SUN Database: Exploring a Large Collection of Scene Categories

Citation: Xiao, Jianxiong et al. "SUN Database: Exploring a Large Collection of Scene Categories." International Journal of Computer Vision 119.1 (2016): 3–22.

As Published: http://dx.doi.org/10.1007/s11263-014-0748-y

Publisher: Springer US

Persistent URL: http://hdl.handle.net/1721.1/106970

Version: Author's final manuscript: final author's manuscript post peer review, without publisher's formatting or copy editing

International Journal of Computer Vision manuscript No.
(will be inserted by the editor)

SUN Database: Exploring a Large Collection of Scene Categories

Jianxiong Xiao · **Krista A. Ehinger** · **James Hays** · **Antonio Torralba** · **Aude Oliva**

Received: date / Accepted: date

Abstract Progress in scene understanding requires reasoning about the rich and diverse visual environments that make up our daily experience. To this end, we propose the Scene Understanding (SUN) database, a nearly exhaustive collection of scenes categorized at the same level of specificity as human discourse. The database contains 908 distinct scene categories and 131,072 images. Given this data with both scene and object labels available, we perform in-depth analysis of co-occurrence statistics and the contextual relationship. To better understand this large scale taxonomy of scene categories, we perform two human experiments: we quantify human scene recognition accuracy, and we measure how typical each image is of its assigned scene category. Next, we perform computational experiments: scene recognition with global image features, indoor versus outdoor classification, and "scene detection," in which we relax the assumption that one image depicts only one scene category. Finally, we relate human experiments to machine performance and explore the relationship between human and machine recognition errors and the relationship between image "typicality" and machine recognition accuracy.

Jianxiong Xiao
Princeton University
E-mail: xj@princeton.edu

Krista A. Ehinger
Harvard Medical School
E-mail: kehinger@mit.edu

James Hays
Brown University
E-mail: hays@cs.brown.edu

Antonio Torralba
Massachusetts Institute of Technology
E-mail: torralba@csail.mit.edu

Aude Oliva
Massachusetts Institute of Technology
E-mail: oliva@mit.edu

Fig. 1 Examples of scene categories in our SUN database.

Keywords Scene Recognition · Scene Detection · Scene Descriptor · Scene Typicality · Scene and Object · Visual Context

1 Introduction

Scene understanding is the gateway to many of our most valued behaviors, such as navigation, recognition, and reasoning with the world around us. By "scene" we mean a place within which a person can act, or a place to which a person could navigate. In this paper we hope to address many questions about the "space of scenes" such as: How many kinds of scenes are there? How can scene categories be organized? Are some exemplars better than others? Do scenes co-occur in images? How do the spatial envelope properties correlate with the social functions of scenes? How do the current state-of-the-art scene models perform on hundreds of scene categories encountered by humans, and how do these computational models compare to human judgments about scenes?

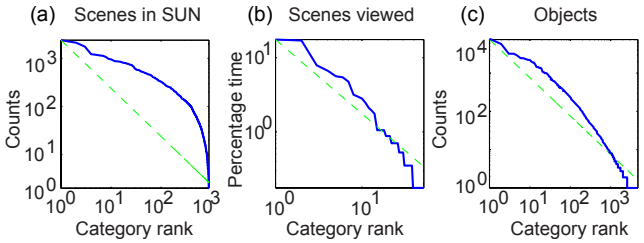


Fig. 2 (a) Sorted distribution of scene classes in the SUN database. (b) Sorted distribution of scene classes encountered while recording daily visual experience. (c) Sorted distribution of object counts in the SUN database. The dashed line corresponds to the function $A/rank$, where the constant A is the max of each curve.

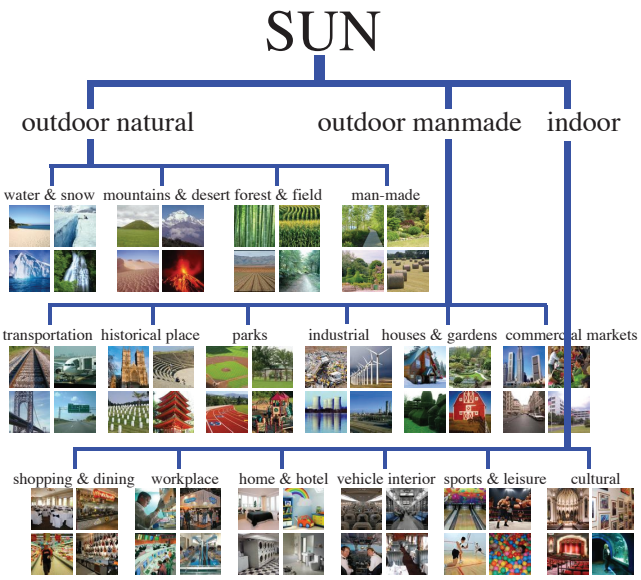


Fig. 3 The first two levels of a hierarchy of scene categories.

Given that most of the places we experience are built by and for people, the number of scene classes or partitions one can make of the world is in constant evolution: there may be finer-grained categories emerging from economical or functional constraints (e.g., compact apartment) or categories with only one exemplar (e.g., a specific space station) [8]. Despite this variability, there is a core number of places that people tend to encounter in the world, that form the basis of categorical knowledge for the field of scene understanding. The list proposed here represents a lower bound on the number of places that can be named.

Whereas most computational work on scene and place recognition has used a limited number of semantic categories, representing typical indoor and outdoor settings [25, 16, 32, 50, 29, 4, 46], access to large quantities of images on the Internet now makes it possible to build comprehensive datasets of images organized in categories [19, 45, 10] in order to capture the richness and diversity of environments that make up our daily experience.

Although the visual world is continuous, most environmental scenes, like objects, are visual entities that can be organized in functional and semantic groups. Like objects, particular environments will trigger specific actions, such as eating in a restaurant, drinking in a pub, reading in a library, and sleeping in a bedroom. However, when faced with environments from a given basic-level semantic category (e.g. kitchen), people may behave differently and have different expectations depending on the specifics of the place (e.g. a house kitchen, a restaurant kitchen, an industrial kitchen). Therefore, it is critical for artificial vision systems to discriminate the type of environments at the same level of specificity as humans. Here, we provide a fine-grained taxonomy and dataset representing the diversity of visual scene categories that can be encountered in the world (Fig. 1), and we provide computational benchmarks for large-scale scene categorization tasks.

This paper has the following four objectives. First, we propose a method to, as thoroughly as possible, determine the number of different scene categories. We identify all the scenes and places that are important enough to have unique identities in discourse, and build a large scale dataset of scene image categories. Second, we perform experiments to measure how accurately humans can classify exemplars of scenes into hundreds of categories, how "typical" particular scenes are of their assigned scene category, and how scene categories relate in terms of high-level semantic properties. Third, we evaluate the scene recognition and indoor vs outdoor classification on this large scale scene database using a combination of many image features. Finally, we introduce the scene detection task with the goal of determining which scene categories are present in local image regions. Where appropriate, we explore the relationship between human experiments and machine performance.

2 Building the SUN database

In this section we describe our procedure to build a large-scale database of scenes. We provide a rough estimate of the number of common scene types that exist in the visual world and build an extensive image database to cover as many of these as possible. We refer to this dataset as the *SUN* (Scene UNderstanding) database[1] [52].

2.1 Constructing scene taxonomy

In order to define a list of scene categories, we follow a procedure similar to Biederman's [6] process for determining

[1] All the images and scene definitions are available at `sundatabase.mit.edu` or `sun.cs.princeton.edu`.

