Semantic Kernel Forests from Multiple Taxonomies

Sung Ju Hwang
University of Texas
sjhwang@cs.utexas.edu

Fei Sha
University of Southern California
feisha@usc.edu

Kristen Grauman
University of Texas
grauman@cs.utexas.edu

Abstract

We propose a discriminative feature learning approach that leverages *multiple* hierarchical taxonomies representing different semantic views. For each taxonomy, we first learn a tree of semantic kernels, where each node has a Mahalanobis kernel optimized to distinguish between the classes in its children nodes. Then, using the resulting *semantic kernel forest*, we learn class-specific kernel combinations to select only those kernels relevant for category recognition, with a novel hierarchical regularizer that exploits the taxonomies' structure. We demonstrate our method on challenging object recognition datasets.

1 Introduction

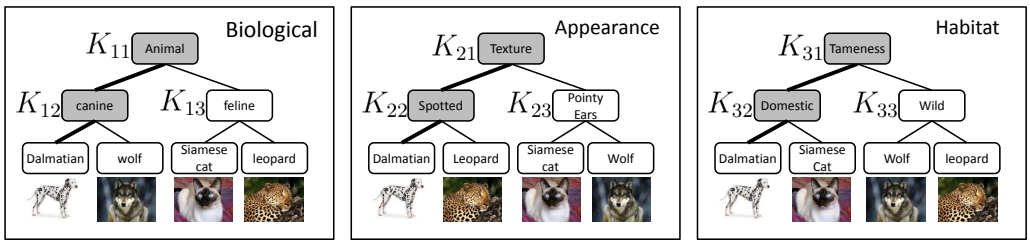
Object recognition research has made impressive gains in recent years, with particular success in using discriminative learning algorithms. However, as the basic "image features + labels + classifier" paradigm has reached a level of maturity, it is time to reach beyond it by incorporating richer *semantic* knowledge about the object categories themselves. Large-scale recognition is not merely about using massive datasets; in fact, the semantic structure underlying that data may be a key to scalability for recognition algorithms.

One appealing source of external knowledge is a taxonomy (e.g. WordNet), which is a tree that groups classes together in its nodes according to some human-designed merging or splitting criterion. Such trees implicitly embed cues about human perception of categories, and how they relate to one another at different granularities. Thus, in the context of visual object recognition, such a structure has the potential to guide the selection of meaningful low-level features.

Two fundamental issues, however, complicate its use. First, a given taxonomy may offer hints about visual relatedness, but its structure need not entirely align with useful splits for recognition. (For example, *monkey* and *dog* are fairly distant semantically according to WordNet, yet they share a number of visual features. An *apple* and *applesauce* are semantically close, yet are easily separable with basic visual features.) Second, given the complexity of visual objects, it is highly unlikely that some *single* optimal semantic taxonomy exists for recognition. Rather, objects can be organized along many semantic "views". See Figure 1.

Motivated by these issues, we present a discriminative feature learning approach that leverages *multiple* taxonomies capturing different semantic views of the object categories. Our key insight is that some combination of the semantic views will be most informative to distinguish a given visual category. While each view differs in its implicit human-designed splitting criterion, all separate some classes from others, thereby lending (often complementary) discriminative cues. Thus, rather than commit to a single representation, we aim to inject pieces of the various taxonomies as needed.

To this end, we propose *semantic kernel forests*. Our method takes as input training images labeled by their object category, as well as a series of taxonomies, each of which hierarchically partitions the labels by a different semantic view. For each taxonomy, we first learn a tree of semantic kernels that capture granularity-specific similarities. Then, using the resulting semantic kernel forest from all taxonomies, we learn a class-specific kernel combination that selects only the features relevant for categorizing that class against the rest.



$$K_{dalmatian} = B_{11}K_{11} + B_{12}K_{12} + B_{21}K_{21} + B_{22}K_{22} + B_{31}K_{31} + B_{32}K_{32}$$

Figure 1: **Main idea:** We assume that multiple semantic taxonomies exist, each one representing a different semantic "view". Rather than commit to a single taxonomy, we learn a tree of kernels for each taxonomy that captures the granularity-specific similarity at each node, and then combine them into a "kernel forest" that exploits the inter-taxonomic structure for object categorization.

Our main contribution is to simultaneously exploit multiple semantic taxonomies for visual feature learning. Whereas past work focuses on building object hierarchies for scalable classification (e.g., [1, 2]) or using WordNet to gauge semantic distance (e.g., [3, 4, 5]), we learn discriminative kernels that capitalize on the cues in diverse taxonomy views, leading to better recognition accuracy.

2 Approach

Our method has two main steps: learning the base kernels—which we call a *semantic kernel forest* (Sec. 2.1), and learning their combination across taxonomies (Sec. 2.2).

