Ensemble of Exemplar-SVMs for Object Detection and Beyond

Tomasz Malisiewicz
Carnegie Mellon University

Abhinav Gupta
Carnegie Mellon University

Alexei A. Efros
Carnegie Mellon University

Abstract

This paper proposes a conceptually simple but surprisingly powerful method which combines the effectiveness of a discriminative object detector with the explicit correspondence offered by a nearest-neighbor approach. The method is based on training a separate linear SVM classifier for every exemplar in the training set. Each of these Exemplar-SVMs is thus defined by a single positive instance and millions of negatives. While each detector is quite specific to its exemplar, we empirically observe that an ensemble of such Exemplar-SVMs offers surprisingly good generalization. Our performance on the PASCAL VOC detection task is on par with the much more complex latent part-based model of Felzenszwalb et al., at only a modest computational cost increase. But the central benefit of our approach is that it creates an explicit association between each detection and a single training exemplar. Because most detections show good alignment to their associated exemplar, it is possible to transfer any available exemplar meta-data (segmentation, geometric structure, 3D model, etc.) directly onto the detections, which can then be used as part of overall scene understanding.

1. Motivation

A mere decade ago, automatically recognizing everyday objects in images (such as the bus in Figure 1) was thought to be an almost unsolvable task. Yet today, a number of methods can do just that with reasonable accuracy. But let us consider the output of a typical object detector – a rough bounding box around the object and a category label (Figure 1 left). While this might be sufficient for a retrieval task ("find all buses in the database"), it seems rather lacking for any sort of deeper reasoning about the scene. How is the bus oriented? Is it a mini-bus or a double-decker? Which pixels actually belong to the bus? What is its rough geometry? These are all very hard questions for a typical object detector. But what if, in addition to the bounding box, we are able to obtain an *association* with a very similar exemplar from the training set (Figure 1 right), which can provide a high degree of correspondence. Suddenly, any kind of meta-data provided with the training sample (a pixel-wise annotation or label such as viewpoint, segmentation, coarse geometry, a 3D model, attributes, etc.) can be simply transferred to the new instance.

Of course, the idea of associating a new instance with

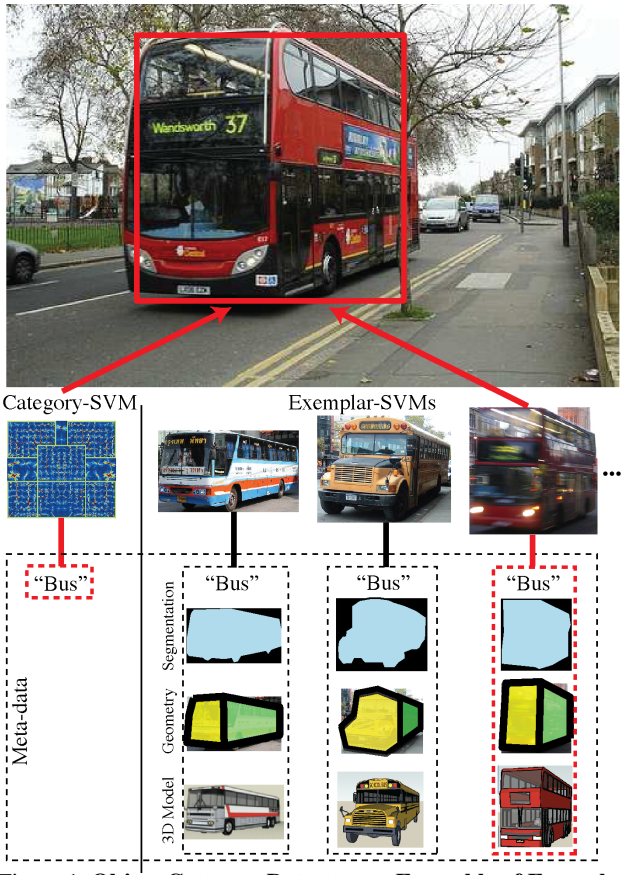

Figure 1. **Object Category Detector vs. Ensemble of Exemplar Detectors.** Output of a typical object detector is just a bounding box and a category label (left). But our ensemble of Exemplar-SVMs is able to associate each detection with a visually similar training exemplar (right), allowing for direct transfer of meta-data such as segmentation, geometry, even a $3D$ model (bottom).

something seen in the past has a long and rich history, starting with the British Empiricists, and continuing as exemplar theory in cognitive psychology, case-based reasoning in AI, instance-based methods in machine learning, data-driven transfer in graphics, etc. In computer vision, this type of non-parametric technique has been quite successful at a variety of tasks including: object alignment [1, 2], scene recognition [19, 21], image parsing [13], among others. However, for object detection data-driven methods, such as [17, 14], have not been competitive against discriminative approaches (though the hybrid method of [6] comes close). Why is this? In our view, the primary difficulty