Fig. 4 Visualization of the scene categories based on the number of images in each category. Larger font size indicates more images in the corresponding category. Color is used randomly for visualization purpose only.

the number of objects by counting object names in the dictionary. Here, we used WordNet [17], an electronic dictionary of the English language containing more than 100,000 words. We first selected the 70,000 words that correspond to non-abstract terms and that are available in the Tiny Images dataset [45] . We then manually selected all of the terms that described scenes, places and environments (any concrete noun which could reasonably complete the phrase "I am in a *place*", or "Let's go to the *place*"). Most of the terms referred to entry-level places [47, 33, 34, 22]. In the categorization literature, "entry-level" refers to the level of categorization most commonly used in everyday situations, (e.g., "kitchen" or "classroom"). In reference to visual scenes, these entry-level terms would refer to a set of environments that share visual similarities and objects, which may lead to similar interactions and activities. We did not include specific place names (like Grand Canyon or New York) or terms that did not seem to evoke a specific visual identity (territory, workplace, outdoors). Non-navigable scenes (such as desktop) were not included, nor were vehicles (except for views of the inside of vehicles) or scenes with mature content. We included specific types of buildings (skyscraper, house, hangar), because, although these can be seen as objects, they are known to activate scene-processing-related areas in the human brain. [13]. We also included many vocabulary terms that convey significance to experts in particular domains (e.g. a baseball field contains specialized subregions such the pitcher's mound, dugout, and bullpen; a wooded area could be identified as a broadleaf forest, rainforest, or orchard, depending upon its layout and the particular types of plants it contains). To the WordNet collection we added a few categories that seemed like plausible scenes but were missing from WordNet, such as jewelry store and mission.

This gave about 2500 initial scene words, and after manually combining synonyms (provided by WordNet) and separating scenes with different visual identities (such as indoor and outdoor views of churches), this was refined to a final dataset of 908 categories.

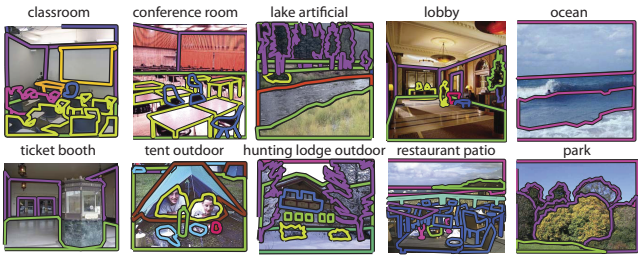


Fig. 5 Examples from 19,503 fully annotated images in SUN.

It is possible to use a similar procedure to get an estimate of the number of object words in the WordNet database. As with the scenes, we started with the 70,000 non-abstract terms from WordNet. We then selected a random 2% of the words and determined what proportion of these were objects. Including synonyms, there are about 2,500 scene words and about 27,000 object words; that is to say, there are about 10 times as many object words as there are scene words. This difference reflects the fact that there are more subordinate-level terms in the object domain (e.g., names for each individual species of plant and animal) and more synonyms for the same object (an individual species has both a scientific name and one or more common names). [2] What this analysis makes clear is that there are far more words for objects than scenes, and likely more distinct categories of objects than there are distinct categories of scene. Although we can think of scenes as distinct combinations of objects, there are far fewer scene categories than there are possible configurations of objects. This is because not all distinct object configurations give rise to different scene categories, and most scene categories are flexible in terms of their constituent objects (e.g., living rooms can contain many different types of objects in various configurations).

2.2 Collecting images

Once we have a list of scenes, the next task is to collect images belonging to each scene category. Since one of our goals is to create a very large collection of images with variability in visual appearance, we collected images available on the Internet using online search engines for each scene category term. Similar procedures have been used to create object databases such as Caltech 101 [15], Caltech 256 [19], Tiny Images [45] and ImageNet [10].

For each scene category, images were retrieved using a WordNet term from various search engines on the web. When a category had synonyms, images for each term were retrieved and then the images were combined. Only color images of 200×200 pixels or larger were kept. For similar

[2] This difference also explains why our count is much higher than Biederman's [6] estimate of about 1000 basic-level objects – we included all object words in our count, not just basic-level terms.

Fig. 6 Object annotation in the SUN database. At left is a visualization of the object categories based on the number of annotations in each category. Larger font size indicates more examples of that object category. Color is used randomly for visualization purpose only. At right are examples of the 12,839 annotated chairs in the SUN database.

scene categories (e.g. "abbey", "church", and "cathedral") explicit rules were formed to avoid overlapping definitions. Images that were low quality (very blurry or noisy, black-and-white), clearly manipulated (distorted colors, added text or borders, or computer-generated elements) or otherwise unusual (aerial views, incorrectly rotated) were removed. Duplicate images, within and between categories, were removed. Then, a group of participants (N=9, including some of the authors) manually removed all the images that did not correspond to the definition of the scene category.

For many of the 908 SUN categories an image search returns relatively few unique photographs. The success of each search depends upon how common the scene category is in the world, how often people photograph that type of scene (and make their photos available on the Internet), and how often the photos are labeled with the category name or a synonym from our list. For example, the SUN database contains more images of living rooms than airplane cabins; this is probably because airplane cabins are encountered less often, and even when people do take photos inside airplanes, they don't necessarily label the image as "airplane cabin." Because it is much more difficult to find images for some scene categories than for others, the distribution of images across scene categories in the database is not uniform (Fig. 4). Examples of scene categories with more images are living room, bedroom, and bookstore. Examples of under-sampled categories include airlock, editing room, grotto, launchpad, naval base, oasis, ossuary, salt plain, signal box, sinkhole, sunken garden, and winners circle.

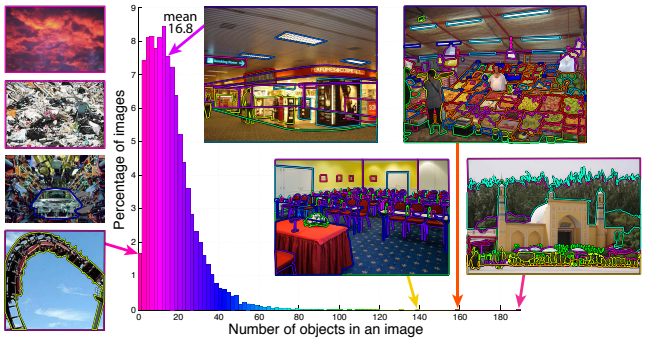
To provide data for research of objects in scenes, using LabelMe [36], we have also labeled objects in a large portion of the image collection with polygonal outlines and object category names. We describe the details for object labeling protocol in a separate technical report [5]. To date, there are 326,582 manually segmented objects for the 5,650 object categories labeled. Example annotations are shown in Fig. 5.

2.3 Taxonomy and database limitations

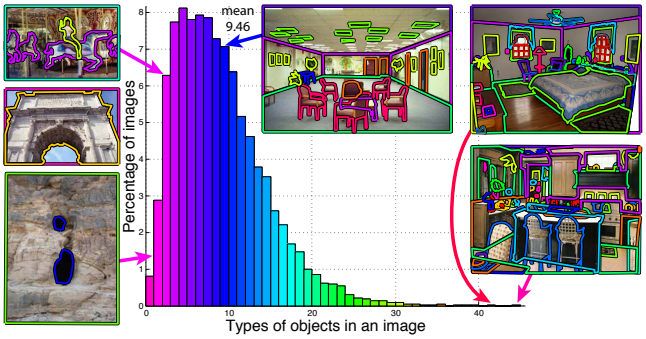
Estimating the number of categories that compose a set of items from a finite sample is a challenging task (see [8] for a review). In the case of scene categories, this number might be infinite as there might always be a new, very rare, category with a specific function not considered before. Our dataset is not be an exhaustive list of all scene categories, but we expect that the coverage is large enough as to contain most of the categories encountered in everyday life.