We assume that we are given a labeled dataset $\mathcal{D} = \{(\boldsymbol{x}_i, y_i)\}_{n=1}^{\mathsf{N}}$ where $(\boldsymbol{x}_i, y_i)$ stands for the ith instance (feature vector) and its class label is y_i, as well as a set of tree-structured taxonomies $\{\mathcal{T}_t\}_{t=1}^{\mathsf{T}}$. Each taxonomy $\mathcal{T}_t$ is a collection of nodes. The leaf nodes correspond to class labels, and the inner nodes correspond to superclasses—or, more generally, *semantically meaningful groupings of categories*. We index those nodes with double subscripts tn, where t refers to the tth taxonomy and n to the nth node in that taxonomy. [1]

2.1 Learning a semantic kernel forest

The first step is to learn a forest of granularity- and view-specific base kernels, that are tuned to similarities implied by the given taxonomies. Formally, for each taxonomy $\mathcal{T}_t$, we learn a set of Gaussian kernels for the superclass at every internal node tn for which $n \geq \mathsf{C}+1$, parameterized as

$$K_{tn}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp\{-\gamma_{tn} d^2_{\boldsymbol{M}_{tn}}(\boldsymbol{x}_i, \boldsymbol{x}_j)\} = \exp\{-\gamma_{tn}(\boldsymbol{x}_i - \boldsymbol{x}_j)^{\mathrm{T}} \boldsymbol{M}_{tn}(\boldsymbol{x}_i - \boldsymbol{x}_j)\}, \quad (1)$$

where the Mahalanobis distance metric $\boldsymbol{M}_{tn}$ is used in lieu of the conventional Euclidean metric.

We want the base kernels K_{tn} to encode similarity between examples using features that reflect their respective granularity in the taxonomy, which will select features that are helpful to distinguish the node tn's subclasses. Beyond that, however, we specifically want it to use features that are *as different as possible* from the features used by its ancestors. Doing so ensures that the subsequent combination step can choose a sparse set of "disconnected" features.

To that end, we use our Tree of Metrics (ToM) technique [6] to learn the Mahalanobis parameters $\boldsymbol{M}_{tn}$. In ToM, metrics are learned by balancing two forces: i) discriminative power and ii) a preference for different features to be chosen between parent and child nodes. The latter exploits the taxonomy semantics, based on the intuition that features used to distinguish more abstract classes (dog vs. cat) should differ from those used for finer-grained ones (Siamese vs. Persian cat).

Briefly, for each node tn, the training data is reduced to $\mathcal{D}_n = \{(\boldsymbol{x}_i, y_{in})\}$, where y_{in} is the label of n's *child* $\boldsymbol{x}_i$. The metrics are learned jointly, with each node mutually encouraging the others to use non-overlapping features. ToM achieves this by augmenting a large margin nearest neighbor [7] loss function $\sum_n \ell(\mathcal{D}_n; \mathcal{M}_{tn})$ with the following *disjoint sparsity regularizer*:

$$\Omega_d(\boldsymbol{M}) = \lambda \sum_{n \geq \mathsf{C}+1} \mathsf{Trace}[\boldsymbol{M}_{tn}] + \mu \sum_{n \geq \mathsf{C}+1} \sum_{m \sim n} \|\mathsf{diag}(\boldsymbol{M}_{tn}) + \mathsf{diag}(\boldsymbol{M}_{tm})\|_2^2, \quad (2)$$

[1] We assign the leaf (class) nodes a number between 1 and C, where C is the number of class labels.

where $m \sim n$ denotes that node m is either an ancestor or descendant of n. The first part of the regularizer encourages sparsity in the diagonal elements of $\boldsymbol{M}_{tn}$, and the second part incurs a penalty when two different metrics "compete" for the same diagonal element, i.e., to use the same feature dimension. The resulting optimization problem is convex and can be solved efficiently [6].

After learning the metrics $\{\boldsymbol{M}_{tn}\}$ in each taxonomy, we construct base kernels as in eq. (1). We call the collection $\mathcal{F} = \{K_{tn}\}$ a *semantic kernel forest*. Figure 1 shows an illustrative example.

2.2 Learning class-specific kernels across taxonomies

We next combine the semantic base kernels discriminatively to improve classification.

Basic setting To learn class-specific features (or kernels), we compose a one-versus-rest supervised learning problem. From each taxonomy, we select base kernels that correspond to the nodes on the path from the root to the leaf node class. For example, in the Biological taxonomy of Figure 1, for the category *Dalmatian*, this path includes the nodes (superclasses) *canine* and *animal*. Thus, for class c, the linearly combined kernel is given by

$$F_c(\boldsymbol{x}_i, \boldsymbol{x}_j) = \sum_t \sum_{n \sim c} \beta_{ctn} K_{tn}(\boldsymbol{x}_i, \boldsymbol{x}_j), \tag{3}$$

where $n \sim c$ indexes the nodes that are ancestors of c, which is a leaf node. The combination coefficients β_{ctn} must be nonnegative to ensure the positive semidefiniteness of $F_c(\cdot, \cdot)$.

