Learning Mid-Level Features from Object Hierarchy for Image Classification

Somayah Albaradei[1,2], Yang Wang[1], Liangliang Cao[3] and Li-Jia Li[4]

[1]Department of Computer Science, University of Manitoba
[2]Department of Computer Science, King Abdulaziz University
[3]IBM T. J. Watson Research Center
[4]Yahoo Research

Abstract

We propose a new approach for constructing mid-level visual features for image classification. We represent an image using the outputs of a collection of binary classifiers. These binary classifiers are trained to differentiate pairs of object classes in an object hierarchy. Our feature representation implicitly captures the hierarchical structure in object classes. We show that our proposed approach outperforms other baseline methods in image classification.

1. Introduction

Image classification is a core research area in computer vision. Most approaches in this area use low-level image representations (e.g. color, texture, shape). For example, the most widely used approach is based on local features, or bag-of-words (BoW) representation. First, local descriptors (e.g. SIFT [17]) are extracted from images. The local descriptors from the training images are clustered to form the visual codebook. Each image is then represented by a BoW representation by counting the frequency of visual words in the image. Finally, a predictive model is learned based on this BoW representation of images.

Although bag-of-words models have been very popular, the visual words used in these models usually have no explicit semantic meanings. So they fail to offer sufficient discriminative power. This is commonly known as the *semantic gap* [12] in visual recognition. To address this issue, some recent work has introduced more semantically meaningful mid-level features for visual recognition. Object Bank [13] and Classme [24] are representative examples in this line of work. Both of these works use feature representations that are directly based on semantic high-level knowledge, e.g. object classes.

Object categories naturally have a hierarchical structure (i.e. taxonomy) with different levels of abstraction. For example, a path in the object hierarchy could be "dog $\rightarrow$ mammal $\rightarrow$ animal $\rightarrow$ living thing". In computer vision, object hierarchy has been used to organize images [14], provide fast run-time algorithms [2], enable novel applications [7]. However, there is little work on using object hierarchy to construct semantic features.

In this paper, we propose a new approach for constructing mid-level feature representation by exploiting the hierarchical structure of object categories. Similar to object bank and classme, our approach uses the output of pre-trained object classifiers as image features. The main difference lies in how those object classifiers are constructed. In our work, we learn a set of one-vs-one binary classifers, where each binary classifier will differentiate between a pair of object categories. We select a pair of object classes if they have the same parent (i.e. they are siblings) in the object hierarchy. Our work is motivated by the following observation. Low-level visual features (e.g. bag-of-words) usually have enough discriminiave power for object categories that are very different in semantic space. But for object categories they are close in semantic space, low-level features are not sufficiently discriminative. By constructing binary classfiers for object classes that are siblings (i.e. close in semantic space) in the object hierarchy, we are effectively learning mid-level features that are likely to overcome the limitation of low-level features.

1.1. Related work

Most image classification methods learn classifiers based on low-level feature representations, e.g. bag-of-words of SIFT descriptors [17], spatial pyramid [11]. The limitation of low-level features is that they do not offer sufficient discriminative power for high-level recognition tasks in computer vision. This is known as the semantic gap.

There has been some recent work on building mid-level image representation for high-level recogition. An example