It is important to acknowledge that the procedure used here is not the only way to create a list of scene categories and collect images. There are different ways to define and categorize "scenes", which would generate different organizations of the images, and different categories, than the one used here. For instance, an alternate strategy would be to record the visual experience of an observer and to count the number of different scene categories viewed. We had 7 participants (including 2 of the authors) write down, every 30 minutes, the name of the scene category in which they were located, for a total of 284 hours across participants. During that time, the participants reported a total of 54 distinct places. All the scenes provided were already part of the previous list produced from WordNet which we take as an indication of the completeness of the list provided by WordNet. This procedure is unlikely to produce a complete list of all scene categories, as many scene categories are only viewed on rare occasions (e.g., cloister, corn field, etc.) and would be dependent on the individual daily activities. However, this method would have the advantage of producing a list that would also provide information about the real frequency of environments encountered under normal viewing conditions.

Fig. 2(b) shows the sorted distribution of scenes obtained in this way. In this plot, the vertical axis corresponds to the percentage of time spent in each scene type (which is an indication of the number of images that would be collected if we recorded video). Note that the distribution looks quite different from the one in Fig. 2(a). For comparison, we also



(a) Number of objects per image



(b) Types of objects per image

Fig. 7 Per-image object statistics in the SUN database.

show the distribution of objects in the SUN dataset ([44]). The distributions in Fig. 2(b) and Fig. 2(c) look similar and can be approximated by a Zipf law with the form $A/rank$, where A is a constant and the number of instances of a category is inversely proportional to its rank in the sorted list.

There are also different ways to sample the visual world in order to create a collection of images for each category. For example, one might decide that different views of the same place qualify as different scenes, or one might choose to subdivide scenes based on spatial layout or surface features (e.g., forests with or without snow). Our goal here is to propose an initial list that is extensive enough as to cover most of plausible scene categories. Like estimating the number of visual object categories, counting the number of scene categories is an open problem and here we are providing a first estimate.

It should also be noted that this database is biased to a U.S. / English-speaking perspective of scenes, because the database was constructed around the set of English-language scene names. Likewise, all the human experiments we report on this database were conducted in English and some (particularly the scene typicality ratings) were restricted to U.S.-based participants. There is no doubt an important role of culture in the way people perceive and categorize scenes, but trying to build a scene dataset that encompasses all languages and cultures is beyond the scope of this paper. Instead, it seems a reasonable starting point to focus on scene categorization in a particular group.

3 Analyzing the SUN database

The SUN database is the first dataset with a large coverage of natural scenes with objects. Therefore, it provides us the unprecedented opportunity to obtain the natural statistics for objects and scenes, and study their relationship. In this section, we conduct in-depth analysis in these aspects.

3.1 Statistics of scenes

The final dataset contains 908 categories and 131,072 images[3]. Fig. 2(a) shows the distribution of the number of images collected for each scene category in the SUN database, where the categories are sorted in decreasing order of available images.

3.2 Statistics of objects

A visualization of label counts is shown in Fig. 6. The most common objects are those which appear frequently in a large number of scene categories, so the most common objects in the database are structural regions (wall, window, floor, ceiling, and sky), followed by ubiquitous objects like chairs, people, trees, and cars. Objects which occur only in particular scene categories, like soccer goal or X-ray machine, are of course much less common. As with scenes, the distribution of objects in the database is a function of how common the object is in the world, the likelihood that a photographer will choose to include the object in a scene, and the likelihood that it will be labeled with a particular term.

Our object annotation dataset differs from other popular object datasets, such as ImageNet and PASCAL VOC. These databases were created by finding images for specific object categories, but the SUN database was created by collecting scene categories – we did not have object recognition in mind. Therefore, the statistics of objects in the SUN database might be expected to better match the statistics of objects in real-world settings. Fig. 7 shows the distribution of images over different numbers of object instances and object types. Fig. 7(a) shows that on average, there are 16.8 object instances in each image, ranging from 1 object in a sky picture to 190 object instances in a mosque picture which shows a crowd of people. In comparison, the average number of instances per image is 1.69 in the PASCAL dataset,

[3] The number of images is continously growing as we run our scripts to query more images from time to time.

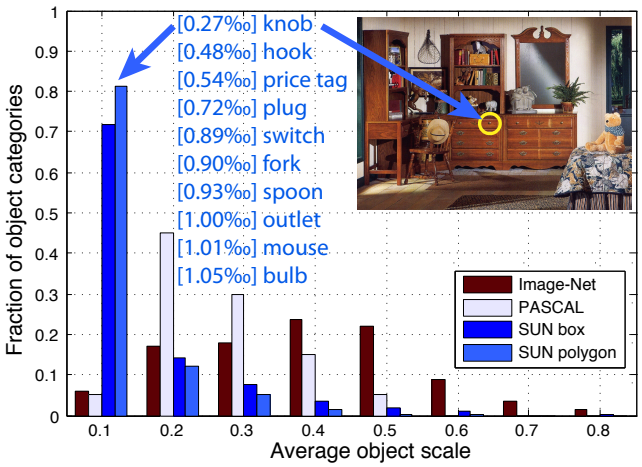
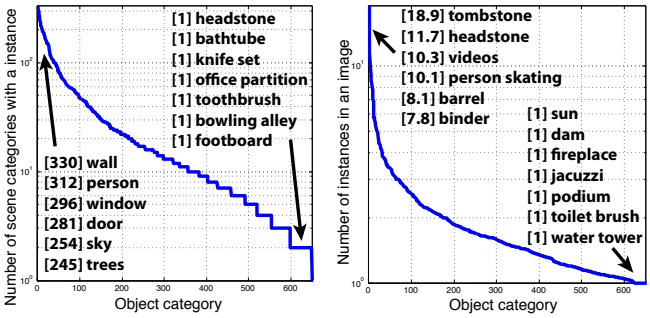


Fig. 8 Average object scale compared to PASCAL VOC and ImageNet ILSVRC. SUN polygon is computed using the normalized area of the bounding polygon, and SUN box is computed using a the bounding box around the object. The blue text lists examples of object categories with very small areas in a typical image, with their average scale in per mil image pixels.



(a) For each object category, number of scene categories with at least one instance of the object.

(b) For each object category, average number of instances per image.

Fig. 9 Per object category statistics on SUN database.

1.59 in the ImageNet ILSVRC, and 1.91 in the normalized ILSVRC [35]. Fig. 7(b) shows that on average, there are 9.46 different object categories in a single image, with the most diverse image being a kitchen photo that has 46 different object categories in the same image. In general, indoor spaces such as bedrooms, kitchens, and dining rooms have a greater variety of object categories, which makes the object detection task more challenging. On the other hand, some outdoor natural spaces, such as savannas and ice floes, have very few possible object categories, which makes the standard object detection task easier. In comparison, PASCAL VOC 2012 and ImageNet ILSVRC 2013 have 1.5 object categories and 2.7 object instances per image, respectively.