We apply the kernel $F_c(\cdot, \cdot)$ to construct the one-versus-rest binary classifier to distinguish instances from class c from all other classes, and optimize $\boldsymbol{\beta}_c = \{\beta_{ctn}\}$ such that the classifier attains the lowest empirical misclassification risk. This can be solved by multiple kernel learning (MKL) [8].

Hierarchical regularization With multiple taxonomies, we have *multiple* different splits to differentiate a class from the target class, which requires selection of a better split. Here, we want to favor kernels at higher-level nodes to lower-level nodes, because intuitively, higher-level kernels relate to more classes, thus are likely essential to reduce loss. To this end, we design a novel structured MKL regularizer Ω that prefers larger weights for a parent node compared to its children:

$$\Omega(\boldsymbol{\beta}_c) = \lambda \sum_{t, n \sim c} \beta_{ctn} + \mu \sum_{t, n \sim c} \max(0, \beta_{ctn} - \beta_{ctp_n} + 1). \tag{4}$$

The first part prefers a sparse set of kernels. The second hinge loss term enforces weight assigned to a node n be less than the weight assigned to the node's parent p_n, by a large margin.

Our learning problem is cast as a convex optimization that balances the discriminative MKL loss and the regularizer in eq. (4). We use the projected subgradient method to solve it, for its ease of implementation and practical effectiveness [9]. For more details, see [10].

3 Experiments

We validate our approach on multiple image datasets, and compare to several informative baselines.

Image datasets and taxonomies We consider two publicly available image collections: Animals with Attributes (AWA) [11] and ImageNet [12][2]. We form two datasets from AWA: **1) AWA-4**: the four classes shown in Fig. 1, with $2,228$ images, and **2) AWA-10**: the ten classes in [11], with $6,180$ images. The third dataset, **ImageNet-20**, consists of $28,957$ total images spanning 20 non-animal classes from ILSVRC2010. The raw image features are bag-of-words on SIFT, provided with the datasets, with the dimensionality reduced to 100 with PCA to speed up the ToM training.

To obtain multiple taxonomies per dataset, we use WordNet and attribute labels. To form semantic taxonomies on attributes, we first manually divide the attributes into subsets (e.g. Appearance, Habitat) according to their mutual semantic relevance; then, for each subset, we perform agglomerative clustering on vectors of the training images' real-valued attributes. (See trees in Figure 2).

Baselines We compare our method to three baselines: **1) Raw feature kernel:** an RBF kernel on the original image features. **2) Raw feature kernel + MKL:** MKL combination of the RBF kernels

[2] `attributes.kyb.tuebingen.mpg.de/` and `image-net.org/challenges/LSVRC/2011`

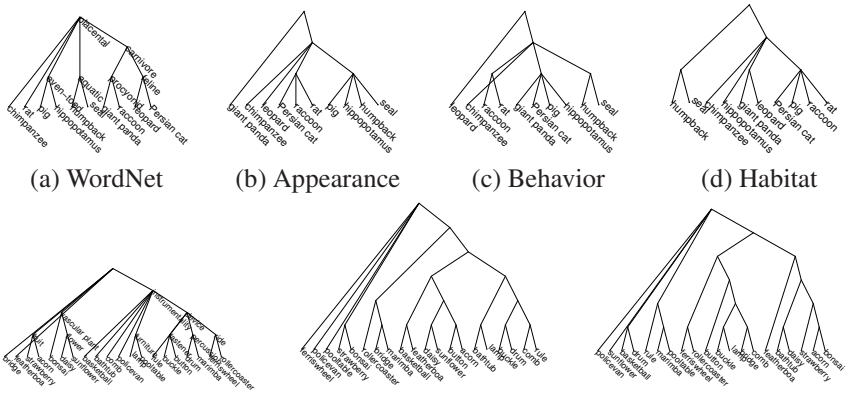


Figure 2: Taxonomies for the AWA-10 (a-d) and ImageNet-20 (e-g) datasets.

