How important are "Deformable Parts" in the Deformable Parts Model?

Santosh K. Divvala, Alexei A. Efros, and Martial Hebert

Robotics Institute, Carnegie Mellon University.

Abstract. The Deformable Parts Model (DPM) has recently emerged as a very useful and popular tool for tackling the intra-category diversity problem in object detection. In this paper, we summarize the key insights from our empirical analysis of the important elements constituting this detector. More specifically, we study the relationship between the role of deformable parts and the mixture model components within this detector, and understand their relative importance. First, we find that by increasing the number of components, and switching the initialization step from their aspect-ratio, left-right flipping heuristics to appearance-based clustering, considerable improvement in performance is obtained. But more intriguingly, we observed that with these new components, the part deformations can now be turned off, yet obtaining results that are almost on par with the original DPM detector.

1 Introduction

Consider the images of category horse in Figure 1 (row1) from the challenging PASCAL VOC dataset [1]. Notice the huge variation in the appearance, shape, pose and camera viewpoint of the different horse instances – there are left and right-facing horses, horses jumping over a fence in different directions, horses carrying people in different orientations, close-up shots, etc. How can we build a high-performing sliding-window detector that can accommodate the rich diversity amongst the horse instances?

Deformable Parts Models (DPM) have recently emerged as a useful and popular tool for tackling this challenge. The recent success of the DPM detector of Felzenszwalb et al., [2] has drawn attention from the entire vision community towards this tool, and subsequently it has become an integral component of many classification, segmentation, person layout and action recognition tasks (thus receiving the lifetime achievement award at the PASCAL VOC challenge).

Why does the DPM detector [2] perform so well? As the name implies, the main stated contribution of [2] over the HOG detector described in [3] is the idea of deformable parts. Their secondary contribution is latent discriminative learning. Tertiary is the idea of multiple components (subcategories). The idea behind deformable parts is to represent an object model using a lower-resolution 'root' template, and a set of spatially flexible high-resolution 'part' templates. Each part captures local appearance properties of an object, and the deformations are characterized by links connecting them. Latent discriminative learning

involves an iterative procedure that alternates the parameter estimation step between the known variables (e.g., bounding box location of instances) and the unknown i.e., *latent* variables (e.g., object part locations, instance-component membership). Finally, the idea of subcategories is to segregate object instances into disjoint groups each with a simple (possibly semantically interpretable) theme e.g., frontal vs profile view, or sitting vs standard person, etc, and then learning a separate model per group.

A common belief in the vision community is that the deformable parts is the most critical contribution, then latent discriminative learning, and then subcategories. Although the ordering somewhat reflects the technical novelty (interestingness) of the corresponding tools and the algorithms involved, it is interesting to check whether is that really the order of importance affecting the performance of the detector in practice.

In this paper, we empirically analyze the relative importance of deformable parts and subcategories within the DPM detector. First, we find that (i) by increasing the number of subcategories in the mixture model, and (ii) switching from their aspect-ratio, left-right flipping heuristics to appearance-based clustering, considerable improvement in performance is obtained. But more intriguingly, we observed that with these new subcategories, the part deformations can be turned off, with only minimal performance loss. These observations reveal that the conceptually simpler notion of subcategories is indeed an equally important contribution in the DPM detector. Their careful use can potentially alleviate the need for deformable parts in the DPM detector for many practical applications and object classes.

2 Understanding Subcategories

In order to deal with significant appearance variations that cannot be tackled by the deformable parts, [2] introduced the notion of multiple components i.e., subcategories [4, 5, 6, 7, 8] into their detector. The first version of their detector [9] only had a single subcategory. The next version [2] had two subcategories that were obtained by splitting the object instances based on aspect ratio heuristic. In the latest version [10], this number was increased to three, with each subcategory comprising of two bilaterally asymmetric i.e., left-right flipped models (effectively resulting in 6 subcategories). The introduction of each additional subcategory has resulted in significant performance gains (e.g., see slide 23 in [11]).