Another important difference in the SUN database is the distribution of object sizes in the image (shown in Fig. 8). Following the criteria used in [35], the object scale in an image is calculated as the ratio between the area of a ground truth bounding box around the object and the area of the image. Using the bounding box measure, the average scale per class on the PASCAL dataset is 0.241 [35], and on the ImageNet ILSVRC is 0.358[35], while it is 0.0863 in the SUN database. In our database, many object categories are not well-approximated by bounding boxes because the object is thin or highly deformable (e.g. rope), or because the object is a region (wall, sky), so we also calculate object scale using the annotated outline of the object. In this case, object scale is the ratio between the area of the bounding polygon and the area of the image. If we calculate object scale using the ground truth bounding polygon, the average scale per class is 0.0550.

By annotating all of the objects in each image and with a large collection of images across a wide variety of natural scenes, we have a unprecedented opportunity to study object scale in real-world viewing conditions. Object detection literature (e.g. PASCAL) has typically been limited to objects over some minimum size (such as tables, cars, and chairs). Although significant progress has been made in detecting these categories, it does not necessarily represent progress on the majority of object categories. Our statistics on a large collection of natural scene images in Fig. 8 show that more than 80% of object categories have average pixel areas smaller than 10% of the image. For example, the average size of a "fork" in the SUN database is 0.09% of the image. Of course, one could take a closer view of a fork and it would fill more of the image, but in general, people typically encounter objects like forks at a very small scale. As Fig. 8 shows, objects at this scale can be recognized in the context of a scene. From the histogram, we can see that 72% of the object categories in the SUN database have smaller average scale than the typical PASCAL object scale, and 94% of the object categories have smaller scale than than typical ImageNet object scale. These statistics highlight the importance of recognizing these small objects, and expose the bias towards larger objects in most object detection literature. While the progress on detecting these larger objects has been impressive, we must remember great variety of small object categories when designing object recognition systems.

3.3 Scene-object co-occurance

Other popular recognition datasets typically only have either object or scene annotated, such as Pascal VOC, ImageNet, LabelMe and 15-scene dataset. Our SUN database uniquely provides a complete annotation for both objects and scenes on the same set of images, covering a large number of both categories. This provides us the unprecedented opportunity to obtain the natural statistics between objects and scenes and to study the interesting relationship among object and scenes.

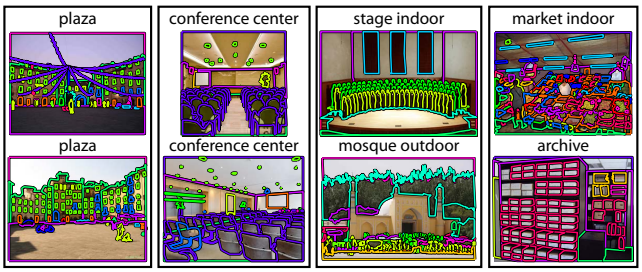


Fig. 10 Four example pairs of most similar images based on object histogram.

Fig. 11 Graphical User Interface of Amazon's Mechanical Turk task for 397-category Alternative Forced Choice.

Fig. 9(a) shows, for all of the labeled object categories, the number of scenes in which each object category appears. Some object categories appear in only one scene category, while common objects such as "wall" appear in 330 difference categories. Leveraging object-scene co-occurrence can be a powerful tool for recognition: some objects are very strong predictors of scene category. For example, if an object detector can find a bathtub in an image, based on the co-occurrence statistics of the SUN database, we can be 100% certain that the image must be a bathroom scene. Naturally, extremely rare objects are very highly predictive (e.g., "xylophone" appears only once in the database, so it is 100% predictive of its scene category), but since these objects are unlikely to occur in other settings, they are less useful for scene classification. Fig. 9(b) illustrates how often a object category occurs within the same image.

Just as objects can provide information about scene category, scene category is a strong predictor of object identity. For example, if a scene is known to be an example of the "bank indoor" category, we can be 100% certain that it will include a "floor" object. Similarly, if the scene category is "barn" we can be 100% certain that it will include a "sky" object. (Note that this is a statement about the database and not all possible images of these scenes: it is possible, though uncommon, to take a picture of a barn which does not include any sky.)

Using the object annotations, we can define the similarity of two images based on the histogram intersection of object instances in the two images, and see whether images from a single scene category tend to be more similar using this metric. Fig. 10 shows four pairs from the ten most similar pairs of images in the SUN database. The first two examples are pairs that share a scene category, which shows that in some cases, the object histogram can be used to detect similar scene categories. However, the third and the fourth example show that the object histogram does not necessarily find nearest neighbors from the same scene category. Both the "stage indoor" image and the "mosque outdoor" image have a lot of people, and both the "market indoor" image and the "archive" image have many boxes, which causes the histogram measure to view these pairs as highly similar. Just knowing the types of objects in a scene is unlikely to provide enough information to classify scenes at a fine-grained level: scene layout and visual features of the objects are necessary to distinguish, for example, a fortress from a hospital, or a hospital from a storefront.

4 Behavioral studies using the SUN database

In this section, we study (1) human scene classification performance on the SUN database, (2) human estimates of the "typicality" of every image with respect to its scene category.

4.1 Scene categorization

We ask human participants to classify images from the database into one of 397 scene categories in an alternative forced choice setting. For this experiment, we have two goals: 1) to show that our database is constructed consistently and with minimal overlap between categories 2) to give an intuition about the difficulty of 397-way scene classification and to provide a point of comparison for computational experiments (Section 5.2).

Measuring human classification accuracy with 397 categories is challenging. We don't want to penalize humans for being unfamiliar with our specific scene taxonomy, nor do we want to train people on the particular category definitions and boundaries used in our database (however, such training was given to those who built the database). To help participants know which labels are available, we provide the interface shown in Fig. 11. Participants navigate through a three-level hierarchy to arrive at a specific scene type (e.g. "bedroom") by making relatively easy choices (e.g., "indoor" versus "outdoor natural" versus "outdoor man-made" at the first level). The 3-level tree contains then 397 leaf nodes (SUN categories) connected to 15 parent nodes at the second level that are in turn connected to 3 nodes at the first level (super-ordinate categories). The mid-level categories were selected to be easily interpreted by workers, have minimal overlap, and provide a fairly even split of the images

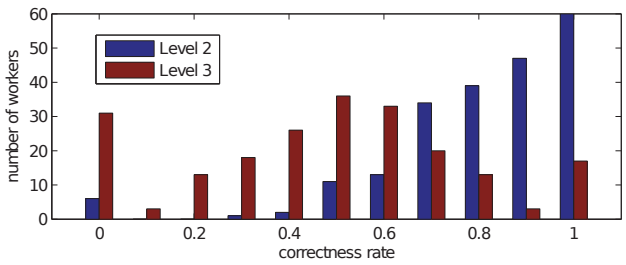


Fig. 12 Histogram of the scene recognition performances of *all* AMT workers. Performance is measured at the intermediate (level 2) and leaf (level 3) levels of our hierarchy.

Fig. 13 SUN categories with the highest human recognition rate.

in each super-ordinate category. When there was any confusion about the best super-ordinate category for a category (e.g., "hayfield" could be considered natural or man-made), the category was included in both super-ordinate categories. This hierarchy is used strictly as a human organizational tool, and plays no roll in the experimental evaluations (although we do report machine classification performance on the different levels of the hierarchy, simply for completeness). For each leaf-level SUN category the interface shows a prototypical image from that category.

We measure human scene classification accuracy using Amazon's Mechanical Turk (AMT). For each SUN category we measure human accuracy on 20 test scenes, for a total of $397 \times 20 = 7940$ trials. We restricted these HITs to participants in the U.S. to help avoid vocabulary confusion.