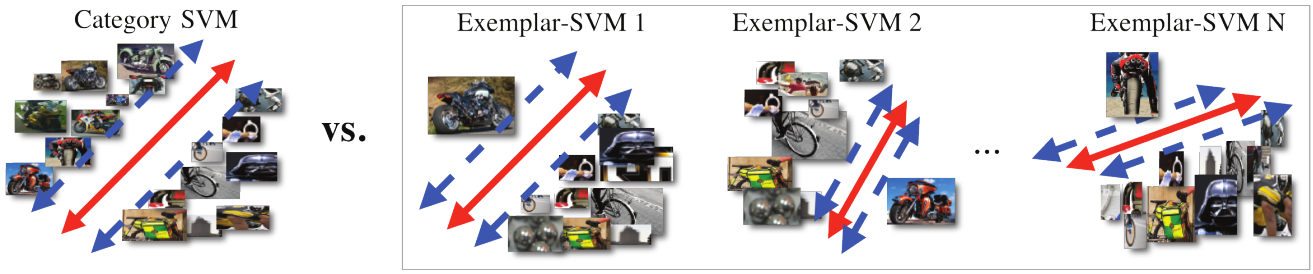

Figure 2. **Category SVM vs. Exemplar-SVMs**. Instead of training a single per-category classifier, we train a separate linear SVM classifier for each exemplar in our dataset with a *single* positive example and millions of negative windows. Negatives come from images not containing any instances of the exemplar's category.

stems from the massive amounts of negative data that must be considered in the detection problem. In image classification, where dataset sizes typically range from a few thousands to a million, using kNN to compute distances to all training images is still quite feasible. In object detection, however, the number of negative windows can go as high as hundreds of millions, making kNN using both positives and negatives prohibitively expensive. Using heuristics, such as subsampling or ignoring the negative set, results in a substantial drop in performance.

In contrast, current state-of-the-art methods in object detection (Dalal-Triggs [7], Felzenszwalb et al. [9] and their derivatives) are particularly well-suited for handling large amounts of negative data. They employ "data-mining" to iteratively sift through millions of negatives and find the "hard" ones which are then used to train a discriminative classifier. Because the classifier is a linear SVM, even the hard negatives do not need to be explicitly stored but are represented parametrically, in terms of a decision boundary.

However, the parametric nature of these classifiers, while a blessing for handling negative data, becomes more problematic when representing the positives. Typically, all positive examples of a given object category are represented as a whole, implicitly assuming that they are all related to each other *visually*. Unfortunately, most standard *semantic* categories (e.g., "car", "chair", "train") do not form coherent *visual* categories [14], thus treating them parametrically results in weak and overly-generic detectors. To address this problem, a number of approaches have used semi-parametric mixture models, grouping the positives into clusters based on meta-data such as bounding box aspect ratio [9], object scale [15], object viewpoint [11], part labels [3], etc. But the low number of mixture components used in practice means that there is still considerable variation within each cluster. As a result, the alignment, or visual correspondence, between the learned model and a detected instance is too coarse to be usable for object association and label transfer. While part-based models [9] allow different localizations of parts within distinct detections, the requirement that they must be shared across all members of a category means that these "parts" are also extremely vague and the resulting correspondences are unintuitive. In general, it might be better to think of these parts as soft, deformable sub-templates. The Poselets approach [3] attempts to address this problem by manually labeling parts and using them to train a set of pose-specific part detectors. While very encouraging, the heavy manual labeling burden is a big limitation of this method.

What seems desirable is an approach that has all the strengths of a Dalal/Triggs/Felzenszwalb/Ramanan-style detector – powerful descriptor, efficient discriminative framework, clever mining of hard-negatives, etc. – but without the drawbacks imposed by a rigid, category-based representation of the positives. To put it another way, what we want is a method that is non-parametric when representing the positives, but parametric (or at least semi-parametric) when representing the negatives. This is the key motivation behind our approach. What we propose is a marriage of the exemplar-based methodology, which allows us to propagate rich annotations from exemplars onto detection windows, with discriminative training, which allows us to learn powerful exemplar-based classifiers from vast amounts of positive and negative data.

2. Approach Overview

Our object detector is based on a very simple idea: to learn *a separate classifier for each exemplar* in the dataset (see Figure 2). We represent each exemplar using a rigid HOG template [7]. Since we use a linear SVM, each classifier can be interpreted as a learned exemplar-specific HOG weight vector. As a result, instead of a single complex category detector, we have a large collection of simpler individual *Exemplar-SVM* detectors of various shapes and sizes, each highly tuned to the exemplar's appearance. But, unlike a standard nearest-neighbor scheme, each detector is discriminatively trained. So we are able to generalize much better without requiring an enormous dataset of exemplars, allowing us to perform surprisingly well even on a moderately-sized training dataset such as the PASCAL VOC 2007 [8].

Our framework shares some similarities with distance-learning approaches, in particular those that learn per-exemplar distance functions (e.g., [10, 14]). However, the crucial difference between a per-exemplar classifier and a

Figure 3. **Comparison.** Given a bicycle training sample from PASCAL (represented with a HOG weight vector $\mathbf{w}$), we show the top 6 matches from the PASCAL test-set using three methods. **Row 1:** naive nearest neighbor (using raw normalized HOG). **Row 2:** Trained Exemplar-SVM (notice how $\mathbf{w}$ focuses on bike-specific edges). **Row 3:** Learned distance function – an Exemplar-SVM but trained in the "distance-to-exemplar" vector space, with the exemplar being placed at the origin (loosely corresponding to [10, 14]).

per-exemplar distance function is that the latter forces the exemplar itself to have the maximally attainable similarity. An Exemplar-SVM has much more freedom in defining the decision boundary, and is better able to incorporate input from the negative samples (see Figure 3 for a comparison, to be discussed later).