Given this observation, what happens if we further increase the number of subcategories in their model? In Section 4, we will see that this does not translate to improvement in performance. This is because the aspect-ratio heuristic does not generalize well to a large number of subcategories, and thus fails to provide a good initialization. Nonetheless, it is possible to explore other ways to generate subcategories. For example, subcategories for cars can be based either on object pose (e.g., left-facing, right-facing, frontal), or car manufacturer (e.g., Subaru, Ford, Toyota), or some functional attribute (e.g., sports car, utility ve-

hicle, limousine). Figure 1 illustrates a few popular subcategorization schemes for horses.

What is it that the different partitioning schemes are trying to achieve? A closer look at the figures reveals that they are trying to encode the homogeneity in appearance. It is the *visual homogeneity* of instances within each subcategory that simplifies the learning problem leading to better-performing classifiers (Figure. 2). What this suggests is, instead of using semantics or empirical heuristics, one could directly use appearance-based clustering for generating the subcategories. We use this insight to define new subcategories in the DPM detector, and refer to them as *visual* subcategories (in contrast to semantic subcategories that involve either human annotations or object-specific heuristics).

3 Learning Subcategories

We first briefly review the key details of using subcategories in the DPM detector, and then explain the details specific to their use in our analysis.

Given a set of n labeled instances (e.g., object bounding boxes) $D = (< x_1, y_1 >, \ldots, < x_n, y_n >)$, with $y_i \in \{-1, 1\}$, the goal is to learn a set of K subcategory classifiers to separate the positive instances from the negative instances, wherein each individual classifier is trained on different subsets of the training data. The assignment of instances to subcategories is modeled as a latent variable z. This binary classification task is formulated as the following (latent SVM) optimization problem that minimizes the trade-off between the l_2 regularization term and the hinge loss on the training data [2]:

$$\arg\min_{w} \frac{1}{2} \sum_{k=1}^{K} ||w_k||^2 + C \sum_{i=1}^{n} \epsilon_i, \tag{1}$$

$$y_i . s_i^{z_i} \geqslant 1 - \epsilon_i, \ \epsilon_i \geqslant 0, \tag{2}$$

$$z_i = \arg\max_{k} s_i^k, \tag{3}$$

$$s_i^k = w_k . \phi_k(x_i) + b_k. \tag{4}$$

The parameter C controls the relative weight of the hinge-loss term, w_k denotes the separating hyperplane for the kth subclass, and $\phi_k(.)$ indicates the corresponding feature representation. Since the minimization is semi-convex, the model parameters w_k and the latent variable z are learned using an iterative approach [2].

Initialization As mentioned earlier, a key step for the success of latent subcategory approach is to generate a good initialization of the subcategories. Our initialization method is to warp all the positive instances to a common feature space $\phi(.)$, and to perform unsupervised clustering in that space. In our experiments, we found the Kmeans clustering algorithm using Euclidean distance function to provide a good initialization.

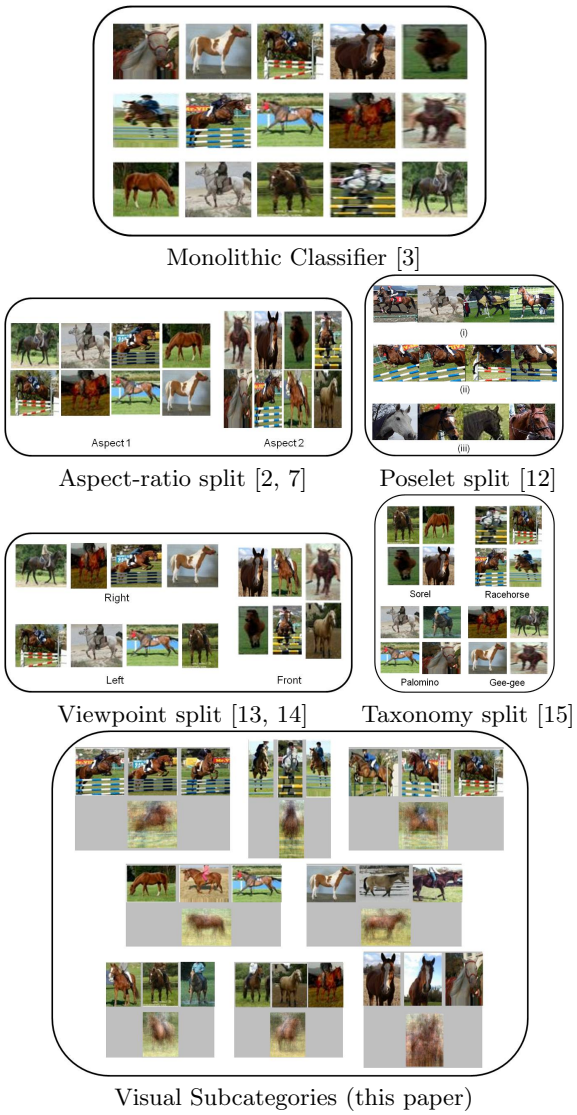
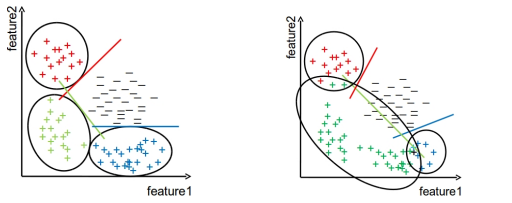