	AWA-4	AWA-10	ImageNet-20
Raw feature kernel	47.67 ± 2.22	30.80 ± 1.36	28.20 ± 1.45
Raw feature kernel + MKL	48.50 ± 1.89	31.13 ± 2.81	27.67 ± 1.50
Perturbed semantic kernel tree	N/A	31.53 ± 2.07	28.20 ± 2.02
Semantic kernel tree + Avg	47.17 ± 2.40	31.92 ± 1.21	28.97 ± 1.61
Semantic kernel tree + MKL	48.89 ± 1.06	32.43 ± 1.93	29.74 ± 1.26
Semantic kernel tree + MKL-H	50.06 ± 1.12	32.68 ± 1.79	29.90 ± 0.70
Semantic kernel forest + MKL	49.67 ± 1.11	34.60 ± 1.78	30.97 ± 1.14
Semantic kernel forest + MKL-H	$\mathbf{52.83 \pm 1.68}$	$\mathbf{35.87 \pm 1.22}$	$\mathbf{32.30 \pm 1.00}$

Table 1: Multi-class classification accuracy (and standard errors at 95% confidence intervals) on all datasets, across 5 train/test splits using $30/30/30$ images per class for training/validation/testing.

constructed by varying γ (e.g., [8]). **3) Perturbed semantic kernel tree:** a semantic kernel tree trained on a taxonomy with randomly swapped leaves.

We evaluate several variants of our approach: **1) Semantic kernel tree + Avg:** an average of the kernels from one taxonomy. **2) Semantic kernel tree + MKL:** the same kernels combined with MKL using sparsity regularization. **3) Semantic kernel tree + MKL-H:** adding our hierarchical regularizer (eq. 4). **4) Semantic kernel forest + MKL:** semantic forest kernels from multiple taxonomies combined with MKL. **5) Semantic kernel forest + MKL-H:** adding our hierarchical regularizer.

Results Figure 1 shows the multi-class classification accuracy on all three datasets. Our semantic kernel forests approach significantly outperforms all three baselines, which clearly shows the impact of injecting semantics into discriminative feature learning. The forests' advantage over the individual trees supports our core claim regarding the value of interleaving semantic cues from multiple taxonomies. Further, the proposed hierarchical regularization (MKL-H) outperforms the generic MKL, particularly for the multiple taxonomy forests. This success is *not* simply due to having access to a variety of kernels, as we can see by comparing our method to both the raw feature MKL and perturbed tree results, which use the same number of kernels. Instead, the advantage is leveraging the implicit discriminative criteria embedded in the external semantic groupings. Additional results and explanation are available in our main conference paper [10], and at the project page[3].

4 Conclusion

We proposed a semantic kernel forest approach to learn discriminative visual features from multiple semantic taxonomies, and combine them discriminatively with hierarchical structure-aware regularization. The results show that it improves object recognition accuracy, and give good evidence that committing to a single external knowledge source is insufficient.

There is a long way to go to embed semantic knowledge into discriminative recognition methods. We expect that work in this area can play an important role in the "Big Vision" challenge; exploiting inter-class relationships should improve scalability for many-class problems in recognition in ways that low-level image descriptors will eventually fall short.

[3] `http://www.cs.utexas.edu/~sjhwang/projects/kernelforest/nips12.html`

References

[1] S. Bengio, J. Weston, and D. Grangier. Label Embedding Trees for Large Multi-Class Task. In *NIPS*, 2010.

[2] J. Deng, S. Satheesh, A. Berg, and L. Fei Fei. Fast and balanced: Efficient label tree learning for large scale object recognition. In *NIPS*, 2011.

[3] M. Marszalek and C. Schmid. Semantic hierarchies for visual object recognition. In *CVPR*, 2007.

[4] R. Fergus, H. Bernal, Y. Weiss, and A. Torralba. Semantic label sharing for learning with many categories. In *ECCV*, 2010.

[5] J. Deng, A. Berg, K. Li, and L. Fei-Fei. What does classifying more than 10,000 image categories tell us? In *ECCV*, 2010.

[6] S. J. Hwang, K. Grauman, and F. Sha. Learning a tree of metrics with disjoint visual features. In *NIPS*, 2011.

[7] K. Weinberger, J. Blitzer, and L. Saul. Distance Metric Learning for Large Margin Nearest Neighbor Classification. In *NIPS*, 2006.

[8] F. Bach, G. Lanckriet, and M. Jordan. Multiple Kernel Learning, Conic Duality, and the SMO Algorithm. In *ICML*, 2004.

[9] D. Bertsekas. *Nonlinear Programming*. Athena Scientific, 1999.

[10] S. J. Hwang, K. Grauman, and F. Sha. Semantic kernel forests from multiple taxonomies. In *TO APPEAR, NIPS 2012, Twenty-Sixth Annual Conference on Neural Information Processing Systems*, 2012.

[11] C. Lampert, H. Nickisch, and S. Harmeling. Learning to Detect Unseen Object Classes by Between-Class Attribute Transfer. In *CVPR*, 2009.

[12] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-F ei. Imagenet: A Large-Scale Hierarchical Image Database. In *CVPR*, 2009.