One would imagine that training an SVM with a single positive example will badly over-fit. But note that we require far less from a per-exemplar classifier as compared to a per-category classifier – each of our detectors only needs to perform well on visually similar examples. Since each classifier is solving a much simpler problem than in the full-category case, we can use a simple regularized linear SVM to prevent over-fitting. Another crucial component is that, while we only have a single positive example, we have millions of negative examples that we mine from the training set (i.e., from images that do not contain any instances of the exemplar's category). As a result, the exemplar's decision boundary is defined, in large part, by what it is *not*. One of the key contributions of our approach is that we show generalization is possible from a single positive example and a vast set of negatives.

At test-time, we independently run each classifier on the input image and use simple non-maximum suppression to create a final set of detections, where each detection is associated with a single exemplar. However, since our independently-trained classifiers might not output directly comparable scores, we must perform calibration on a validation set. The intuition captured by this calibration step is that different exemplars will offer drastically different generalization potential. A heavily occluded or truncated object instance will have poorer generalization than a cleaner exemplar, thus robustness against even a single bad classifier is imperative to obtaining good overall performance. Since our classifiers are trained without seeing any other positive instances but itself, we can use them for calibration in a "leave-all-but-one-out" fashion.

It is worthwhile pointing out some of the key differences between our approach and other related SVM-based techniques such as one-class SVMs [18, 5], multi-class kernel SVMs, kernel-learning approaches [20], and the KNN-SVM algorithm [22]. All of these approaches require mapping the exemplars into a common feature space over which a similarity kernel can be computed (which we avoid), but more importantly, kernel methods lose the semantics of single-exemplar associations which are necessary for high quality meta-data transfer.

3. Algorithm Description

Given a set of training exemplars, we represent each exemplar E via a rigid HOG template, $\mathbf{x}_E$. We create a descriptor from the ground-truth bounding box of each exemplar with a cell size of 8 pixels using a sizing heuristic which attempts to represent each exemplar with roughly 100 cells. Instead of warping each exemplar to a canonical frame, we let each exemplar define its own HOG dimensions respecting the aspect ratio of its bounding box. We create negative samples of the same dimensions as $\mathbf{x}_E$ by extracting negative windows, $\mathcal{N}_E$, from images not containing any objects from the exemplar's category.

Each Exemplar-SVM, $(\mathbf{w}_E, b_E)$, tries to separate $\mathbf{x}_E$ from *all* windows in $\mathcal{N}_E$ by the largest possible margin in the HOG feature space. Learning the weight vector $\mathbf{w}_E$ amounts to optimizing the following convex objective:

$$\Omega_E(\mathbf{w}, b) = ||\mathbf{w}||^2 + C_1 h(\mathbf{w}^T \mathbf{x}_E + b) + C_2 \sum_{\mathbf{x} \in \mathcal{N}_E} h(-\mathbf{w}^T \mathbf{x} - b)$$

We use the hinge loss function $h(x) = \max(0, 1 - x)$, which allows us to use the hard-negative mining approach to cope with millions of negative windows because the solution only depends on a small set of negative support vectors.

Figure 3 offers a visual comparison of the proposed Exemplar-SVM method against two alternatives for the task

Figure 4. **Exemplar-SVMs.** A few "train" exemplars with their top detections on the PASCAL VOC test-set. Note that each exemplar's HOG has its own dimensions. Note also how each detector is specific not just to the train's orientation, but even to the type of train.

of detecting test-set matches for a single exemplar, a snow-covered bicycle. The first row shows a simple nearest-neighbor approach. The second row shows the output of our proposed Exemplar-SVM. Note the subtle changes in the learned HOG vector $\mathbf{w}$, making it focus more on the bicycle. The third row shows the output of learning a distance function, rather than a linear classifier. For this, we applied the single-positive Exemplar-SVM framework in the "distance-to-exemplar" vector space, with the exemplar being placed at the origin (this is conceptually similar to [14, 10]). We observed that the centered-at-exemplar constraint made the distance function less powerful than the linear classifier (see Results section). Figure 4 shows a few Exemplar-SVMs from the "train" category along with their top detections on the test-set. Note how specific each detector is – not just to the train's orientation, but even the type of train.

3.1. Calibration

Using the procedure above, we train an ensemble of Exemplar-SVM, one for each positive instance in the training set. However, due to the independent training procedure, their outputs are not necessarily compatible. A common strategy to reconcile the outputs of multiple classifiers is to perform calibration by fitting a probability distribution to a held-out set of negative and positive samples [16]. However, in our case, since each exemplar-SVM is supposed to fire only on *visually similar* examples, we cannot say for sure which of the held-out samples should be considered as positives *a priori*. For example, for a frontal view of an train, only other frontal views of similar trains should be considered as positives. Fortunately, just like during training, what we can be sure about is that the classifier should not fire on negative windows. Therefore, we let each exemplar select its own positives and then use the SVM output scores on these positives, in addition to lots of held-out negatives, to calibrate the Exemplar-SVM.

