Furniture-Geek: Understanding Fine-Grained Furniture Attributes from Freely Associated Text and Tags

Vicente Ordonez[†], Vignesh Jagadeesh[‡], Wei Di[‡], Anurag Bhardwaj[‡], Robinson Piramuthu[‡]
[†]University of North Carolina at Chapel Hill [‡]eBay Research Labs
vicente@cs.unc.edu,[vjagadeesh, wedi, anbhardwaj, rpiramuthu]@ebay.com

Abstract

As the amount of user generated content on the internet grows, it becomes ever more important to come up with vision systems that learn directly from weakly annotated and noisy data. We leverage a large scale collection of user generated content comprising of images, tags and title/captions of furniture inventory from an e-commerce website to discover and categorize learnable visual attributes. Furniture categories have long been the quintessential example of why computer vision is hard, and we make one of the first attempts to understand them through a large scale weakly annotated dataset. We focus on a handful of furniture categories that are associated with a large number of fine-grained attributes. We propose a set of localized feature representations built on top of state-of-the-art computer vision representations originally designed for fine-grained object categorization. We report a thorough empirical characterization on the visual identifiability of various fine-grained attributes using these representations and show encouraging results on finding iconic images and on multi-attribute prediction.

1. Introduction

The vision community has relied on crowdsourcing for human supervision in several image understanding tasks like scene understanding [28], object recognition [10] and human pose estimation [5]. As computer vision systems begin recognizing object categories in the scale of thousands [9] or even hundreds of thousands [8], it is very difficult to make crowdsourcing scale for those scenarios. Recent interest in attribute-based methods for representation [14, 18, 22, 30] and fine-grained object categorization [23, 32, 33] make the annotation task for supervised learning even more expensive. We avoid explicitly asking users to annotate images by using text cues such as tags/titles readily available on e-commerce websites where sellers, with first hand knowledge of their inventory, tag and describe images.

Figure 1: Iconic images for six sample furniture attribute categories. These are sample images for which our individual attribute predictors output a high confidence score.

Another interesting dimension of our problem is dealing with furniture object categories. Gibson [15] states that objects are not defined only by shapes or appearance but also their affordances, which are the object possibilities for actions. This makes the detection of furniture objects like *chair* more difficult because, due to their functional nature, these objects exhibit high intra-class variation. This is also evidenced by the last Pascal VOC Challenge results [13] where even highly deformable objects like *cats* are detected with higher precision than *chairs*. Some authors have even proposed to go as far as to drop appearance modeling completely and rather focus on the affordance detection problem itself [11, 16]. We propose instead to use a large set of fine-grained visual attributes to characterize and better understand furniture categories and deal with this variation in appearance. To our knowledge, this is the first attempt at analyzing large scale attributes for furniture images. We show some iconic examples discovered by our furniture-specific

Figure 2: Sample images named as *accent chairs* by user descriptions vs images named just as *chairs*.

visual attribute predictors in Fig 1.

Mining visual attributes from freely associated descriptions or tags in the wild (in uncontrolled settings) will lead to noisy and imperfect annotations. Yet, this will potentially produce knowledge that might prove difficult or relatively expensive to obtain from users in a crowdsourcing platform. For instance, it is hard to assess which visual features might indicate that a *chair* is an *accent chair*. A quick search reveals the following definition:

> "Accent chair: An accent chair can be used to pick up on a highlight color within the theme of a room adding visual interest and pulling a color scheme together. The accent chair is often a different style, is not part of a suite of furniture and is often upholstered in a different, patterned fabric than the rest of the furniture in the room." [1]

While the object is still mostly defined based on its function, given the definition, a human could reasonably guess what kind of chairs might be better candidates for *accent chairs*, given solely an image. Some of the attributes are visual or at least somewhat visual, e.g. *upholstered*, *adding visual interest*, *patterned fabric*. See Figure 2.

With all the previous considerations, we present a system that a) Takes as input unstructured (title/captions or descriptions) and semi-structured (tags) data as noisy image annotations (section 2) b) Takes advantage of the catalog image assumption where images are biased towards the center of the picture (section 3). c) Discovers and learns visual attribute models from such input (section 4), d) Produces highly specialized furniture-specific tag suggestions on novel images and discovers iconic images for fine-grained furniture attributes (section 6).

In summary, our contributions are three-fold:

- Sidestepping crowdsourcing by utilizing noisy text and tags as a proxy.
- The design of a recognition system for fine-grained attributes of furniture.
- Thorough empirical analysis of visual identifiability of a large set of visual attributes.