Fig. 1. The standard *monolithic classifier* is trained on all instances together. *Viewpoint split* partitions the data using viewpoint annotations into left, right, and frontal subcategories. *Poselets* clusters the instances based on keypoint annotations in the configuration space. *Taxonomy split* groups instances into subordinate categories using a human-defined semantic taxonomy. *Aspect-ratio split* uses a simple bounding box aspect-ratio heuristic. *Visual subcategories* are obtained using (unsupervised) appearance-based clustering (top: few examples, bottom: mean image)



Visual Subcategories Semantic Subcategories

Fig. 2. A single linear model cannot separate the data well into two classes. (Left) When similar instances (nearby in the feature space) are clustered into subcategories, good models can be learned per subcategory, which when combined together separate the two classes well. (Right) In contrast, a semantic clustering scheme (based on human annotations) also partitions the data but leads to subcategories that are not optimal for learning the category-level classifier.

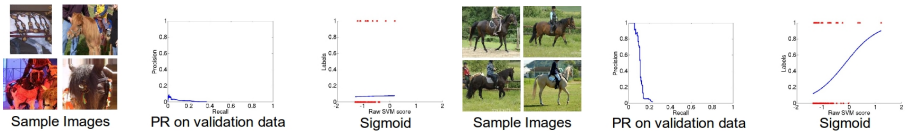
Calibration One difficulty in merging subcategory classifiers is to ensure that the scores output by individual SVM classifiers (learned with different data distributions) are calibrated appropriately, so as to suppress the influence of noisy ones. Note that, although the subcategory classifiers are coupled in the latent SVM formulation (1), a careful observation reveals that the classifiers are actually being learned independently. The coupling of classifiers only happens via the latent step (3) (i.e., the assignment of positive and negative instances to the different subcategories). Subsequently the SVM learning per subcategory is independent [2, 14].

We address this problem by transforming the output of each SVM classifier by a sigmoid to yield comparable score distributions [16][1](Figure 3). Given a thresholded output score s_i^k for instance i in subcategory k, its calibrated score is defined as

$$g_i^k = \frac{1}{1 + exp(A_k.s_i^k + B_k)}, \tag{5}$$

where A_k, B_k are the learned parameters of the logistic loss function $\arg\min_{A_k,B_k} \sum_{i=1}^{n} t_i \log g_i^k + (1 - t_i) \log(1 - g_i^k)$ with $t_i = Or(W_i^k, W_i)$, where $Or(w_1, w_2) = \frac{|w_1 \cap w_2|}{|w_1 \cup w_2|} \in [0, 1]$ indicates the overlap score between two bounding boxes [17], W_i is the ground-truth bounding box for the ith training sample, and W_i^k indicates the predicted bounding box by the kth subcategory. In our experiments, we found this calibration step to help improve the performance (mean A.P. increase of 0.5%).

[1] Even though the bias term b_k is used in equation (4) to make the scores of multiple classifiers comparable, we have found that it is possible for some of the subcategories to be very noisy (specifically when K is large and the subcategories have unequal distribution of positives across them), in which case their output scores cannot be compared directly with other, more reliable ones.