To obtain each exemplar's calibration positives, we run the Exemplar-SVM on the validation set, create a set of non-redundant detections using non-maximum suppression, and compute the overlap score between resulting detections and ground-truth bounding-boxes. We treat all detections which overlap by more than 0.5 with ground-truth boxes as positives (this is the standard PASCAL VOC criterion for a successful detection). All detections with an overlap lower than 0.2 are treated as negatives, and we fit a logistic function to these scores. Note that, although we cannot guarantee that highly overlapping correct detections will indeed be visually similar to the exemplar, with very high probability they will be, since they were highly ranked by the exemplar-SVM in the first place.

Our calibration step can be interpreted as a simple re-scaling and shifting of the decision boundary (see Figure 5) – poorly performing exemplars will be suppressed by having their decision boundary move towards the exemplar and well-performing exemplars will be boosted by having their decision boundary move away from the exemplar. While the resulting decision boundary is no longer an optimal solution for the local-SVM problem, empirically we found this procedure greatly improves the inter-exemplar ordering. Given a detection $\mathbf{x}$ and the learned sigmoid parameters (α_E, β_E), the calibrated detection score for exemplar E is as follows:

$$f(\mathbf{x}|\mathbf{w}_E, \alpha_E, \beta_E) = \frac{1}{1 + e^{-\alpha_E(\mathbf{w}_E^T\mathbf{x} - \beta_E)}}$$

While the logistic fitting is performed independently for each exemplar, we found that it gives us a considerable boost in detection performance over using raw SVM output scores. At test-time, we create detections from each classifier by thresholding the raw SVM output score at -1 (the negative margin) and then rescale them using each exemplar's learned sigmoid parameters.

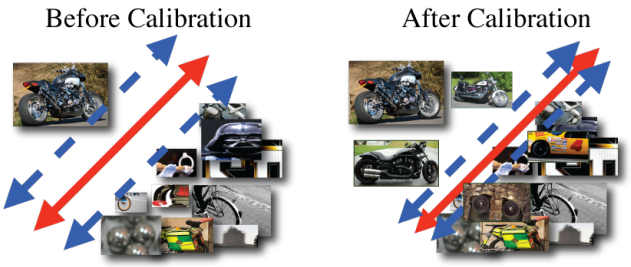

Figure 5. **Exemplar-SVM calibration**. The calibration step rescales the SVM scores but does not affect the ordering of the matches, allowing us to compare the outputs of multiple independently-trained Exemplar-SVMs.

3.2. Implementation Details

We use `libsvm` [4] to train each exemplar's $\mathbf{w}$. We alternate between learning the weights given an active set of negative windows, and mining additional negative windows using the current $\mathbf{w}$ as in [9]. We use the same regularization parameter $C_1 = 0.5$ and $C_2 = .01$ for all exemplars, but found our weight vectors to be robust to a wide range of Cs, especially since they are re-scaled during calibration. While we learn $\mathbf{w}$'s for a large number of exemplars, each exemplar's learning problem and calibration can be solved *independently* allowing for easy parallelization. We consider images as well as their left-right flipped counterparts for both training and testing.

The run-time complexity of our approach at test time scales linearly with the number of positive instances (but unlike kernel-SVM methods, not the negatives). However, in practice, the bottleneck appears to be per-image tasks (loading, computing HOG pyramid etc.) – the actual per-instance computation is just a single dot-product, which can be done extremely fast. For an average PASCAL class ($\sim$ 300 training examples yielding $\sim$ 300 separate classifiers) our method is only 6 times slower than a category-based method such as [9]. More generally, because of the long-tailed distribution of objects in the world (10% of objects own 90% of exemplars [19]), the extra cost of using exemplars vs. categories will greatly diminish as the number of categories increases.

4. Experimental Evaluation

We first evaluate our Exemplar-SVM framework on the well-established benchmark task of object detection. We then showcase the ability of our method to produce a high-quality alignment between training exemplars and their associated detections. For this, we present results on a set of tasks including segmentation, qualitative-geometry estimation, 3D model transfer, and related object priming.

For our experiments, we use a single source of exemplars: the PASCAL VOC 2007 dataset [8] – a popular dataset used to benchmark object detection algorithms. During training, we learn a separate classifier $\mathbf{w}$ for each of the $12,608$ exemplars from the 20 categories in $5,011$ `trainval` images. We mine hard negatives from out-of-class images in the `train` set and perform calibration using all positive and negative images in `trainval` (See Section 3.1).

4.1. Object Detection

At test time, each Exemplar-SVM creates detection windows in a sliding-window fashion, but instead of using a standard non-maxima-suppression we use an exemplar co-occurence based mechanism for suppressing redundant responses. For each detection we generate a context feature similar to [3, 9] which pools in the SVM scores of nearby (overlapping) detections and generates the final detection score by a weighted sum of the local SVM score and the context score. Once we obtain the final detection score, we use standard non-maximum suppression to create a final, sparse set of detections per image.