1.1. Related work

Our paper relates to several other lines of work: Attribute and category discovery from unstructured and semi-structured annotations [2, 7, 21, 27]. Toderici *et al*. [27] mine category names from a large pool of semi-structured data (user tags) but their main focus is on audiovisual data. Berg *et al*. [2] introduced the construction of vocabularies of attributes from purely unstructured data (descriptions), we extend this to the mixed scenario of structured and semi-structured data and use annotation-specific semantics to define hard negative examples (sec. 4). Parikh and Grauman [21] combine iterative learning of potentially meaningful visual models with human annotators in the loop. The more recent work of Crowley *et al*. [7] is similar to ours in the sense that it learns from brief text descriptions but it does so for the analysis of ancient vases and focuses on a more specific set of attributes. In contrast, our approach tries to learn as many attributes as possible, regardless of type of attributes or whether they are detectable. A slightly different line of work seeks to understand images and text using generative models [4, 12] where both images and text are considered annotations. With a practical application in mind, we take the approach of the more recent works of [2, 21, 27] and adopt a discriminative framework in favor of potentially improved performance and speed, especially given the scale of the problem and noise in our data. One disadvantage of the discriminative approach is that relationships between the variables in the input and output domains are not explicitly modeled.

2. Analyzing e-commerce data

We collected approximately 120,000 images of furniture with associated title descriptions and user supplied tags (when available) from an e-commerce application. There are a total of 22 furniture categories spanning from *tables* (9827 images) to *vanities&makeup tables* (857 images) [2]. We also selected the images preferring those coming from top sellers so that we can get richer descriptions from users who are more likely to be domain experts. This will allow us to provide tag suggestions to beginner users using the knowledge of more advanced users.

[1] Ali McCulloch, eHow Contributor. Retrieved on Aug/2013 http://www.ehow.com/info_8300531_accent-chair.html

[2] *tables*, *chairs*, *entertainment units & tv stands*, *desks & home office furniture*, *dining sets*, *bar stools*, *beds & bed frames*, *cabinets & cupboards*, *benches & stools*, *bookcases*, *dressers & chests of drawers*, *ottomans footstools & poufs*, *bedroom sets*, *screens & room dividers*, *nightstands*, *trunks & chests*, *bean bags & inflatables*, *futons frames & covers*, *sideboards & buffets*, *cd & video racks*, *armories & wardrobes*, *vanities & makeup tables*