The accuracy of all AMT workers is shown in Fig. 12. On average, workers took 61 seconds per HIT and achieved 58.6% accuracy at the leaf level. This is quite high considering that chance is 0.25% and numerous categories are closely related (e.g., "church", "cathedral", "abbey", and "basilica"). However, a significant number of workers have 0% accuracy – they do not appear to have performed the experiment rigorously. If we instead focus on the "good workers" who performed at least 100 HITs and have accuracy greater than 95% on the relatively easy first level of the hierarchy the leaf-level accuracy rises to 68.5%. These 13 "good workers" accounted for just over 50% of all HITs. For reference, an author involved in the construction of the database achieved 97.5% first-level accuracy and 70.6% leaf-level accuracy. In the remainder of the paper, all evaluations and comparisons of human performance utilize only the data from the good AMT workers.

Fig. 13 and 14 show the SUN categories for which the good workers were most and least accurate, respectively. For the least accurate categories, Fig. 14 also shows the most frequently confused categories. The confused scenes are semantically similar – e.g. abbey and church, bayou and river, and sandbar and beach. Within the hierarchy, indoor sports and leisure scenes are the most accurately classified (78.8%) while outdoor cultural and historical scenes were least accurately classified (49.6%). Even though humans perform poorly on some categories, the confusions are typically restricted to just a few classes (Fig. 21).

Human and computer performance are compared extensively in Section 5.2. It is important to keep in mind that the human and computer tasks are not completely equivalent. The "training data" for AMT workers was a text label, a single prototypical image, and their past visual experience with each category (which could be extensive for everyday categories like "bedroom" but limited for others). The computational model had 50 training examples per category. It is also likely that human and computer failures are qualitatively different – human misclassifications are between semantically similar categories (e.g. "food court" to "fast food restaurant"), while computational confusions are more likely to include semantically unrelated scenes due to spurious visual matches (e.g., "skatepark" to "van interior"). In Fig. 22 we analyze the degree to which human and computational confusions are similar. The implication is that the human confusions are the most reasonable possible confusions, having the shortest possible semantic distance. But human performance isn't necessarily an upper bound – in fact, for many categories the humans are less accurate than the best computational methods (Fig. 20).

4.2 Typicality of scenes

In the computer vision literature, the organization of visual phenomena such as scenes into *categories* is ubiquitous. Each particular instance is assumed to be an equally good representative of the category. This is a useful high level model for many computational experiments, but most theories of categorization and concepts agree that category membership is graded - some items are more typical examples of their category than others [47]. The most typical examples of a category show many advantages in cognitive tasks: for example, typical examples are more readily named when people are asked to list examples of a category, and response times are faster for typical examples when people are asked to verify category membership [33].

To study the typicality of scenes, we ran a task on Amazon's Mechanical Turk to ask human annotators to choose most and least typical examples from a list of images [12].

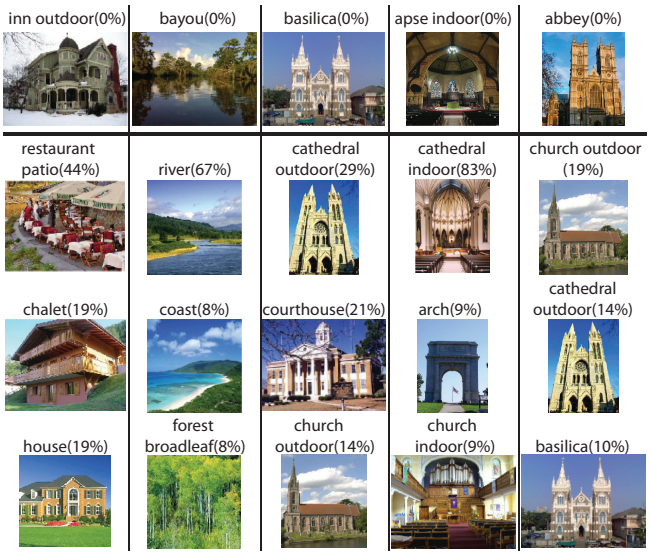


Fig. 14 Top row: SUN categories with the lowest human recognition rate. Below each of these categories, in the remaining three rows, are the most confusing classes for that category.

Participants were told that the goal of the experiment was to select illustrations for a dictionary. Each trial consisted of three parts. First, participants were given the name of a scene category from the database, a short definition of the scene category, and four images. Workers were asked to select which of the four images matched the category name and definition (one of the four images was drawn from the target category and the other three were randomly selected from other categories). The purpose of this task was to ensure that participants read the category name and definition before proceeding to the rating task. Next, participants were shown 20 images in a 4×5 array. These images were drawn randomly from the target category, and did not include the image which had served as the target in the previous task. Images were shown at a size of 100×100 pixels, but holding the mouse over any image caused a larger 300×300 pixel version of that image to appear. An example of this display is shown in Fig. 15. Workers were asked to select, by clicking with the mouse, three images that best illustrated the scene category. In the third part of the task, workers were shown the same 20 images (but with their array positions shuffled) and were asked to select the three worst examples of the target scene category.

For this experiment we used the 706 scene categories from our SUN database that contained at least 22 exemplars[4]. On each trial, the set of 20 images was drawn randomly from the set of images in the target category. These random draws were such that each image appeared at least 12 times, and no more than 15 times over the course of

Fig. 15 The display seen by participants in the typicality rating task.

the experiment. This resulted in 77,331 experimental trials. Each trial was completed by a single participant. 935 people participated in the experiment[5]. Participants could complete as many trials as they wished; the average number of trials completed per participant was 82.7 trials (median 7 trials).

Participants' performance was evaluated using two measures: 1) performance on the 4AFC task, and 2) whether they selected different images as the best and worst examples on a single trial. In general, participants performed well on the 4AFC task, with an average correct response rate of 97% (s.d. 0.13%). Participants also reliably selected different images as the best and worst examples of their category: participants marked an image as both best and worst on only 2% of trials (s.d. 0.10%); the likelihood of re-selecting an image by chance is 40%. We identified 19 participants (2% of total participants) who re-selected the same images as both best and worst on at least 25% of trials, which suggests that they were selecting images at random with no regard for the task. Together these participants had submitted 872 trials (1.13% of trials), which were dropped from further analysis.

A typicality score was obtained for each image in the dataset. The typicality score was calculated as the number of times the image had been selected as the best example of its category, minus a fraction (0.9) of the number of times it was selected as the worst example, divided by the number of times the image appeared throughout the experiment:

$$\text{typicality} = \frac{\#\text{ of ``best'' votes} - 0.9 \times \#\text{ of ``worst'' votes}}{\text{number of appearances}}.$$

Taking a fraction of the worst votes allows the number of best votes to be used as a tie-breaker for images that performed similarly. A typicality score near 1 means an image is extremely typical (it was selected as the best example of its category nearly every time it appeared in the experiment), and a typicality score near -1 means an image is extremely atypical (it was nearly always selected as a worst example). Although the fraction 0.9 was chosen arbitrarily, any value

[4] Category size ranged from 22 images in the smallest categories to 2360 in the largest. A total of 124,901 images were used in the experiment.

[5] All workers were located in the United States and had a good performance record with the service (at least 100 HITs completed with an acceptance rate of 95% or better). Workers were paid $0.03 per trial.

(a) Most typical beaches (b) Least typical beaches

(c) Most typical bedrooms (d) Least typical bedrooms

Fig. 16 Example images rated as the most and least typical by participants from Amazon's Mechanical Turk.

in the range 0.500 to 0.999 gives essentially the same results: changing this value changes the range of possible scores, but doesnt significantly change the rank order of scores within a category (90% of images move by less than 5 percentile points). Examples of the most and least typical images from various categories are shown in Fig. 16.