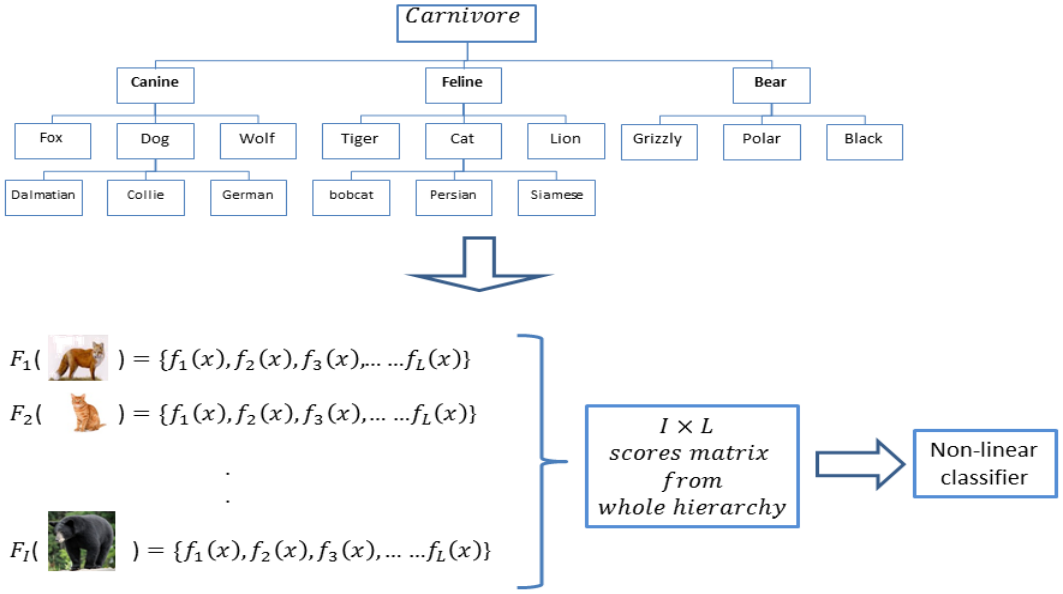

Figure 1. An overview of our approach. Top: we are given a hierarchy of object categories. From this hierarchy, we construct L pairs of object categories and learn a binary classifier for each pair (details in Sec. 3.1). Bottom: for an image, we apply these L binary classifiers and treat the scores of these L binary classifiers as mid-level feature representation of this image. We then learn a non-linear classifier to recognize the image category corresponding to a leaf node in the hierarchy based on this mid-level image representation (details in Sec. 3.2).

is the *object bank* representation [13] for scene classification. In this representation, an image is represented by the response map of a set of pre-trained generic object detectors. An extension of object bank, called *action bank* [20], is proposed to represent complex activities in videos. Similar ideas have been used in the development of the *Classme* descriptor in [24], where an image is represented by the output of a large number of weakly trained object classifiers. There is also a line of work on using attributes as mid-level representations for images [8, 10] and videos [16]. Our work is also related to a line of research on exploiting object taxonomy for recognition. An example of the object taxonomy is the one used in ImageNet [6]. Object taxonomy is particularly useful when dealing with large-scale image classification tasks that involve a large number of object categories, e.g. [5, 4, 15]. Some of the work (e.g. [21]) uses the hierarchical structure of the object taxonomy to deal with object detection for rare objects. It allows rare objects to share visual appearance with common objects. Other work uses the hierarchical structure of object classes to develop efficient sublinear algorithms, e.g. [2, 23]. Object taxonomy has also been used to develop recognition systems that can operate on different levels of abstraction, e.g. [7].

The closest work to ours is the method in Cao et al. [3]. It represents an image by the output of a set of binary classifiers. Each classifier is trained to differentiate between two object classes (i.e. one-vs-one). The main difference between our work and [3] is that we construct the one-vs-one binary classifers by exploit the hierarchical structure of object categories.

1.2. Contribution

In this paper, we propose a new image representation that exploits the hierarchical structure of object classes to build mid-level features. Similar to Cao et al. [3], we represent an image as the scores of a set of binary classifiers. The main difference lies in how we select pairs of object categories for learning the binary classifiers. Given a semantic hierarchy (e.g. a tree-structured taxonomy) of object categories, the method in Cao et al. [3] only considers pairs of object classes at the leaf nodes of the hierarchy. In contrast, our proposed method will exploit the hierarchical structure and construct a richer set of object pairs. Our method considers not only object pairs at the leaf nodes (e.g. cat-vs-dog), but also pairs that correspond to more abstract object categories (e.g. mammal-vs-plant). Our work is motivated by the observation that visual features might have different discriminative power at various levels in the hierarchy. For example, certain features might be useful for differentiating high-level abstract categories (e.g. animal vs plant), while others are useful for more fine-grained object categories (e.g. Shepard dog vs Eskimo dog). By constructing

object pairs at different levels of the hierarchy, we are able to learn a diverse set of discriminative mid-level features.

2. Problem Setup

We assume that we are given a tree-structured semantic hierarchy of object classes. For example, ImageNet [6] uses the WordNet hierarchy to organize all the object classes at many levels of abstractions (also called "synsets" in [6]). Two connected nodes in the hierarchy indicate the "is-a" relationship between them. For example, a "dog" is a "mamanal", an "animal", and a "living thing". Our goal is to classify images into one of the object classes corresponding to the leaf nodes in the hierarchy. Note that in this paper, we only predict the most specific class label, e.g. "dog" (if it is a leaf node in the hierarchy) instead of "living thing". This is different from [7] in which an image can be classified at different levels of abstraction. However, we will use the more abstract classes (e.g. "animal", "living thing") to construct some mid-level feature representation. Since the class labels for the internal nodes of the hierarchy (e.g. "living thing") do not necessarily correspond to concrete objects, we will also use the term "concepts" to refer to an object category in the rest of the paper.