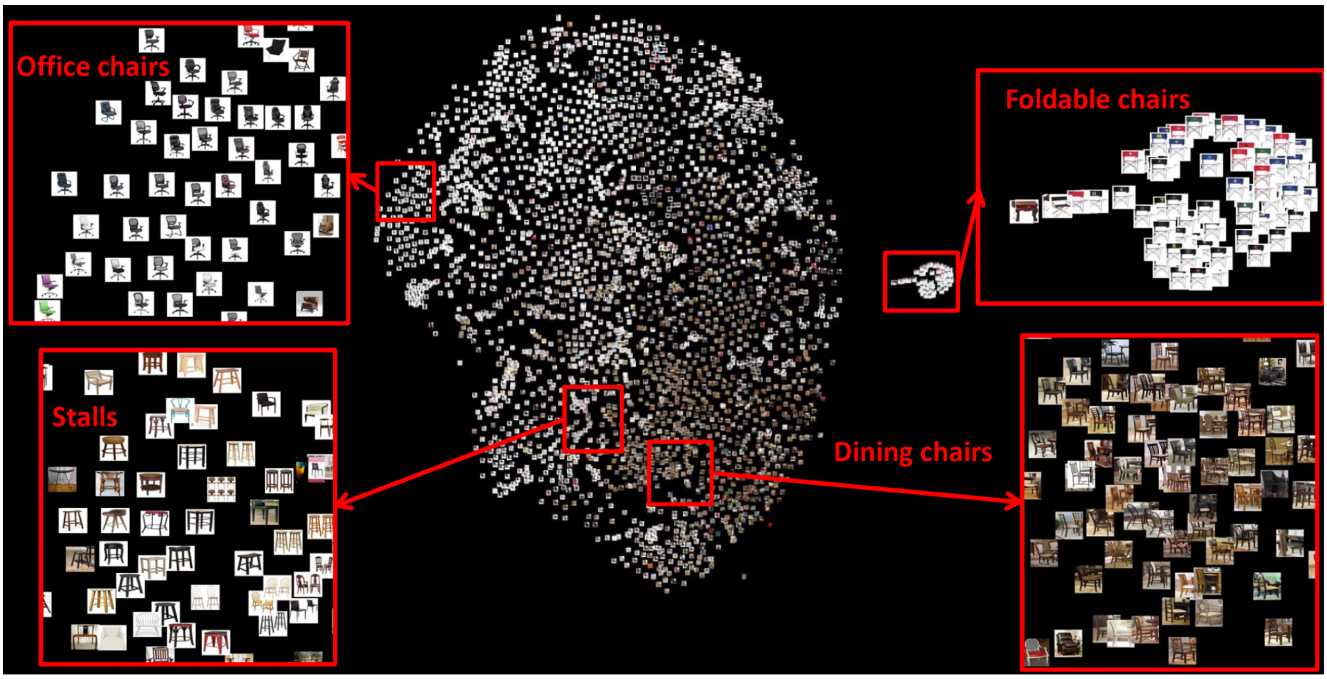


Figure 4: Visualization of a subset of chair images from our data using the tSNE [29] embedding technique using visual features for representation.

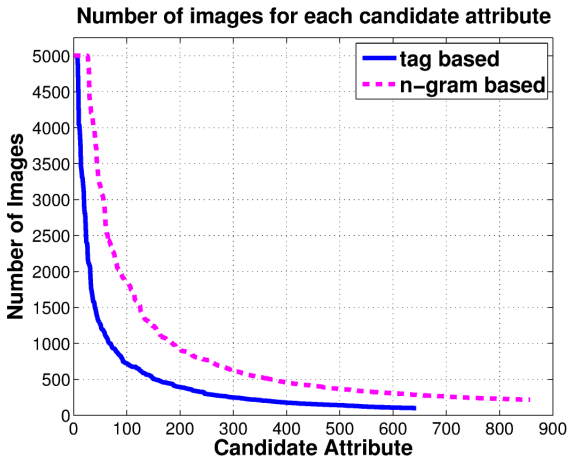


Figure 3: Number of images for each candidate attributes: tag-based attributes in solid blue and n-gram based attributes in dashed magenta.

Every image has a title description and roughly 80,000 images have at least one tag-value pair. The set of tags is very rich with a total of 323 unique tag-value pairs with at least 300 images each. To provide some structure to title descriptions we compute all possible n-grams up to 3-grams and count the occurrence of each across the entire data. We end up with 576 n-grams each associated with at least 300 images. We consider any of those attributes (323 + 576) as potential visual attributes and we measure the visual identifiability based on held out test data for each attribute. We also cap the maximum number of images for each candidate attribute to 5000.

To have a better idea of the scale of this data we show in Figure 3 the number of images we use for each candidate binary attribute of each type. We observe that both sources of annotations follow a power-law long tailed distribution, typical of this kind of data. We effectively alleviate this high imbalance in our discriminative framework by specifying a reasonable amount of negative samples based on the available amount of possible samples (section 4).

We work under the assumption that text and image features are correlated, we use the t-Distributed Stochastic Embedding (tSNE) technique of van der Maaten and Hinton [29] for visualizing a large subset of our data using standard GIST [20] visual features (Figure 4). We observe that images do not necessarily cluster into clearly defined groups but rather follow a smooth transition into different types of chairs, highlighting the potential for attribute-based recognition where we not only try to categorize, but also characterize objects based on individual traits.

3. Feature Representation

We use three different types of feature representations: Dense SIFT, Grabcut Localized Dense SIFT and Grabcut Localized Color.

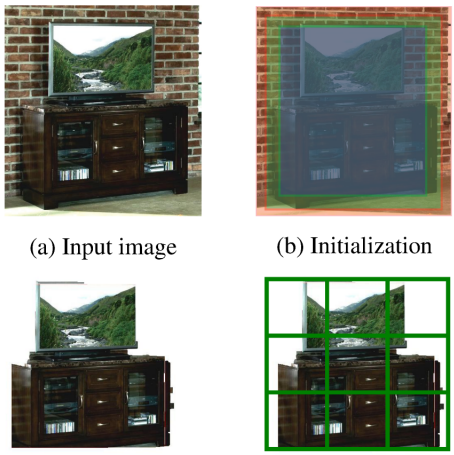
Dense SIFT: We use the bag-of-visual words feature representations with a combination of non-linear encoding

and spatial binning. Recent benchmarks on feature representations show that dictionary size and appropriate feature encodings are crucial for improved performance [6], moreover they often outperform or are comparable in performance to other methods relying on higher level image representations [17]. We use SIFT features [19] computed on a regular grid at three different scales using a codebook of 10,000 descriptors and assign visual words using Locality-constrained linear coding (LLC) [31] with $knn = 5$. We use two levels for spatial pooling: over the entire image and on a 3x3 grid covering the entire image extents.