(a) Outdoor images mis-classified as indoor scenes.

(b) Indoor images mis-classified as outdoor scenes.

Fig. 18 Typical errors for the indoor-vs-outdoor classification.

5 Scene recognition on the SUN database

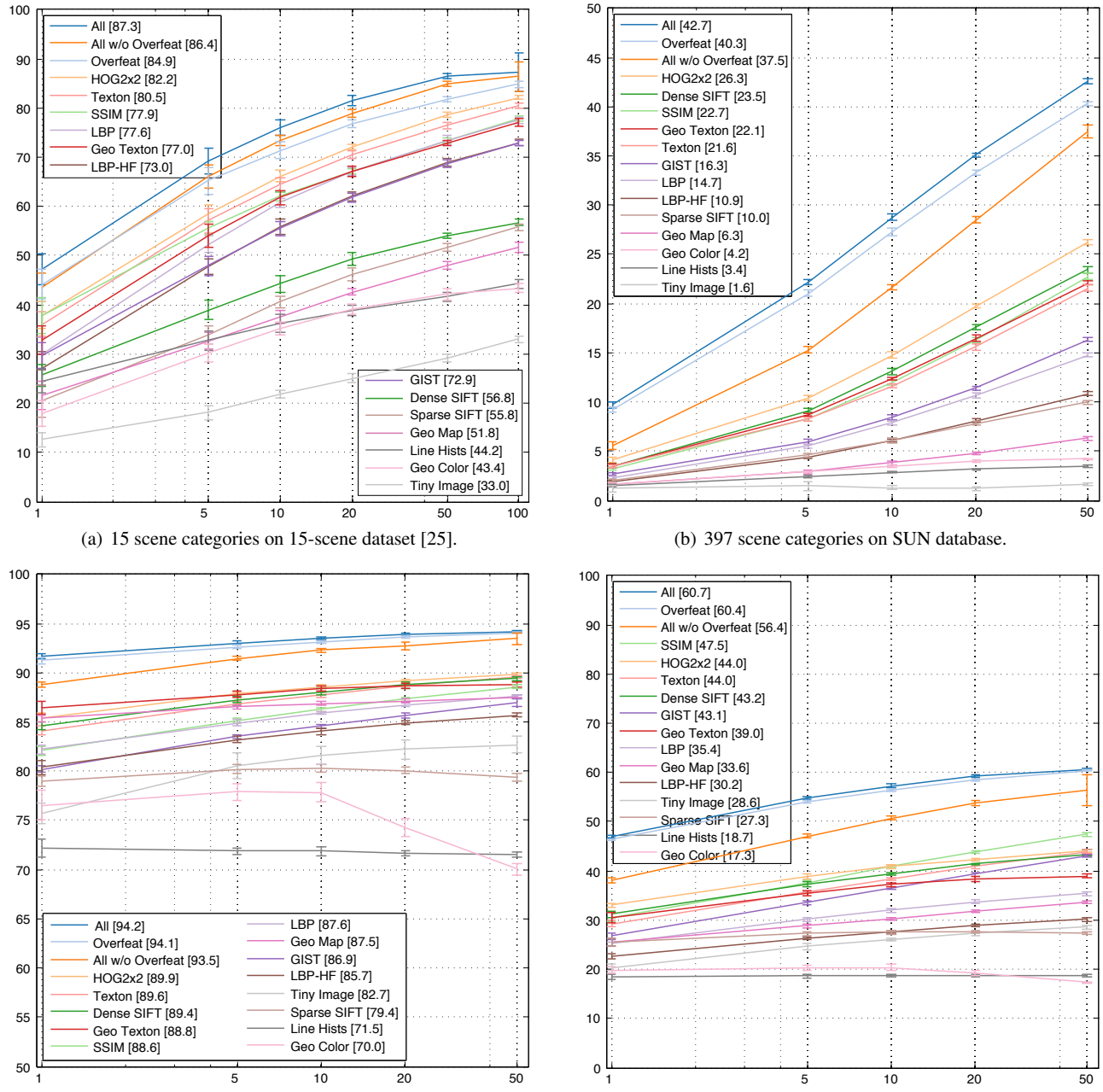
In this section we explore how discriminable the SUN categories and exemplars are with a variety of image features and kernels paired with One-vs-Rest Support Vector Machines.

5.1 Scene features

We selected or designed several state-of-the-art features that are potentially useful for scene classification: GIST, SIFT, and HOG (which are all local gradient-based approaches), SSIM (which relates images using their internal layout of local self-similarities), and Berkeley texton. As a baseline, we also include Tiny Image [45], and straight line histograms. To make our color and texton histograms more invariant to scene layout, we also build histograms for specific geometric classes as determined by [21]. The geometric classification of a scene is then itself used as a feature, hopefully being invariant to appearance but responsive to layout.

GIST: The GIST descriptor [29] computes a wavelet image decomposition. Each image location is represented by the output of filters tuned to different orientations and scales. We use a Gabor-like filters steerable pyramid with 8 orientations and 4 scales applied to the intensity (monochrome) image. To capture global image properties while keeping some spatial information, we take the mean value of the magnitude of the local features averaged over large spatial regions. The square output of each filter is averaged on a 4×4 grid. This results in an image descriptor of $8 \times 4 \times 16 = 512$ dimensions. GIST features [29] are computed using the code available online and we use an exponential χ^2 kernel. To establish a comparison between these hand designed features and features that learned from data, we ue the state-of-the-art Convolutional Neural Networks feature Overfeat [39].

HOG2×2: The histogram of oriented edges (HOG) descriptors are widely use for pedestrian and object detection. HOG decomposes an image into small squared cells (typically 8×8 pixels), computes a histogram of oriented gradients in each cell, normalizes the result using a block-wise pattern (with 2×2 square HOG blocks for normalization), and return a descriptor for each cell. HOG exists in two major variants: the original Dalal-Triggs variant [9] and the UoCTTI variant [18]. Dalal-Triggs HOG [9] works with undirected gradients only and does not do any compression, for a total of 36 dimension. UoCTTI HOG [18] computes instead both directed and undirected gradients as well as a four dimensional texture-energy feature, but projects the result down to 31 dimensions. In may applications, UoCTTI HOG tends to perform better than Dalal-Triggs HOG. Therefore, we use UoCTTI HOG in our experiments, computed using the code available online provided by [18]. In [52], we argue that stacking the features from multiple HOG cells into one feature is very important, because the higher feature dimensionality provides more descriptive power, and significantly improves the performance in our experiments. In our experiment, we tried different sizes of windows for stacking the HOG features, and only report the best performing one. 2×2 neighboring HOG descriptors are stacked together to form a 124 dimensional descriptor. The stacked descriptors spatially overlap. The descriptors are quantized into 300 visual words by k-means. With this visual word representation, three-level spatial histograms are computed on grids of 1×1, 2×2 and 4×4. Histogram intersection[25] is used to define the similarity of two histograms at the same pyramid



(a) 15 scene categories on 15-scene dataset [25].

(b) 397 scene categories on SUN database.

(c) Indoor-vs-outdoor binary classification on SUN database.

(d) 16-way classification at the second-level hierarchy on SUN.

Fig. 17 (a) Classification accuracy on the 15 scene dataset[29,25,16]. (b) Classification accuracy on the 397 well-sampled categories from SUN database. (c) Classification accuracy for indoor-vs-outdoor task using SUN database. (d) Classification accuracy for the 16 categories at the second level of the scene hierarchy using SUN database.

level for two images. The kernel matrices at the three levels are normalized by their respective means, and linearly combined together using equal weights.