3. Our Approach

An overview of our approach is illustrated in Fig. 1. Given an input image, we first represent the image as a vector of standard low-level features (e.g. color, texture, etc). We then apply a large number of binary classifiers on this low-level feature vector. Each classifier will output a score. We concatenate the scores of all binary classifiers to form a mid-level feature representation of this image. We then apply a non-linear classifier on this mid-level representation to predict the class label. The key of our approach is how to define the binary classifiers used to constructed the mid-level features.

Our image representation is constructed from the responses (i.e. scores) of many binary classifiers. Similar to Cao et al. [3], we learn each classifier to differentiate between one pair of semantically exclusive concepts (one-vs-one). An alternative is to learn a classifier that differentiates between one concept from all other concepts (one-vs-all). As demonstrated by Cao et al. [3], the former (one-vs-one) is preferable, since it is easier for the learning algorithm to find the discriminative features that distinguishes two concepts. While the latter (one-vs-all) is arguably more challenging for the learning algorithm, since it has to learn features that distinguish a concept from a diverse set of other concepts. In our work, we choose the one-vs-one strategy.

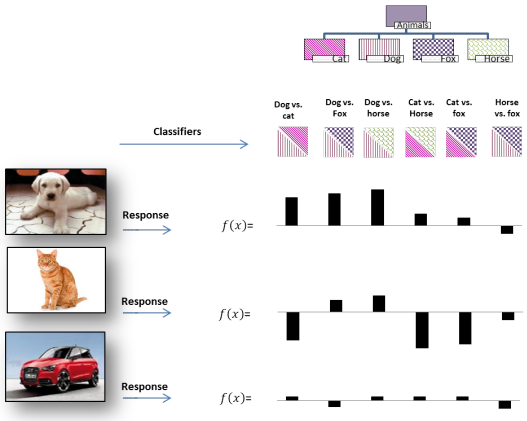

Figure 2. Illustration of how to construct the scores at a single node ("animal") in the hierarchy. We consider each pair between two children of this node, e.g. "cat" vs "dog", "cat" vs "fox", "dog" vs "fox", etc. For each pair, we learn a binary classifier differentiating the two object categories. The outputs of these classifiers are the mid-level features constructed at this node.

3.1. Selecting concept pairs

Suppose we have a set of K concepts $\mathcal{C} = \{C_1, C_2, ..., C_K\}$. These concepts correspond to the leaf nodes in the object hierarchy. The method in [3] constructs $\frac{K(K-1)}{2}$ concept pairs by considering every possible combination of two concepts. For each concept pair (e.g. "bicycle" vs "bus"), it then learns a linear SVM classifier that differentiates these two concepts.

We propose a new way of constructing concept pairs by exploiting the semantic hierarchical structure of object categories. Let us consider a non-leaf node V in the hierarchy, e.g. V might correspond to the concept "animal". Suppose the node V has t child nodes. The child nodes of "animal" might correspond to concepts such as "dog", "cat", "horse", etc. For the node V, we construct $\frac{t(t-1)}{2}$ concept pairs by choosing all pairs of concepts from the child nodes of V. For example, the concept pairs for "animal" will include "dog-vs-cat", "dog-vs-horse", "cat-vs-horse", etc.

We repeat the process for all internal nodes in the hierarchy. In this end, we will get a collection of concept pairs. Some of the concept pairs will correspond to abstract concepts, such as "animal-vs-plant". Others will corresponds to fine-grained concepts, such as Shepard dog vs Eskimo dog.

