tradeoff between a node and its children is the same throughout the tree, (2) the number of children is no larger than 4, and (3) the partitioning of a set of parts into K subsets (one for each child) is fixed using a clustering algorithm.

While our algorithm is generic, in the case of HOG-based poselets, our node classifiers are linear SVMs over the HOG features in a horizontal stripe of the input image patch. We pick the stripe that best separates the node parts from the background class. Our design choices allow for efficient and memory-cache friendly classifier.

We use a dynamic programming approach to find the optimal cascade structure in this restricted state space. An example of the classification cascade is shown on Figure 2. We test our system on the PASCAL dataset [2] and show an order of magnitude speedup at less than 1% loss in AP (Figure 3-left). We also show that our algorithm evaluation cost scales logarithmically with the number of poselet classifiers (Figure 3-right).

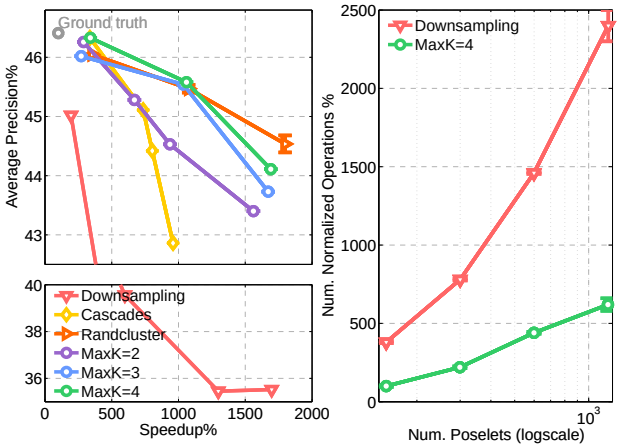

Figure 3: **Left:** Average precision of our classifier (*MaxK*=4) on the PASCAL 2007 set for the Person category as a function of evaluation speed. We compare against the AP of *Downsampling*: standard poselet detector with coarser sampling in space and scale; *Cascades*: independent cascades for each individual poselet; *Randcluster*: cascade hierarchy with random partition and *MaxK=k*: cascade hierarchy where each node can have at most k children. **Right:** Computation time for the same detection rate as a function of the number of poselets.

[1] Lubomir Bourdev and Jitendra Malik. Poselets: Body part detectors trained using 3D human pose annotations. In *ICCV*, 2009.

[2] Mark Everingham, Luc Van Gool, Chris K. I. Williams, John Winn, and Andrew Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results, 2007.