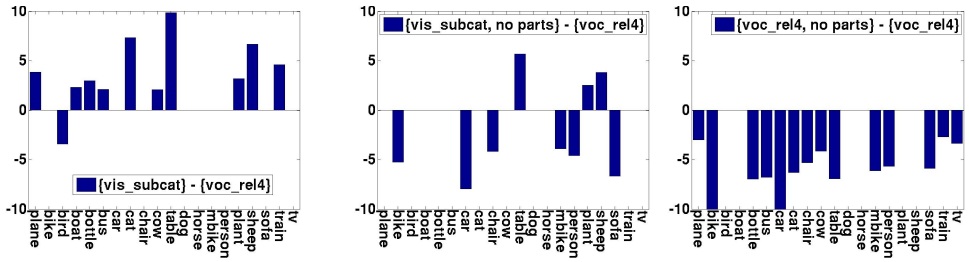
(a) ‘Noisy’ Subcategory (b) ‘Good’ Subcategory

Fig. 3. The classifier trained on a noisy subcategory (horses with extreme occlusion and confusing texture) performs poorly on the validation dataset. As a result, its influence is suppressed by the sigmoid. While a good subcategory (horses with homogeneous appearance) classifier leads to good performance on the validation data and hence its influence is boosted by the calibration step.

4 Experimental Analysis

We performed our analysis on the PASCAL VOC 2007 comp3 challenge dataset [1]. We used the standard PASCAL VOC comp3 test protocol, which measures detection performance by average precision (AP) over different recall levels. As our baseline system, we use the latest release of the DPM detector [10] (without the bounding-box prediction and context-rescoring steps). Figure 4 compares the results obtained using the different methods with respect to the baseline for the 20 PASCAL object categories.

The first sub-figure shows the improvements offered by using visual subcategories (with K=15) in the DPM detector. The mean relative improvement (over the baseline) across 20 classes is 9.4% (the mean A.P. improves from 0.32 to 0.35). Figure 7 shows the top detections obtained for train category. The individual detectors do a good job at localizing instances of their respective subcategories. In Figure 5, the discovered subcategories for symmetric (pottedplant) and deformable (cat) classes are displayed.



(a) Subcategories, with part deformations (b) Subcategories, no part deformations (c) [10], no part deformations

Fig. 4. Performance difference with respect to the baseline [10] (x-axis: 20 VOC classes, y-axis: difference in A.P.).

Given the high-degree of alignment across the instances within each subcategory, it is interesting to now check the importance of modeling the deforma-

tions across the parts within each subcategory. Would a simpler model (without deformations) suffice for training the discriminative detectors? We tested this hypothesis with an experiment by turning off the deformable parts. More specifically, rather than sampling "parts" from the high-resolution HOG template (sampled at twice the spatial resolution relative to the features captured by the root template) and modeling the deformation amongst them, we directly use all the features from the high-resolution template. This update to the DPM detector results in a simple multi-scale (two-level pyramid) representation with the finer resolution catering towards improved feature localization.

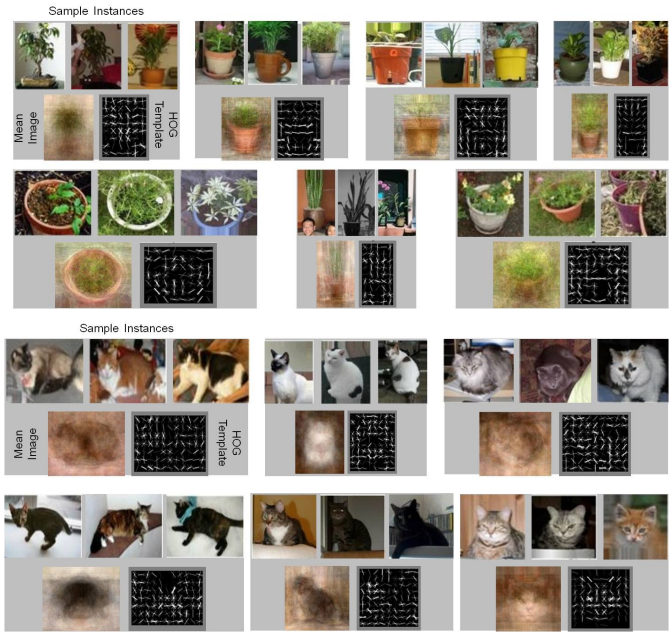


Fig. 5. The visual subcategories discovered for pottedplants correspond to different camera viewpoints, while cats are partitioned based on their pose. The baseline system [10] based on the aspect-ratio, left-right flipping heuristic cannot capture such distinctions (as many of the subcategories share the same aspect-ratio and are symmetric).