We report results on the 20-category PASCAL VOC 2007 `comp3` object detection challenge. Figure 6 shows several detections (green boxes) produced by our Exemplar-SVM framework. We also show the super-imposed exemplar (yellow boxes) associated with each detection. Following the protocol of the VOC Challenge, we evaluate our system on a per-category basis on the `test` set, consisting of $4,952$ images. We compare the performance of our approach (**ESVM+Co-oc**) to several exemplar baselines apart from the VOC results reported in [9, 6]. These results have been summarized in Table 1 as Average Precision per class. Our results show that standard Nearest Neighbor [1] (NN) does not work at all. While the performance improves after calibration (NN+Cal), it is still not comparable to other approaches due to its lack of modeling negative data. We also compared against a distance function formulation similar to the one proposed in [14] but learned using a single positive instance. The results clearly indicate that the extra constraint due to a distance function parameterization is worse than using a hyperplane. To highlight the importance of using the co-occurence mechanism above, we also report our results using calibration (ESVM+Cal).

On the PASCAL test set, our full system obtains a mean Average Precision (mAP) of .227, which is competitive with with Felzenszwalb's state-of-the-art deformable part-based mixture model. Note however, that our system does not use parts (though they could be easily added) so the comparison is not entirely fair. Therefore, we also compare our performance to Dalal/Triggs baseline, which uses a single category-wise linear SVM with no parts, and attains a mAP of .097, which is less than half of ours. We also compared against the PASCAL VOC 2007 winning entry, the exemplar-based method of Chum et al. [6], and found that our system beats it on 4 out of 6 categories for which they

[1] We experimented with multiple similarity metrics and found that a dot product with a normalized HOG template worked the best. The normalized HOG template is created by subtracting a constant from the positive HOG features to make them 0-mean.

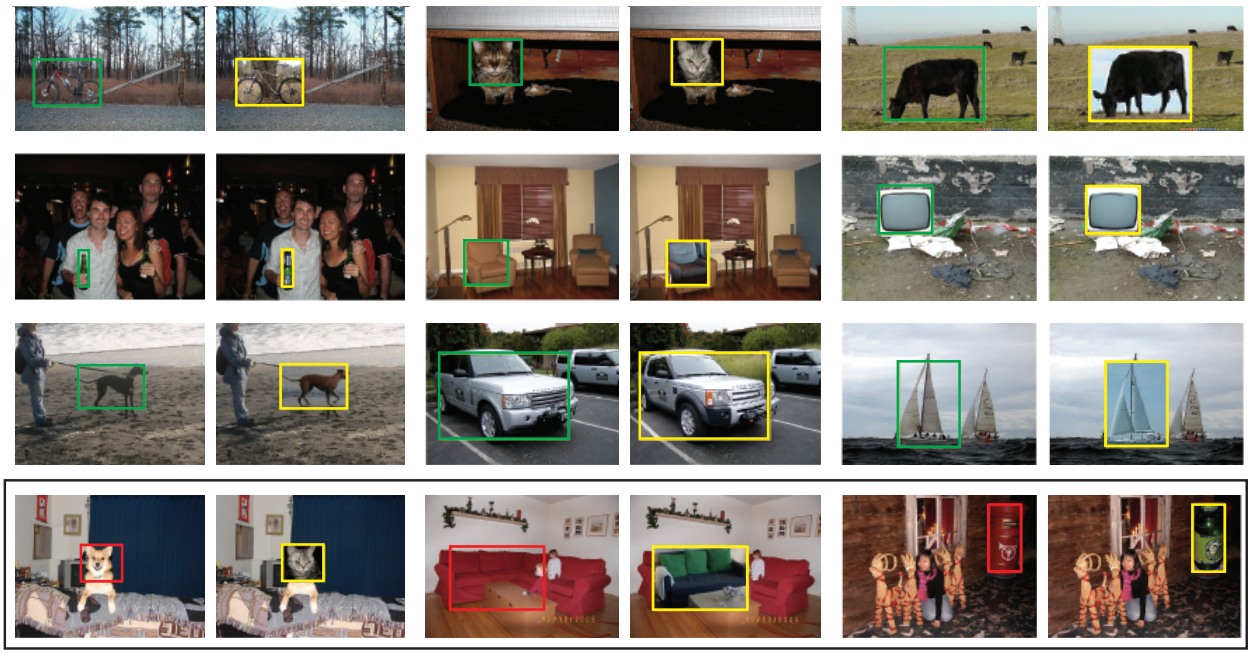

Figure 6. **Object Detection and Appearance Transfer.** Each example shows a detection from our ensemble of Exemplar-SVMs along with the appearance transferred directly from the source exemplar, to demonstrate the high quality of visual alignment. Bottom row shows object category detection failures.