Grabcut Localized Dense SIFT: For some contextual attributes like the *bedroom setting*, the global image contents beyond the object of interest might be helpful. For a lot of other attributes the background will act as a distractor. In principle, the discriminative framework should be able to discount the features from the background but in practice it still hinders performance on a sizable set of attributes as evidenced in our evaluation section (sec 5). Additionally, even though the bag-of-words model assumes orderless features, the spatial pooling step assumes at least a coarse degree of registration.

We rely on the popular Grabcut algorithm [26] that is able to separate background from foreground even from a very weak initial labeling. In our case we define two rectangular areas, one covering 70% and another covering 90% of the image (see Figure 5b). The innermost region defined by those rectangles (in blue shade) is labeled as probably foreground, the outermost region (in red shade) is labeled as definitely background and the region in-between (in green shade) is labeled as probably background. We use the foreground region in two ways: a) To constrain the spatial pooling to the rectangle circumscribing the foreground mask and b) To sample the LLC codes that fall within the foreground mask only. This scheme improves the performance of the overall discovery process for at least 27% of 323 tag-based attributes and 17% of a 576 ngram-based attributes for which we have more than 300 test images (see sec 5).

Grabcut Localized Color: Color features and color-naming patterns are an interesting case of language grounding. While recognizing images containing a given color might seem trivial, predicting when a user will name some particular object as having certain color is a different problem. E.g. a white object might be named as *white*, but in the presence of a *red* feature, people are more likely to name it as *red*. There are dominant colors and there might even be other biases about the location at which colors appear. We experimented with computing both global color features and localized color features. We also experimented with several variations of color representations, we chose an illumination invariant color histogram [3]. We found that better localization with simpler color representations seem to fare better than more complex representations computed globally. We tested both methodologies on a small scale experiment that automatically constructs visual palettes from color-attributes and decided to adopt the latter.



(a) Input image (b) Initialization

(c) Foreground (d) Spatial pooling

Figure 5: Grabcut based foreground extraction. In the initialization step we label the outermost region (in red) as definitely background, the innermost region (blue) as probably foreground and what lies in between (green) as probably background.

Color features did not improve much the overall performance because they were only successful in predicting some color words (e.g. *green*, *red*, *blue*, *cream*) or some material words (e.g. *black leather*, *ivory*) which represent only a small fraction of our set of attributes. Some color words were even predicted reasonably well using appearance which means that further analysis on combinations of appearance and color might be required to predict this kind of categories. In summary, our color features improved performance on 33 out of 323 tag-based attributes and 11 out of 576 ngram-based attributes, mostly those attributes related to color words.

4. Attribute Discovery and Categorization

Linear SVMs are trained for each potential visual attribute from our pool of 899 candidate binary attributes described in section 2 (we binarize multi-valued attributes obtained from tags for uniformity since n-gram mined attributes are always binary). We rely on the non-linearity of our feature encoding (LLC) so that we can avoid using the more expensive nonlinear SVMs while retaining comparable discriminative power. In this way, we learn a relatively large set of models and discard the ones that do not seem to be useful based on performance on a validation set.

Since we are dealing with real world data, the amount of images that we have for each category follow a long-tail

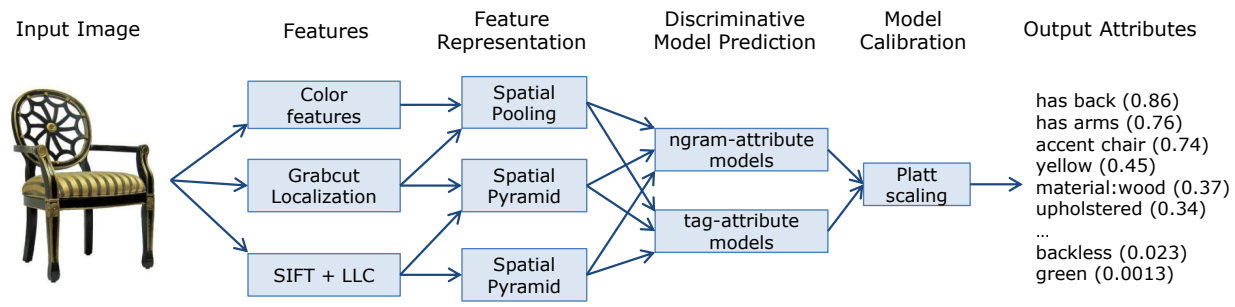


Figure 6: General overview of our proposed attribute estimation system based on tag and text annotations.

power-law distribution (Figure 3) where there exists a lot of images for a few categories and fewer images for a large number of categories, we use up to 5,000 positive images for each attribute type.