Dense SIFT: As with HOG2×2, SIFT descriptors are densely extracted [25] using a flat rather than Gaussian window at two scales (4 and 8 pixel radii) on a regular grid at steps of 1 pixels. First, a set of orientation histograms are created on 4×4 pixel neighborhoods with 8 bins each. These histograms are computed from magnitude and orientation values of samples in a 16×16 neighboring region such that each histogram contains samples from a 4×4 subregion of the original neighborhood region. The magnitudes are further weighted by a Gaussian function with equal to one half the width of the descriptor window. The descriptor then becomes a vector of all the values of these histograms. Since there are 4×4 histograms each with 8 bins the vector has 128 elements. This vector is then normalized to unit length in order to enhance invariance to affine changes in illumi-

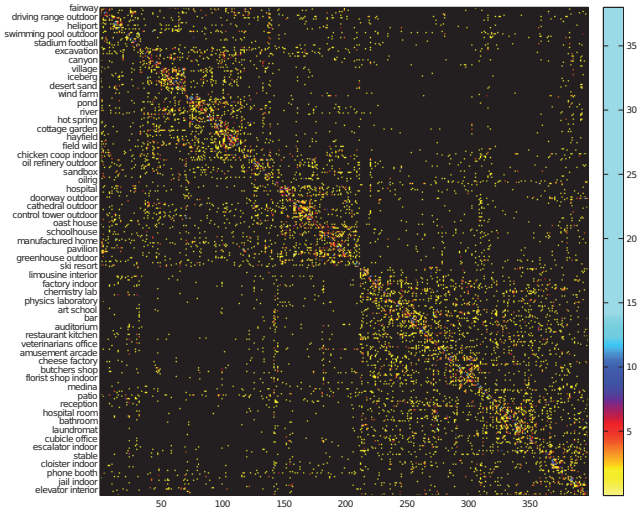


Fig. 19 Pattern of confusion across categories. The classes have been ordered to reveal the blocky structure. For clarity, the elements in the diagonal have been set to zero in order to increase the contrast of the off-diagonal elements. On the Y axis we show a sample of the scene categories. Confusions seem to be coherent with semantic similarities across classes. The scenes seem to be organized as natural (top), urban (center) and indoor (bottom).

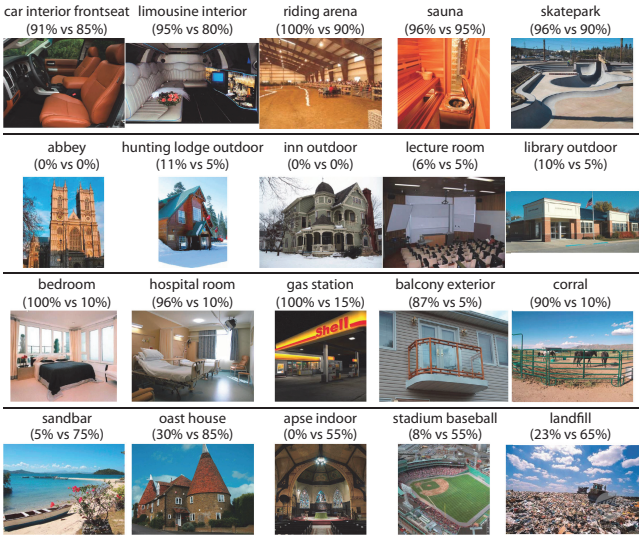


Fig. 20 Categories with similar and disparate performance in human and "all features" SVM scene classification. Human accuracy is the left percentage and computer performance is the right percentage. From top to bottom, the rows are 1) categories for which both humans and computational methods perform well, 2) categories for which both perform poorly, 3) categories for which humans perform better, and 4) categories for which computational methods perform better. The "all features" SVM tended to outperform humans on categories for which there are semantically similar yet visually distinct confusing categories, e.g., sandbar and beach, baseball stadium and baseball field, landfill and garbage dump.

nation. To reduce the effects of non-linear illumination a threshold of 0.2 is applied and the vector is again normalized. The three descriptors are stacked together for each HSV color channels[6], and quantized into 300 visual words by k-means. Next, kernels are computed from spatial pyramid histograms at three levels by the same method above for HOG2$\times$2. SIFT descriptors are computed using the VLFeat library [48] and we also use a histogram intersection kernel as in [25].

LBP: Local Binary Patterns (LBP) [28] is a multi-resolution approach to gray-scale and rotation invariant texture classification based on local binary patterns and nonparametric discrimination of sample and prototype distributions. The method is based on recognizing that certain local binary patterns are fundamental properties of local image texture, and their occurrence histogram has proven to be a powerful texture feature. We can regard the scene recognition as a texture classification problem and therefore apply this model to our problem. Timo et al. also extended this approach to be a rotation invariant image descriptor, called Local Binary Pattern Histogram Fourier (LBP-HF)[1]. We try this descriptor to examine whether rotation invariance is suitable for scene recognition. For both LBP and LBP-HF, we use histogram intersection kernel.

[6] Note that we use color for dense SIFT computation and train the feature codebook using SUN database that contains color images only. The 15-scene dataset from [25] contains several categories of grayscale images, which do not have color information. Therefore, the result of our color-based dense SIFT on the 15-scene database (see Fig. 17(a)) is much worse than what is reported in [25].

Texton: A traditional and powerful local image descriptor is to convolve the image with Gabor-like filter bank [41]. Therefore, we use eight oriented even and odd symmetric Gaussian derivative filters and a center surround (difference of Gaussians) filter, as the popular image segmentation framework [2,3]. We use a filter bank containing 8 even and odd-symmetric filters and one center-surround filter at 2 scales. The even-symmetric filter is a Gaussian second derivative, and the odd-symmetric filter is its Hilbert transform. We build a 512 entry universal texton dictionary [26] by clustering responses to the filter bank. For each image we then build a 512-dimensional histogram by assigning each pixel's set of filter responses to the nearest texton dictionary entry. We compute an exponential kernel from χ^2 distances.

Sparse SIFT: As in "Video Google" [42], we build SIFT features at Hessian-affine and MSER [27] interest points. We cluster each set of SIFTs, independently, into dictionaries of 1,000 visual words using k-means. An image is represented by two histograms counting the number of sparse SIFTs that fall into each bin. An image is represented by two 1,000 dimension histograms where each SIFT is soft-assigned, as in [30], to its nearest cluster centers. Kernels are computed using histogram intersection.

SSIM: Self-similarity descriptors [40] are computed on a regular grid at steps of five pixels. Each descriptor is ob-

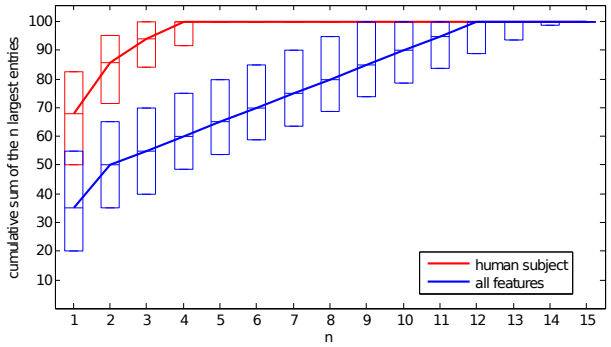


Fig. 21 The cumulative sum of the n largest entries in each row of the confusion matrix. The box indicates the 25th to 75th percentiles at each n.