We then use standard techniques to learn a classifier to differentiates the two concepts in each concept pair. We first extract some standard low-level image features from the images, such as color histogram, SIFT histogram [17],

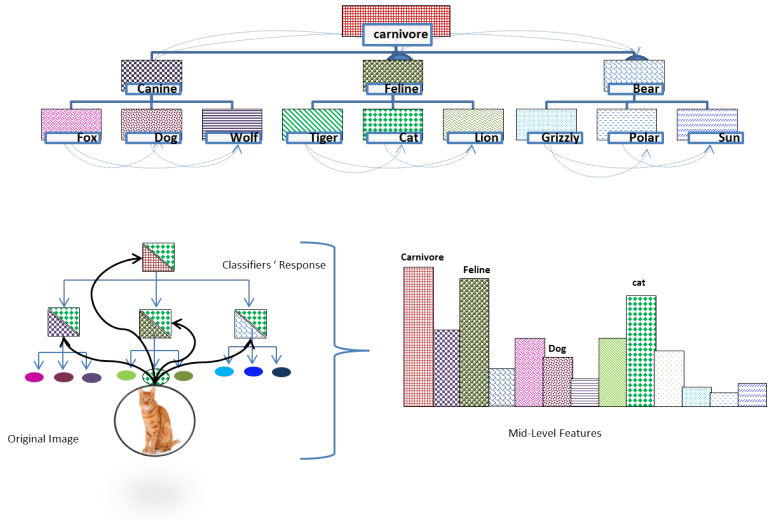

Figure 3. Illustratio of how to construct the mid-level feature from the entire hierarchy for **I** images. We apply the process illustrated in Fig. 2 to every internal node of the hierarchy. Each internal node will give a set of scores. The concatenation of scores across all internal nodes forms our mid-level feature representation.

GIST [19], LBP [1], etc. We will describe in detail the low-level features we used on each dataset in the experiment later (Sec. 4). For a concept (e.g. "dog") in the hierarchy, its training examples are those labeled as descendants of this concept. In other words, all the images that are labeled as specific species of dogs in the training data will be considered as "dog" images. We then learn a binary linear SVM classifier for each concept pair. We represent the classifier for the i-th concept pair as f_i. Given a new input image $\mathbf{x}$, this classifier will return a score $f_i(\mathbf{x})$. We can interpret this score as the confidence of differentiating $\mathbf{x}$ between the positive/negative class in the i-th concept pair. If f_i corresponds to the "dog-vs-cat" concept pair with the "dog" being the positive class, we would expect $f_i(\mathbf{x})$ to be high if $\mathbf{x}$ is an image of a dog. Similarly, we would expect a low score it is an image of a cat, and a score close to zero if it is neither a dog or a cat image. Figs. 2 and 3 illustrate the whole process.

3.2. Image representation

The method in Sec. 3.1 gives us a collection of L binary classifiers f_1, f_2, ..., f_L. We can interpret these binary classifiers as defining a L-dimensional semantic concept space. Due to the way we construct these binary classifiers, this concept space encodes information about the hierarchical structure of object categories.

For an image $\mathbf{x}$, we encode this image as a L-dimenaional vector $F(\mathbf{x})$ by concatenating the scores of these binary classifiers applied on $\mathbf{x}$, i.e. $F(\mathbf{x}) = [f_1(\mathbf{x}), f_1(\mathbf{x}), ..., f_L(\mathbf{x})]$. We treat $F(\mathbf{x})$ as the mid-level feature representation of the image $\mathbf{x}$. Using this L-dimensional feature representation on training data, we then learn a K-class nonlinear SVM classifier to predict the class label of any given image.

During testing, we are given an unseen image $\mathbf{x}$. Similarly, we encode $\mathbf{x}$ using the L binary classifiers as $F(\mathbf{x}) = [f_1(\mathbf{x}), f_1(\mathbf{x}), ..., f_L(\mathbf{x})]$. We then apply the learned K-class kernel SVM to predict the class label of this new image.

4. Experiments

4.1. Datasets

We evaluate our proposed approach on four publicly available datasets. On each dataset, we randomly choose 90% of the examples for training the remaining ones for testing. For simplicity, we use the pre-computed low-level features that come with each of the dataset; such as, color histogram, SIFT histogram, LBP, GIST, and others. But it is important to note that our proposed approach can be used together with any low-level features.

Datasets	ImageNet65	AwA	CIFAR	Yahoo Shoes
Raw features	23.8%	23.1%	25.7%	61.1.7%
Cao et al. [3]	29.7%	24.5%	28.6%	62.4%
Ours	**36.2%**	**26.5%**	**30.5%**	**64.7%**

Table 1. Comparison of overall accuracies of our approach with two baseline methods on three datasets.