Another challenge is to define a set of negative examples and the number of negative examples. For n-gram attributes we use the closed world assumption, where we assume that any image that does not contain a given n-gram attribute is considered a potential negative example for that n-gram attribute. For tag attributes that are multi-valued we choose negative examples based on the complements of the multi-valued attribute, e.g. we know that the negative examples of *leather material* are things like *ivory material*, *plastic material*, *metallic material*. If we do not have enough such negative examples (4 times more than positive examples) for a given multi-valued attribute we mine negatives randomly from other attribute categories.

5. Experimental Evaluation

We make several comparisons of our models from Section 4. We keep non-overlapping training data, validation data and a test data for each attribute. We present on Table 1 the percentage of attributes for which each type of feature representation (LLC + SIFT, Grabcut + LLC + SIFT and Grabcut+Color) performed better on test data.

Attribute type	SIFT + LLC	Grabcut + SIFT + LLC	Grabcut + Color
Tag attributes	63%	27%	10%
Ngram attributes	81%	17%	2%

Table 1: Percentage of attribute categories where each feature performs the best. Total tag attributes used for evaluation: 323, total n-gram attributes used for evaluation: 576.

We also compare individual attribute categorization performance. Since we have very different data set size for each visual attribute, we quantify the number of errors made by the classifier at a given recall regime. Figures 7a and 7b show the average number of errors across all classifiers at 5 recall values (10% to 50%). We use validation data to decide what is the best feature representation to use for each attribute and use this to report a combined performance by averaging across the errors of the best performing models. We do not report average error for color features since, in general, they perform poorly on non-color attributes.

We finally conduct an experiment to have an idea of the set of attributes that can actually be predicted reliably or are visually identifiable. Our test data is distributed similar to our training data, containing at least 4 times as many negative examples compared to positive examples. Figure 7c reports the number of attributes for which we can perform significantly better than random chance precision for different recall regimes, i.e. at 50% recall we can identify reliably almost 100 attribute types.

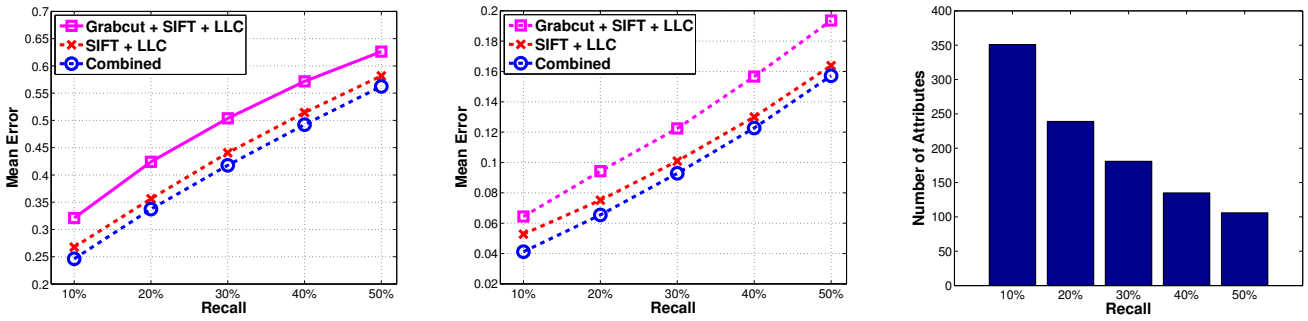
6. Applications

We present two applications of our system: A generic tag recommendation system using multi-attribute prediction and an application on iconic image discovery. Both tasks are from the perspective of furniture data.

6.1. Multi-attribute prediction

We apply the parametric models of visual attributes that were learned independently in Section 4. For this application, we also add a calibration step for each of our SVMs to obtain a well calibrated probabilistic output. We fit a sigmoid using Platt scaling [24] independently for each model on a small non-overlapping validation set with a size of 50% of the number of images used for training for each attribute.

Figure 6 illustrates the entire system workflow for multi-attribute prediction for a given image. We also present qualitative results on a separate set of images of furniture and show the top attribute tags for each in Figure 9. We include here results before calibration and after calibration using Platt scaling. The calibration step also takes care of attributes which are not easily visually identifiable or which overfit to a particular type of data by consistently making those attribute models output very small probabilities. So, while not all visual attributes can be detected reliably, we do not need to explicitly drop any attributes *a priori*.



(a) Tag Attributes (b) Title Attributes (c) Attribute identifiability

Figure 7: Collective evaluation of individual attribute predictors collected from tags (a) and title descriptions (b) using both localized and whole image features. (c) shows the number of attribute categories that can be identified with a minimum reliability (10x better than random chance on validation data) at different recall regimes.