Figure 4(b) displays the results obtained. We observe that for 11 of the 20 classes (e.g., pottedplants, tvmonitor, trains, etc) there is no difference in performance. For 6 classes (e.g, person, sofa, etc), turning off deformations hurt the performance, while for 3 classes (diningtable, sheep, etc) performance actually improves. On average, using this two-level pyramid representation for the visual subcategories yields a mean A.P. of 0.31 that is almost on par as the full deformable parts baseline (0.32). These observations suggest that, in practice, the relatively simpler concept of visual subcategories is indeed an equally important

contribution in the DPM detector. They can potentially alleviate the need for part deformations for many object categories.

Computational Issues. In terms of computational complexity, the two-scale visual subcategory detector (K=15) involves one coarse (root) and one fine resolution template per subcategory, totaling a sum of 30 HOG templates. Whereas the DPM detector has K=6 subcategories each with one root and eight part templates, totaling 54 HOG templates, which need to be convolved at test time. In terms of model learning, the DPM detector has the subcategory, as well as the part deformation parameters (six) as latent variables for each of the 24 parts (total of 145 latent variables), while the visual subcategory detector only has the subcategory label as latent. Therefore it not only requires fewer rounds of latent training than required by the DPM detector (leading to faster convergence), but also is less susceptible to getting stuck in a bad local minima. As emphasized in [2], simpler models are preferable, as they can perform better in practice than rich models, which often suffer from difficulties in training.

Number of subcategories. One important parameter is the number of subcategories K. We analyze the influence of K by using different values ($K = [3, 6, 9, 12, 15, 20, 25, 50, 100]$) for a few classes ('boat', 'dog', 'train', 'tv') on the validation set. We plot the variation in the performance over different K in figure 6. The performance gradually increases with increasing K, but stabilizes around K=15. We used $K = 15$ in all the detection experiments.

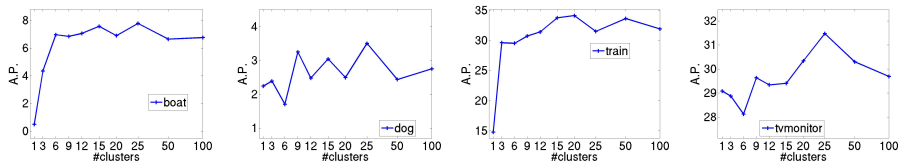


Fig. 6. Variation in detection accuracy as a function of number of subcategories for four distinct VOC2007 classes. The A.P. gradually increases with increasing number of subcategories and stabilizes beyond a point.

Initialization. Proper initialization of subcategories is a key requirement for the success of latent variable models. We analyzed the importance of appearance-based initialization by comparing it with the aspect-ratio based initialization of [2]. Simply increasing the number of subcategories from $K = 6$ to 15 in case of aspect-ratio clustering drops the mean A.P by 1.2%, while appearance-based clustering improves the mean A.P. by 2.3%. (When $K = 6$, aspect-ratio and appearance produced similar result.)

We noticed minimal variation in the final performance on multiple runs with different Kmeans initialization. We found the (latent) discriminative reclustering

step helps in cleaning up any *mistakes* of the initialization step. (Also we observed that most of the reclustering happens in the first latent update.)