Approach	aeroplane	bicycle	bird	boat	bottle	bus	car	cat	chair	cow	diningtable	dog	horse	motorbike	person	pottedplant	sheep	sofa	train	tvmonitor	mAP
NN	.006	.094	.000	.005	.000	.006	.010	.092	.001	.092	.001	.004	.096	.094	.005	.018	.009	.008	.096	.144	.039
NN+Cal	.056	.293	.012	.034	.009	.207	.261	.017	.094	.111	.004	.033	.243	.188	.114	.020	.129	.003	.183	.195	.110
DFUN+Cal	.162	.364	.008	.096	.097	.316	.366	.092	.098	.107	.002	.093	.234	.223	.109	.037	.117	.016	.271	.293	.155
E-SVM+Cal	.204	.407	.093	.100	.103	.310	.401	.096	.104	.147	.023	.097	.384	.320	.192	.096	.167	.110	.291	.315	.198
E-SVM+Co-occ	.208	.480	.077	.143	.131	.397	.411	.052	.116	.186	.111	.031	.447	.394	.169	.112	.226	.170	.369	.300	.227
CZ [6]	.262	.409	–	–	–	.393	.432	–	–	–	–	–	–	.375	–	–	–	–	.334	–	–
DT [7]	.127	.253	.005	.015	.107	.205	.230	.005	.021	.128	.014	.004	.122	.103	.101	.022	.056	.050	.120	.248	.097
LDPM [9]	.287	.510	.006	.145	.265	.397	.502	.163	.165	.166	.245	.050	.452	.383	.362	.090	.174	.228	.341	.384	.266

Table 1. **PASCAL VOC 2007 object detection results.** We compare our full system (ESVM+Co-occ) to four different exemplar based baselines including NN (Nearest Neighbor), NN+Cal (Nearest Neighbor with calibration), DFUN+Cal (learned distance function with calibration) and ESVM+Cal (Exemplar-SVM with calibration). We also compare our approach against global methods including our implementation of Dalal-Triggs (learning a single global template), LDPM [9] (Latent deformable part model), and Chum et al. [6]'s exemplar-based method. [The NN, NN+Cal and DFUN+Cal results for person category are obtained using 1250 exemplars]

submitted results.

4.2. Association and Meta-data transfer

To showcase the high quality correspondences we obtain with our method, we looked at several meta-data transfer tasks. For the transfer applications we used the ESVM+Cal method because even though using the exemplar co-occurence matrix boosts object detection performance, it uses multiple overlapping exemplars to score windows. Calibration produces much higher quality alignments because associations are scored independently. Once we establish an association between a detection and an exemplar, we simply transfer the exemplar-aligned meta-data onto the detection.

Segmentation and Geometry Estimation: For the task of segmentation, the goal is to estimate which pixels belong to a given object and which do not. Figure 7 shows some qualitative segmentation examples on a wide variety of object classes.

For quantitative evaluation, we asked labelers to segment and geometrically annotate all of the instances in the "bus" category in the PASCAL VOC 2007 dataset. For the segmentation task, our method performs at a pixel-wise accuracy of 90.6%. For geometry estimation, the goal is to assign labels to pixels indicating membership to one of 3 "left," "front," and "right" dominant orientation classes [12]. We compare our Exemplar-SVM system against two baselines: (a) Hoiem's pre-trained generic geometric class estimation algorithm [12]; (b) Using [9] to detect objects followed by simple NN to create associations. We obtain a 62.3% pixelwise labeling accuracy us-

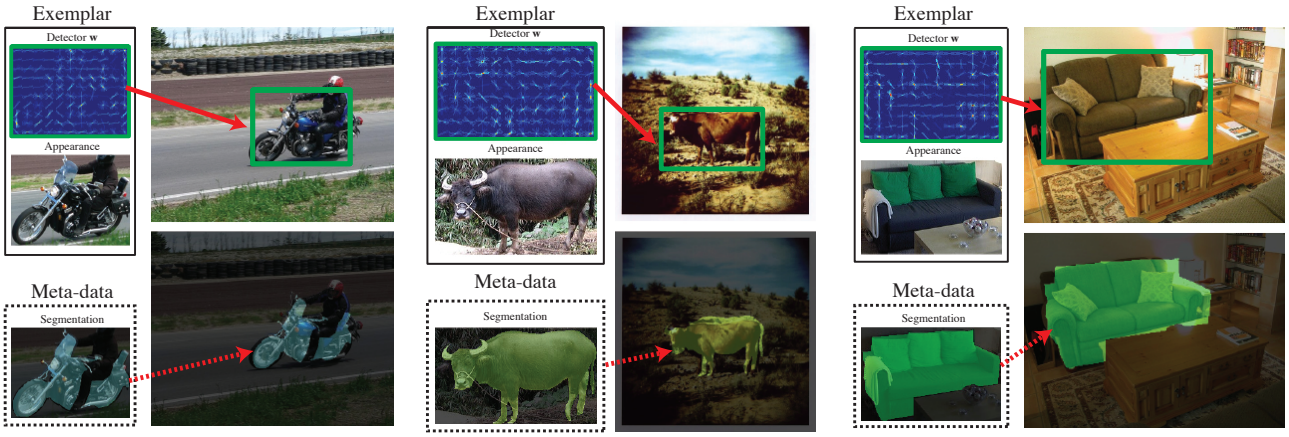

Figure 7. **Segmentation Transfer.** Object segmentations are transferred from the exemplar directly onto the detection window.