For quantitative analysis of multi-attribute prediction, we picked 100 new furniture images at random. For half of these images Grabcut was able to reasonably segment foreground from background. We retrieved the top 10 predicted attributes based on their calibrated score for each image. There were about 300 unique attributes across all image predictions, out of which 231 predicted attributes were common to both approaches. The overall accuracy for common attributes was 40.7% and 38.1% respectively for SIFT + LLC and Grabcut + SIFT + LLC. Considering only images where the Grabcut segmentation was reasonable, the overall accuracy was 40.8% and 42.8% respectively. Thus, on average, Grabcut + SIFT + LLC performed better when the Grabcut segmentation was reasonable, and SIFT + LCC performed better otherwise.

6.2. Iconic image discovery

Another application that has found ground recently in the vision literature is that of iconicity or finding prototypical images for a given visual concept [1, 25]. Iconic images are the ones that better represent a given object category or attribute. They are potentially useful for applications in graphics, aesthetics, retrieval and advertising. Iconic images are a natural extension of our models trained in section 4, we simply score a set of images of attribute type A using the model learned on a set of images with attribute type A and pick the top scored images as our iconic set. We show iconic images for several categories in Figure 8.

7. Conclusions

We could learn reliable models for a large number of visual attributes for challenging furniture categories. Mining attributes from freely associated text descriptions from seller titles augmented the amount and quality of image annotations significantly. Performing weak localization for feature representation together with global image features increases overall performance when categorizing attributes for images of furniture. We finally show encouraging results on two direct applications of this approach.

References

[1] T. Berg and A. Berg. Finding iconic images. In *CVPR Workshop on Internet Vision 2009*, pages 1–8, June 2009. 6

[2] T. L. Berg, A. C. Berg, and J. Shih. Automatic attribute discovery and characterization from noisy web data. In *ECCV*. 2010. 2

[3] A. Bhardwaj, A. D. Sarma, W. Di, R. Hamid, R. Piramuthu, and N. Sundaresan. Palette power: Enabling visual search through colors. In *KDD*, 2013. 4

[4] D. M. Blei and M. I. Jordan. Modeling annotated data. In *SIGIR*, 2003. 2

[5] L. Bourdev and J. Malik. Poselets: Body part detectors trained using 3d human pose annotations. In *CVPR*, 2009. 1

[6] K. Chatfield, V. Lempitsky, A. Vedaldi, and A. Zisserman. The devil is in the details: an evaluation of recent feature encoding methods. In *BMVC*, 2011. 4

[7] E. J. Crowley and A. Zisserman. Of gods and goats: Weakly supervised learning of figurative art. *BMVC*, 8:14, 2013. 2

[8] T. Dean, M. A. Ruzon, M. Segal, J. Shlens, S. Vijayanarasimhan, and J. Yagnik. Fast, accurate detection of 100,000 object classes on a single machine. In *CVPR*, 2013. 1

[9] J. Deng, A. C. Berg, K. Li, and L. Fei-Fei. What does classifying more than 10,000 image categories tell us? In *ECCV*. 2010. 1

[10] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 1

[11] C. Desai and D. Ramanan. Predicting functional regions of objects. In *CVPR Workshops*, 2013. 1

[12] P. Duygulu, K. Barnard, J. F. de Freitas, and D. A. Forsyth. Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In *ECCV*. 2006. 2

[13] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes

Figure 8: Here we present the most iconic images for several attribute categories. Note the semantic similarity between mined functional attributes like *hutch* and *curio*, a *hutch* is a cupboard with drawers for storage on top and a *curio* is a type of cabinet often used to display collector's items.

Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html. 1

[14] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth. Describing objects by their attributes. In *CVPR*, 2009. 1

[15] J. J. Gibson. *The ecological approach to visual perception*. Routledge, 1986. 1

[16] H. Grabner, J. Gall, and L. Van Gool. What makes a chair a chair? In *CVPR*, 2011. 1

[17] M. Juneja, A. Vedaldi, C. Jawahar, and A. Zisserman. Blocks that shout: Distinctive parts for scene classification. In *CVPR*, 2013. 4

[18] C. H. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *CVPR*, 2009. 1

[19] D. G. Lowe. Object recognition from local scale-invariant features. In *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, volume 2, pages 1150–1157. Ieee, 1999. 4

[20] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *IJCV*, 42(3):145–175, 2001. 3

[21] D. Parikh and K. Grauman. Interactively building a discriminative vocabulary of nameable attributes. In *CVPR*, 2011. 2