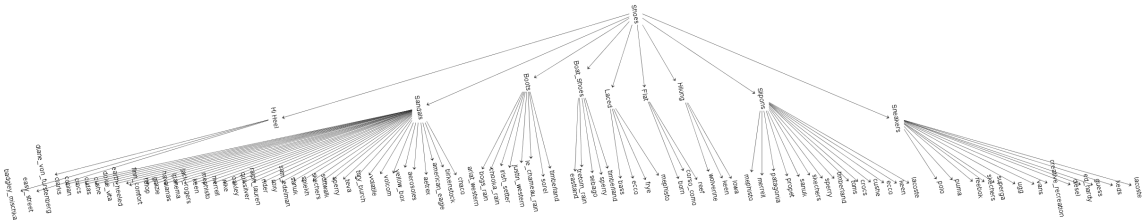

Figure 4. (Best viewed with PDF magnification) Hierarchy of Yahoo Shoes datasets used in our experiment

ImageNet65 [6]: this dataset is a subset of the ImageNet [6]. It is constructed from the "plant" "animal" and "vehicle" subtrees in ImageNet. This dataset contains 39600 images of 65 leaf nodes.

Animal-with-Attributes (AwA) [10]: this dataset consists of 30475 images of 50 different animal classes. Since this dataset does not come with a hierarchy, we build the hierarchy ourselves using the WordNet.

CIFAR [9]: this dataset consists of 60000 images of 100 object classes. Each object class belongs to one of the 20 *superclasses*. For example, the "fish" superclass contains 4 object classes: aquarium fish, flatfish, ray, shark, trout. We build a two-layer hierarchy according to this superclass relation. In other words, the hierarchy has 20 internal nodes corresponding to the superclasses and 50 leaf nodes corresponding to the object classes.

Yahoo Shoes [26]: this dataset consists of 5250 images of 107 shoes classes. Each shoes class belongs to one of the 10 *superclasses*. For example, the "Boots " superclass contain 13 shoes classes(e.g): Ariat-Westren boots, Boges-Rain boots, Timber-Land boots , Justin-Western boots...,etc. We build a two-layer hierarchy according to this superclass relation. In other words, the hierarchy has 10 internal nodes corresponding to the superclasses and 107 leaf nodes corresponding to the shoes classes. The hierarchy of this datasets is shown in Fig. 4. We extract several features for representing both local and global features. For example, we extract: local features using SIFT histogram [18], color features using color histogram [25], texture features using Local Binary Pattern [22], and global features using GIST [19].

4.2. Experimental results

We compare our method with the following two baselines.

1) Raw features: this baseline method learns a nonlinear kernel SVM based on the raw low-level features.

2) Cao et al. [3]: this baseline approach is the method in [3]. It is similar to our method. The difference is that it only consider pairs of concepts from leaf nodes. So it ignores the hierarchical structure of the object classes.

The overall accuracies of these two baselines are shown in the 1st and 2nd rows of Table 1. The accuracies of our method is shown in the 3rd row of Table 1. We can see that our proposed method outperforms the two baseline approaches. example predictions of our method and Cao et al. [3] are shown in Fig. 5.

5. Conclusion

We have introduced a new mid-level feature representation for image classification. Our representation uses the output of a set of binary classifiers as the feature vector of an image. Each binary classifier is trained to differentiate between a pair of object classes in the object hierarchy. Our experimental results have shown the effectiveness of our approach compared with other methods in the literature. In the future, we would like to reduce the complexity of our proposed method as it grow exponentially with the number of leaves in the hierarchy.

Acknowledgment

A special thanks for King Abdulaziz University for their generous support of Somayah Albaradei.

References

[1] T. Ahonen, A. Hadid, and M. Pietikainen. Face description with local binary patterns: Application to face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(12):2037–2041, 2006. 4

[2] S. Bengio, J. Weston, and D. Grangier. Label embedding trees for large multi-class tasks. In *Advances in Neural Information Processing Systems*. MIT Press, 2010. 1, 2

[3] L. Cao, L. Gong, J. R. Kender, N. C. Codella, and J. R. Smith. Learning by focusing: A new framework for concept recognition and fea-

		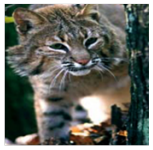				
Cao et al. [3]	Bear	Tiger	Dog	Bed	Bird	sweet peppers
Ours	**Beaver**	**Bobcat**	**Fox**	**Television**	**speedboat**	**Can**

Figure 5. Some example predictions of our method and Cao et al. [3].

