

# Object Detection on a Budget

Sergey Karayev

19 October 2011

## Abstract

Notes on the POMDP-like approach to multi-class object detection that I am currently coding up.

## 1 Our Approach

### 1.1 Assumptions and Notations

We have a dataset of images  $d \in D$ . Each image contains at least one object with class  $k \in K$ . “Background” is not considered to be a class.

For each class  $k$ , we have  $|C_k|$  classifiers  $c_{k,i}$ , with  $1 \leq i \leq |C_k|$ .  $|C_k|$  must be at least 1, but there may be more than one classifier per class—for example, a slow but powerful and a fast but weak classifier for each class.

The task of object detection is defined as evaluating windows (bounding boxes) in the image with these classifiers. This is a common formulation; if the window proposals are exhaustive and the classifiers are perfect, then all objects in the image should be perfectly localized. An important aside is the role of post-processing of detections. Because a single ground truth bounding box will give rise to multiple correct detections (the overlap requirement is not 1), but only one detection can be counted as correct for a given bounding box according to the PASCAL evaluation, detections are usually reduced with non-maximum suppression, which finds the bounding box with the highest confidence. The score of a classifier should then correspond to the amount of overlap with the ground truth, not only to the binary presence decision.

Following [1], we parameterize a window  $w$  by its center  $(x, y)$ , scale  $s$ , and aspect ratio  $r$ . While some classifiers are able to handle a window of any dimensions, some expect to evaluate windows of a certain aspect ratio, and may be sensitive to the scale. Accordingly, the bounds on  $s$  and  $r$  depend on the model of the classifier. The set of possible window proposals for an image of size  $n \times m$ , according to the model of classifier  $c_{k,i}$  is therefore defined by  $W_{k,i} = \{x, y, s, r | x \in [0, m], y \in [0, n], s \in [\underline{s}_{k,i}, \bar{s}_{k,i}], r \in [\underline{r}_{k,i}, \bar{r}_{k,i}]\}$ .

This notation is also summarized in Table 1.

In a nutshell, our approach consists of

- Modeling the classifier toward the goal of inferring the object posterior from classifier scores.
- Maintaining prior distributions over the object class presence in a region.

Table 1: Summary of our notation.

Term	Definition
$D$	Set of images $d$ composing the dataset.
$K$	Set of object classes $k$ present in the dataset.
$C$	Set of classifiers $c_{k,i}$ , where $k \in K$ and $1 \leq i \leq  C_k $ allows the possibility of more than one classifier per class.
$ C_k $	Number of classifiers for class $k$ .

- Updating the priors for all classes after inferring the posterior of one class by running a detector.

## 1.2 Modeling the Classifier

A classifier  $C_k$  takes an image window  $w$  and answers the question, “is the object of class  $k$  present in window  $w$ ?” For shorthand, we define  $w_K$  to be the class of object present in window  $w$ .

We model  $C_k$  with two distributions over its confidence:  $P(\text{score}|w_K = k)$  and  $P(\text{score}|w_K \neq k)$ . We model the score likelihood distributions with, for example, the Beta distribution, and learn the parameters from data.

The score (confidence) of the classifier is supposed to represent the posterior  $P(w_K = k|\Psi(w), C_k)$ , where  $\Psi(w)$  represents the features extracted from  $w$ , according to the model of  $C_k$ . But of course, the score is affected by lossy signal and limited by the modeling capacity of  $C_k$ . A classifier that is not able to model the object class or is overrun by noise will have a uniform distribution. A perfect classifier will have two point distributions at the 1 and 0 values.

To infer the  $P(w_K = k)$  from the classifier score, we can write:

$$P(w_K = k|\text{score}) = \frac{P(\text{score}|w_K = k)P(w_K = k)}{P(\text{score}|w_K = k)P(w_K = k) + P(\text{score}|w_K \neq k)P(w_K \neq k)}$$

In this way, the score of a worthless classifier will not play a role in determining the object posterior, and the score of a perfect classifier will completely override the role of the prior.

## 1.3 Detectors, not Classifiers

Up until now, we have ignored the fact that our classifiers are run not just on a single window but on a set of windows in the image region. Accordingly, we obtain a *list* of detections instead of a single score after running  $C_k$  in an image region.

Our class priors are defined as the probability of an object of the given class being present in any given window in the image region. Therefore, the priors are sensitive to the size of the region relative to the size of the image, and to the resolution of the image.

**there is something here about the gradient from classification to detection, but i'm not sure what it is... need to think more**

## 1.4 Window Discretization

We follow [2] in our window discretization scheme, which we summarize here. We define the overlap between window  $t \in T$ , the set of all possible windows in the image, and window  $s \in S$ , the set of all windows with width  $w$  and height  $h$  according to the PASCAL overlap criterion:

$$o(t, s) = \frac{t \cap s}{t \cup s} \quad (1)$$

We say that  $s$  is represented to  $\eta$ -accuracy if  $o(t, s) \leq \eta$ .

## 1.5 Object Co-Occurrence

The object posterior that we inferred from running the detector is information that we should take into account when picking the next action. We can certainly replace the current prior for this class with the posterior, but we also want to influence the priors on the other classes.

We model the object class cooccurrence with a fully connected field of  $K$  random variables  $P(w_K = k)$ , one per class. The variables

## References

- [1] A. D. Lehmann, P. V. Gehler, L. Van Gool, and L. V. Gool, “Branch Rank: Non-Linear Object Detection,” in *BMVC*, 2011. [1](#)
- [2] Z. Zhang, J. Warrell, and P. H. S. Torr, “Proposal Generation for Object Detection using Cascaded Ranking SVMs,” in *CVPR*, 2011. [2](#)