[22] D. Parikh and K. Grauman. Relative attributes. In *ICCV*, 2011. 1

[23] O. M. Parkhi, A. Vedaldi, A. Zisserman, and C. Jawahar. Cats and dogs. In *CVPR*, 2012. 1

[24] J. Platt et al. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74, 1999. 5

[25] R. Raguram, C. Wu, J.-M. Frahm, and S. Lazebnik. Modeling and recognition of landmark image collections using iconic scene graphs. *IJCV*, 95(3):213–239, 2011. 6

[26] C. Rother, V. Kolmogorov, and A. Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. In *ACM Transactions on Graphics (TOG)*, volume 23, pages 309–314. ACM, 2004. 4

[27] G. Toderici, H. Aradhye, M. Pasca, L. Sbaiz, and J. Yagnik. Finding meaning on youtube: Tag recommendation and category discovery. In *CVPR*, 2010. 2

[28] A. Torralba, B. C. Russell, and J. Yuen. Labelme: Online image annotation and applications. *Proceedings of the IEEE*, 2010. 1

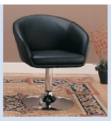
[29] L. van der Maaten and G. Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605, 2008. 3

[30] G. Wang and D. Forsyth. Joint learning of visual attributes, object classes and visual saliency. In *CVPR*, 2009. 1

[31] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong. Locality-constrained linear coding for image classification. In *CVPR*, 2010. 4

[32] P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, and P. Perona. Caltech-ucsd birds 200. 2010. 1

[33] B. Yao, A. Khosla, and L. Fei-Fei. Combining randomization and discrimination for fine-grained image categorization. In *CVPR*, 2011. 1

Image	SIFT+LLC (before calibration)		SIFT+LLC (after calibration)		Image+Grabcut	Grabcut +SIFT+LLC (before calibration)		Grabcut +SIFT+LLC (after calibration)	
	bar	1.27	bar stools	0.97		walnut	1.08	bar stools	0.93
	bar stools	1.07	bar	0.96		bar	0.98	bar	0.92
	black	0.75	rta	0.92		bar stools	0.80	unfinished	0.92
	kitchen	0.74	chrome	0.91		solid	0.70	hillsdale	0.86
	chrome	0.69	kitchen	0.82		oak	0.70	rta	0.86
	oak	0.63	reclaimed	0.81		swivel	0.63	swivel	0.85
	solid wood	0.56	dining table	0.79		unfinished	0.56	walnut	0.84
	dining table	0.54	unfinished	0.78		seat	0.50	ladder	0.80
	leather	1.68	bonded	0.99		leather	1.20	coaster	0.98
	swivel	1.38	swivel	0.99		coaster	1.17	tufted	0.97
	faux leather	1.35	leather	0.99		swivel	1.07	bar stools	0.96
	bonded	1.32	faux leather	0.98		tufted	1.05	swivel	0.96
	stools	1.07	tufted	0.97		chic	1.04	sectional	0.95
	tufted	1.05	leatherette	0.97		bar stools	1.00	leatherette	0.95
	vinyl	1.00	stools	0.96		vinyl	0.97	leather	0.94
	chair ottoman	0.97	chair ottoman	0.95		faux leather	0.95	vinyl	0.94
	solid	0.62	mission	0.82			0.61	hillsdale	0.94
	honey	0.48	powell	0.80		hillsdale	0.60	drawer dresser	0.80
	mission	0.42	antiqued	0.80		classic	0.51	honey	0.77
	antiqued	0.42	honey	0.78		honey	0.51	7pc	0.75
	counter	0.41	rta	0.77		seat	0.47	dining room set	0.75
	classic	0.40	handmade	0.74		dining room set	0.45	old world	0.75
	bar stool	0.36	counter	0.74		old world	0.42	chinese	0.74
	finish	0.36	solid	0.70		drawer dresser dresser	0.42	powell	0.72
	contemporary	0.90	recliner	0.98		adjustable	0.79	recliner	0.94
	recliner	0.83	world	0.86		stool	0.74	bonded leather	0.91
	mahogany	0.70	chair ottoman	0.80		recliner	0.70	adjustable	0.86
	modern	0.50	contemporary	0.77		bonded leather	0.64	stool	0.86
	chair ottoman	0.49	sectional	0.77		european	0.63	dining table chairs	0.86
	world	0.48	powell	0.76		dining table chairs	0.63	microfiber	0.85
	italian	0.39	bonded leather	0.76		microfiber	0.59	european	0.84
	bar	0.38	mahogany	0.75		modern	0.59	rta	0.81
	bed	0.66	bed	0.84		size	0.88	size	0.90
	queen	0.60	queen	0.83		furniture	0.81	hillsdale	0.90
	size	0.53	log	0.81		queen	0.78	sleigh bed	0.90
	bedroom furniture	0.52	headboard	0.80		bedroom furniture	0.78	queen	0.89
	headboard	0.48	size	0.80		bed	0.62	bedroom furniture	0.89
	traditional	0.42	mattress	0.79		sleigh bed	0.53	sleigh	0.87
	upholstered	0.38	bedroom furniture	0.78		bedroom	0.50	bed	0.81
	old world	0.37	sleigh bed	0.78		queen size	0.48	headboard	0.80