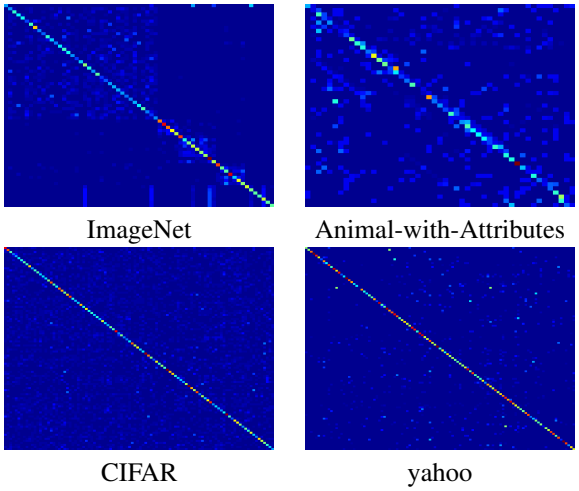

Figure 6. Confusion matrices of our method on four datasets.

ture selection. In *IEEE International Conference on Multimedia and Expo*, 2013. 2, 3, 5, 6

[4] J. Deng, A. C. Berg, and L. Fei-Fei. Hierarchical semantic indexing for large scale image retrieval. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2011. 2

[5] J. Deng, A. C. Berg, K. Li, and L. Fei-Fei. What does classifying more than 10,000 image categories tell us? In *European Conference on Computer Vision*, 2010. 2

[6] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A large-scale hierarchical image database. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2009. 2, 3, 5

[7] J. Deng, J. Krause, A. C. Berg, and L. Fei-Fei. Hedging your bets: Optimizing accuracy-specificity trade-offs in large scale visual recognition. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2012. 1, 2, 3

[8] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth. Describing objects by their attributes. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2009. 2

[9] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, 2012. 5

[10] C. H. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2009. 2, 5

[11] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognition natural scene categories. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2006. 1

[12] L.-J. Li, H. Su, Y. Lim, and L. Fei-Fei. Objects as attributes for scene classification. In *Trends and Topics in Computer Vision*, pages 57–69. Springer, 2012. 1

[13] L.-J. Li, H. Su, E. P. Xing, and L. Fei-Fei. Object bank: A high-level image representation for scene classification and semantic feature sparsification. In *Advances in Neural Information Processing Systems*. MIT Press, 2010. 1, 2

[14] L.-J. Li, C. Wang, Y. Lim, D. M. Blei, and L. Fei-Fei. Building and using a semantivisual image hierarchy. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010. 1

[15] Y. Lin, F. Lv, S. Zhu, M. Y, T. Cour, K. Yu, L. Cao, and T. Huang. Large-scale image classification: Fast feature extraction and SVM training. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2011. 2

[16] J. Liu, Q. Yu, O. Javed, S. Ali, A. Tamrakar, A. Divakaran, H. Cheng, and H. S. Sawhney. Video event recognition using concept attributes. In *IEEE Workshop on Applications of Computer Vision*, 2013. 2

[17] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004. 1, 3

[18] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004. 5

[19] A. Oliva and A. Torralba. Modeling the shape of the scene: a holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42(3):145–175, 2001. 4, 5

[20] S. Sadanand and J. J. Corso. Action bank: A high-level representation of activity in video. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2012. 2

[21] R. Salakhutdinov, A. Torralba, and J. Tenenbaum. Learning to share visual appearance for multiclass object detection. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2011. 2

[22] C. Shan, S. Gong, and P. W. McOwan. Facial expression recognition based on local binary patterns: A comprehensive study. *Image and Vision Computing*, 27(6):803–816, 2009. 5

[23] M. Sun, W. Huang, and S. Savarese. Finding the best path: an efficient and accurate classifier for image hierarchies. In *IEEE International Conference on Computer Vision*, 2013. 2

[24] L. Torresani, M. Szummer, and A. Fitzgibbon. Efficient object category recognition using classmes. In *European Conference on Computer Vision*, 2010. 1, 2

[25] J. Xiao, J. Hays, K. A. Ehinger, A. Oliva, and A. Torralba. SUN database: Large-scale scene recognition from abbey to zoo. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010. 5

[26] Yahoo research labs. Yahoo! Shopping Shoes Image Content, 2013. 5