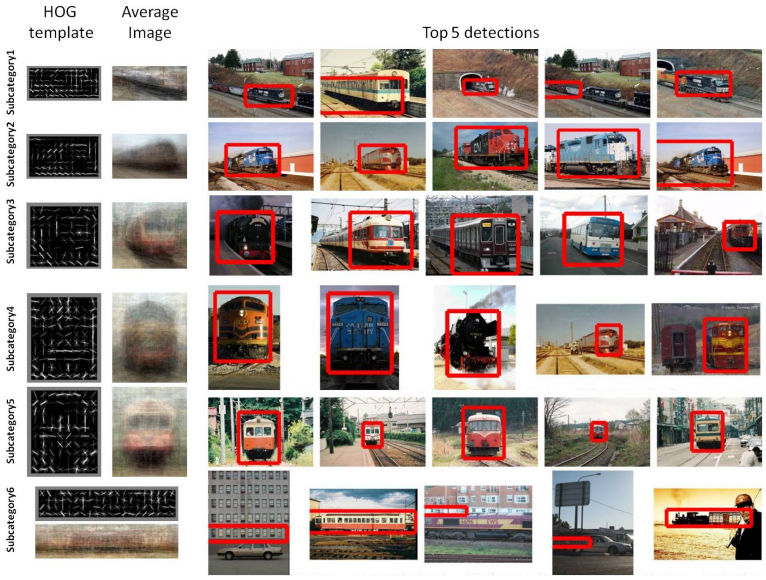


Fig. 7. As the intra-class variance within subcategories is low, the learned detectors perform quite well at localizing instances of their respective subcategories. Notice that for the same aspect-ratio and viewpoint, there are two different subcategories (rows 4,5) discovered for the train category.

5 Conclusion

Given that deformable parts can potentially model exponentially large number of object deformations [2], it is expected that their performance would be far more superior and generalizable in comparison to the use of a fixed number of subcategories. However our empirical analysis has surprisingly pointed out that there is only a minimal performance difference between the use of part deformations compared to the use of subcategories. Further, the fact that a simple method (more subcategories, no parts) does almost as well as the relatively more complex method (fewer subcategories, with parts) is informative as the former is conceptually easy to understand and implement, computationally efficient, and generates easily interpretable models.

Acknowledgments. This research was supported by NSF Grant IIS-0905402.

Bibliography

[1] Everingham, M., Gool, L.V., Williams, C.K.I., Winn, J., Zisserman., A.: The pascal visual object classes challenge (2007) http://pascallin.ecs.soton.ac.uk/challenges/VOC.

[2] Felzenszwalb, P., Girshick, R., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part based models. PAMI (2010)

[3] Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Proc. CVPR. (2005)

[4] Jacobs, R., Jordan, M., Nowlan, S., Hinton, G.: Adaptive mixture of local experts. In: Neural Computation. (1991)

[5] Xu, L., Neufeld, J., Larson, B., Schuurmans, D.: Maximum margin clustering. In: NIPS. (2005)

[6] Seemann, E., Leibe, B., Schiele, B.: Multi-aspect detection of articulated objects. In: CVPR. (2006)

[7] Park, D., Ramanan, D., Fowlkes, C.: Multiresolution models for object detection. In: ECCV. (2010)

[8] Yang, W., Toderici, G.: Discriminative tag learning on youtube videos with latent sub-tags. In: CVPR. (2011)

[9] Felzenszwalb, P., McAllester, D., Ramanan, D.: A discriminatively trained, multiscale, deformable part model. CVPR (2008)

[10] Felzenszwalb, P.F., Girshick, R.B., McAllester, D.: Discriminatively trained deformable part models, release 4. http://people.cs.uchicago.edu/ pff/latent-release4/ (2011)

[11] Felzenszwalb, P.: Object detection grammars. http://www.cs.brown.edu/ pff/talks/grammar.pdf (2011)

[12] Bourdev, L., Maji, S., Brox, T., Malik, J.: Detecting people using mutually consistent poselet activations. In: ECCV. (2010)

[13] Schneiderman, H., Kanade, T.: A statistical method for 3d object detection applied to faces and cars. In: Proc. CVPR. Volume 1. (2000) 746–751

[14] Gu, C., Ren, X.: Discriminative mixture-of-templates for viewpoint classification. In: ECCV. (2010)

[15] Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: CVPR. (2009)

[16] Platt, J.C.: Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In: Advances in Large Margin Classifiers, MIT Press (2000) 61–74

[17] Alexe, B., Petrescu, V., Ferrari, V.: Exploiting spatial overlap to efficiently compute appearance distances between image windows. In: NIPS. (2011)