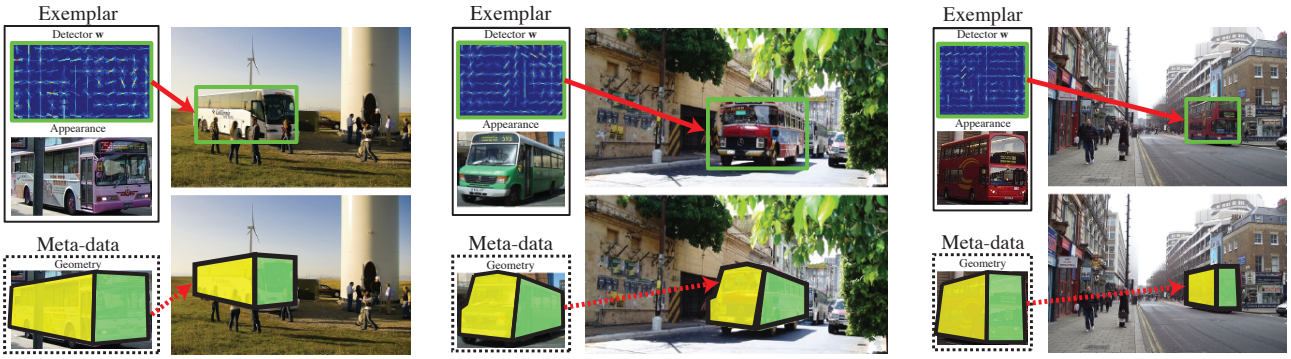

Figure 8. **Qualitative Geometry Transfer.** We transfer geometric labeling from bus exemplars onto corresponding detections.

ing our Exemplar-SVM approach as compared to the 43.0% obtained using [12] and 51.0% using [9]+NN. This clearly shows that while our transfer is simple, it is definitely not trivial as it relies on obtaining strong alignment between the exemplar and the detection (see qualitative results in Figure 8). Global methods fail to generate such alignments, leading to much lower performance.

3D Model Transfer: We annotated a subset of chair exemplars with $3D$ models from Google's $3D$ Warehouse (and aligned with Google Sketch-Up $3D$ model-to-image alignment tool). Given a single exemplar, labelers were asked to find the most visually similar model in the $3D$ Warehouse for that instance and perform the alignment. Due to the high quality of our automatically-generated associations, we were able to simply transfer the exemplar-aligned $3D$ model directly onto the detection window without any additional alignment, see Figure 9.

Related Object Priming: Exemplars often show an interplay of multiple objects, thus any other objects which sufficiently overlap with the exemplar can be viewed as additional meta-data belonging to the exemplar. This suggests using detectors of one category to help "prime" objects of another category. We look at the following task: predicting a bounding box for "person" given a detection of category X, where X is either a horse, motorbike, or bicycle (see Figure 10 for qualitative results). We quantitatively evaluated the prediction performance and compared against a baseline which predicts a person presence based on majority voting. Our method considerably outperforms the baseline (72.46% as compared to 58.67% for the baseline), suggesting that our exemplar associations provide good alignment of related objects as well.

5. Conclusion

We presented a simple yet powerful method which recasts an exemplar-based approach in a discriminative framework. Our method is based on training a separate classifier for each exemplar and we show that generalization is possible from a single positive example and millions of negatives. Our approach performs on par with state-of-the-art methods for object detection but creates a strong alignment between the detection and training exemplar. This allows us to go beyond the detection task and enables a variety of applications based on meta-data transfer. We believe that our work opens up the door for many new exciting applications in object recognition, scene understanding, and computer graphics.

Acknowledgements: This work is supported by MSR-CMU Center for Computational Thinking. Additional support by MURI Grant N000141010934 and NSF IIS-0905402. The authors thank Reviewer#3 for single-handedly rescuing this paper from the clutches of ICCV death.

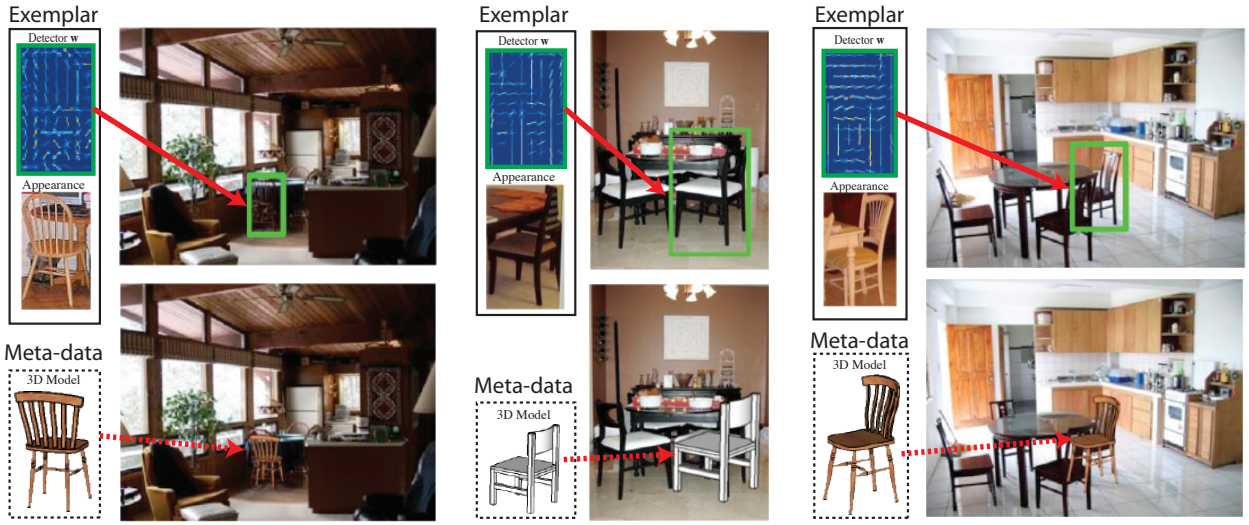

Figure 9. **3D Model transfer.** In each of these 3 examples, the green box in the top image shows the detection window, and the bottom shows the automatically transferred $3D$ model.