(a) Here we show some examples where we succeed at multi-attribute prediction. We show results for SIFT + LLC and Grabcut + SIFT + LLC, before and after Platt scaling calibration. These query images have complex background for which Grabcut does not remove background effectively. Thus, best results are observed for SIFT+LLC after calibration. As can be seen from Figure 4, most images in the test set have clean background. This explains why Grabcut was effective in Figure 7.

Image	SIFT+LLC (before calibration)		SIFT+LLC (after calibration)		Image+Grabcut	Grabcut+SIFT+LLC (before calibration)		Grabcut+SIFT+LLC (after calibration)	
	new	1.28	platform bed	0.95		mid century	0.63	club	0.84
	sofa	0.91	sofa	0.94		natural	0.62	twin bed	0.83
	platform bed	0.81	futon	0.93		glass	0.60	ethan	0.83
	headboard	0.68	sectional	0.91		foot stool	0.60	ottoman	0.83
	foot stool	0.68	headboard	0.88		cream	0.58	vinyl	0.82
	beige	0.64	ottoman	0.88		ottoman	0.58	sleigh bed	0.82
	ottoman	0.64	cover	0.86		chic	0.57	cover	0.80
	bedroom	0.63	foot stool	0.85		vinyl	0.57	futon cover	0.80
	cream	0.88	office desk	0.93		tall	0.95	trundle	0.90
	wide	0.87	mattress	0.91		wood	0.88	tall	0.90
	wood	0.87	maple	0.89		bed	0.88	bed	0.89
	maple	0.85	wide	0.88		solid	0.87	twin bunk	0.87
	italian	0.69	bonded leather	0.87		maple	0.86	maple	0.86
	tall	0.68	italian	0.87		dresser	0.73	dresser	0.86
	desk	0.67	cream	0.86		size	0.71	bed frame	0.86
	bronze	0.60	powell	0.84		foam	0.64	size	0.85
	chair	1.53	ottoman	0.98		ottoman	1.50	ottoman	0.99
	ottoman	1.12	chair	0.98		collection	0.87	sectional	0.91
	bonded	0.84	bonded leather	0.96		chair	0.85	cocktail	0.90
	bonded leather	0.80	sectional	0.96		dining table	0.82	chair	0.89
	chair ottoman	0.78	sofa bed	0.95		cocktail	0.71	chair ottoman	0.89
	dark	0.77	bonded	0.95		walnut	0.70	dining table	0.89
	cream	0.76	chair ottoman	0.92		leather	0.68	bonded leather	0.87
	leather	0.69	futon	0.89		eco	0.67	recliner	0.87
	black	0.96	recliner	0.96		espresso	1.10	arm	0.93
	faux	0.94	futon	0.93		silver	0.89	coaster	0.90
	espresso	0.91	mesh	0.89		arm	0.85	holly	0.87
	new	0.75	step stool	0.89		office	0.81	recliner	0.87
	italian	0.74	italian	0.88		red	0.78	espresso	0.86
	red	0.68	tower	0.86		black	0.76	silver	0.84
	recliner	0.66	faux	0.85		coaster	0.68	italian	0.83
	pair	0.58	pair	0.85		walnut	0.64	office	0.83
	bed	1.11	bed	0.95		twin bed	0.62	twin bed	0.90
	twin	0.94	twin	0.95		bed	0.61	log	0.88
	new	0.76	mattress	0.89		stool	0.54	sleigh bed	0.87
	finish	0.65	industrial	0.81		pine	0.53	step stool	0.82
	frame	0.55	king size	0.81		bench	0.53	cedar	0.80
	bar	0.50	sleigh bed	0.80		sleigh bed	0.48	pine	0.80
	industrial	0.49	cedar	0.80		twin	0.47	bed	0.80
	folding	0.48	folding	0.80		king size	0.47	king size	0.80

(b) Here we show some examples where our method mostly gets confused at multi-attribute prediction. Often, confusion arises due to ambiguity in size of object.

Figure 9: Qualitative results of multi-attribute prediction. Examples of both successful as well as failure cases are shown.