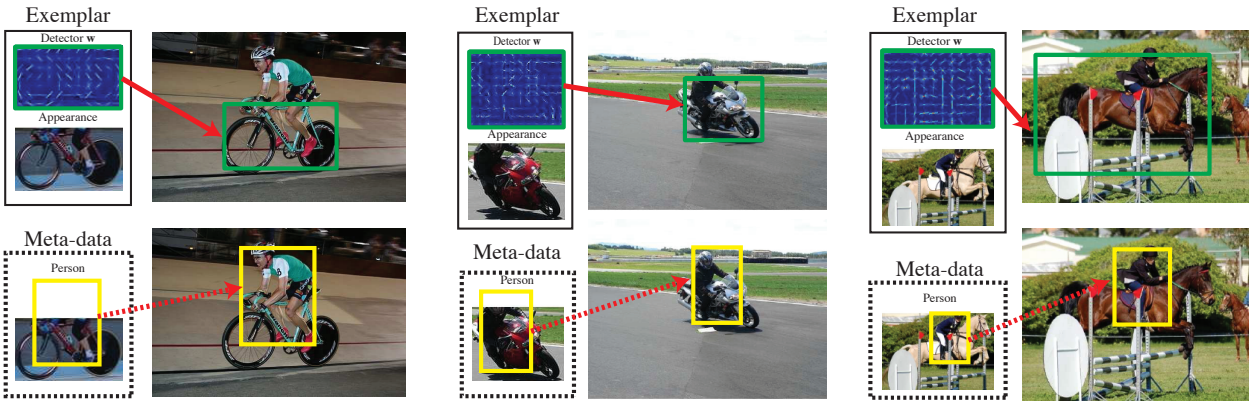

Figure 10. **Related Object Priming.** A bicycle/motorbike/horse exemplar is used to predict bounding box for "person".

References

[1] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *PAMI*, 2002. 1

[2] A. C. Berg, T. L. Berg, and J. Malik. Shape matching and object recognition using low distortion correspondence. *CVPR*, 2005. 1

[3] L. Bourdev, S. Maji, T. Brox, and J. Malik. Detecting people using mutually consistent poselet activations. *ECCV*, 2010. 2, 5

[4] C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001. 5

[5] Y. Chen, X. Zhou, , and T. S. Huang. One-class svm for learning in image retrieval. *ICIP*, 2001. 3

[6] O. Chum and A. Zisserman. An exemplar model for learning object classes. *CVPR*, 2007. 1, 5, 6

[7] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. *CVPR*, 2005. 2, 6

[8] M. Everingham, L. V. Gool, C. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 2010. 2, 5

[9] P. F. Felzenszwalb, R. B. Girschick, D. McCallester, and D. Ramanan. Object detection with discriminatively trained part based models. *PAMI*, 2010. 2, 5, 6, 7

[10] A. Frome and J. Malik. Image retrieval and recognition using local distance functions. *NIPS*, 2006. 2, 3, 4

[11] C. Gu and X. Ren. Discriminative mixture-of-templates for viewpoint classification. *ECCV*, 2010. 2

[12] D. Hoiem, A. A. Efros, and M. Hebert. Geometric context from a single image. *ICCV*, 2005. 6, 7

[13] C. Liu, J. Yuen, and A. Torralba. Nonparametric scene parsing: Label transfer via dense scene alignment. *CVPR*, 2009. 1

[14] T. Malisiewicz and A. A. Efros. Recognition by association via learning per-exemplar distances. *CVPR*, 2008. 1, 2, 3, 4, 5

[15] D. Park, D. Ramanan, and C. Fowlkes. Multiresolution models for object detection. *ECCV*, 2010. 2

[16] J. C. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in Large Margin Classifiers*, 1999. 4

[17] B. C. Russell, A. Torralba, C. Liu, R. Fergus, and W. T. Freeman. Object recognition by scene alignment. *NIPS*, 2007. 1

[18] B. Schlkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson. Estimating the support of a high-dimensional distribution. *Neural Computation*, 2001. 3

[19] A. Torralba, R. Fergus, and W. T. Freeman. 80 million tiny images: a large dataset for non-parametric object and scene recognition. *PAMI*, 30, 2008. 1, 5

[20] A. Vedaldi, V. Gulshan, M. Varma, and A. Zisserman. Multiple kernels for object detection. *ICCV*, 2009. 3

[21] J. Xiao, J. Hays, K. Ehinger, A. Oliva, and A. Torralba. Sun database: Large-scale scene recognition from abbey to zoo. *CVPR*, 2010. 1

[22] H. Zhang, A. Berg, M. Maire, and J. Malik. SVM-KNN: Discriminative nearest neighbor classification for visual category recognition. *CVPR*, pages 2126–2136, 2006. 3