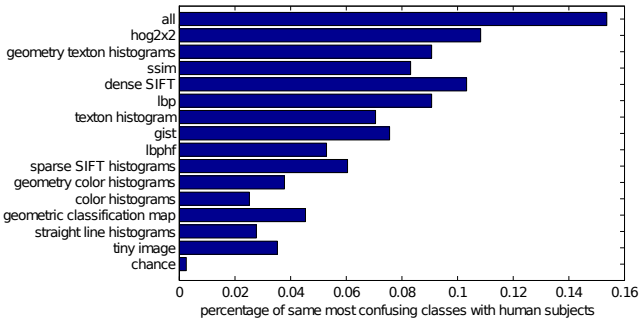


Fig. 22 For each feature, we plot the proportion of categories for which the largest *incorrect* (off-diagonal) confusion is the same category as the largest human confusion.

tained by computing the correlation map of a patch of 5×5 in a window with radius equal to 40 pixels, then quantizing it in 3 radial bins and 10 angular bins, obtaining 30 dimensional descriptor vectors. The descriptors are then quantized into 300 visual words by k-means. After that, kernels are computed from spatial histograms at three levels using exponential χ^2.

Tiny Image: The most trivial way to match scenes is to compare them directly in color image space. Reducing the image dimensions drastically makes this approach more computationally feasible and less sensitive to exact alignment. This method of image matching has been examined thoroughly by Torralba et al. [45] for the purpose of object recognition and scene classification. Inspired by this work we use 32 by 32 color images as one of our features. Images are compared with an exponential χ^2 kernel.

Line Hists: We find straight lines from Canny edges using the method described in Video Compass [23]. For each image we build two histograms based on the statistics of detected lines– one with bins corresponding to line angles and one with bins corresponding to line lengths. We use a histogram intersection kernel to compare these unnormalized histograms. This feature was used in [20].

Geo Map: We compute the geometric class probabilities for image regions using the method of Hoiem et al. [21]. We use only the ground, vertical, porous, and sky classes because they are more reliably classified. We reduce the probability maps for each class to 8×8 and use an RBF kernel. This feature was used in [20].

Geo Texton & Geo Color: Inspired by "Illumination Context" [24], we build color and texton histograms for each geometric class (ground, vertical, porous, and sky). Specifically, for each color and texture sample, we weight its contribution to each histogram by the probability that it belongs to that geometric class. These eight histograms are compared with χ^2 distance.

Overfeat: Overfeat [39] is a state-of-the-art Convolutional Neural Network (CNN) for feature learning. Note that deep CNN algorithms typically require a lot of training data due to the huge number of parameters for learning. We cannot train these models using a small number of images (e.g. 1 image per category as our curves shown in Fig. 17). Therefore, we use the off-the-shelf Overfeat features learned from ImageNet dataset [10]. We normalize the feature and use a χ^2 kernel to train a Support Vector Machine with these learned ImageNet features [31].

5.2 Scene categorization

For comparison with previous works, we show results on the 15 scene categories dataset [29,25,16] in Fig. 17(a). The performance on the 397-category SUN database is shown in Fig. 17(b). For each feature, we use the same set of training and testing splits. For trials with fewer training examples, the testing sets are kept unchanged while the training sets are decreased. The "All" classifier is built from a weighted sum of the kernels of the individual features. The weight of each constituent kernel is proportional to the fourth power of its individual accuracy determined through cross-validation. The confusion matrix of the "All" combined classifier is shown in Fig. 19. Classification results for selected categories are shown in Fig. 23. The best 397-way scene classification performance with all features, 42.7%, is still well below the human performance of 68%. In Fig. 20 we examine the categories for which human and machine accuracy is most similar and most dissimilar. It is interesting to note that with increasing amounts of training data, the performance improvement is more pronounced with the SUN dataset than the 15 scene dataset, probably because the performance for the 15-scene categorization is saturating, while the tasks for 397-way classification is more challenging and more training data would be very helpful. The results in Fig. 17 also show that the combination of the hand-designed features have comparable performance with Overfeat learned from ImageNet. Donahue et al. [11] report slightly better results

more result at `http://vision.princeton.edu/projects/2010/SUN/classification397.html`

Fig. 23 Selected SUN scene classification results using all features.

Fig. 24 Most confused categories for the 397 categories classification task.

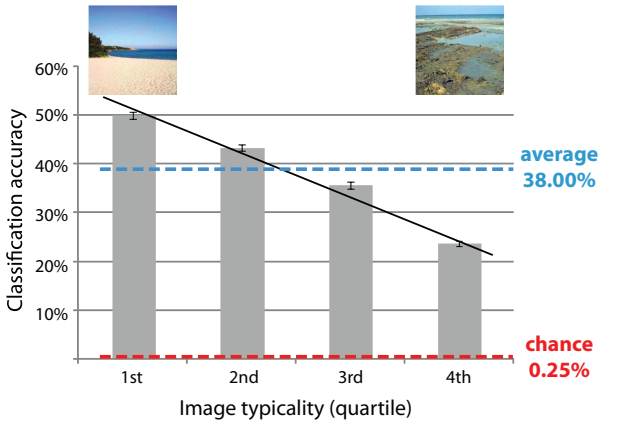


Fig. 25 Performance of the SVM classifier as a function of image typicality. Images are sorted according to their typicality score from least typical (4th quartile) to most typical (1st quartile).

on the SUN database (40.94%) with another deep convolutional network algorithm, DeCAF, trained on the ImageNet dataset. The best performance on this benchmark that we are aware of is based on Fisher vector [38], where they achieve impressive 47.2% accuracy using 50 images per category for training.

Also, humans have fewer confusions than any of the descriptors (the errors concentrate among fewer confusing categories). This is shown in Fig. 21. The plot shows the median value of the 397 cumulative sums of the n largest entries in each row of the confusion matrix. For n=1, the value corresponds to the median of the largest value of each row of the confusion matrix. For n=2, the value corresponds to the median of the sums of the two largest values of each row, and so on. Humans have far fewer confusing categories than the best performing descriptor. For humans, the 3 largest entries in each row of the confusion matrix sum to 95%, while the "all feature" SVM needs 11 entries to reach 95%. Note that this analysis does not relate directly to accuracy – a method might have relatively few entries in its confusion matrix, but they could all be wrong. In Fig. 24 and Fig. 22 we examine the similarity in scene classification *confusions* between humans and machines. The better performing features not only tend to agree with humans on correct classifications, they also tend to make the same mistakes that humans make.

To study indoor-vs-outdoor classification, we use the scene hierarchy (Section 2 and Fig. 3) to divide the 397 scene categories into two classes: indoor and outdoor. Then, we evaluate the same set of features and report their performance in Fig. 17(c). There are two categories, promenade deck and ticket booth, that are considered to be both indoor and outdoor in our scene hierarchy. Therefore, we exclude these two categories in our evaluation. We can see that the order of performance for different features are quite different, probably due to the great difference of the task compared to 397-way scene classification. The overall performance is 94.2%, which suggests that this task is nearly solved. Fig. 18 shows some errors made by the classifier. We can see that the definition of indoor-vs-outdoor is ambiguous in some places, such as images which show both indoor elements and outdoor scenes through a window or a door. Therefore, the accuracy on this task might be actually higher than what the evaluation suggests. Furthermore, we train a 16-way classifier at the second level of the scene hierarchy shown (in Fig. 3), and the results are shown in Fig. 17(d).

As might be expected, performance on this mid-level task is lower than performance on the indoor-outdoor discrimination, but higher than performance on the 397-category classification. The difficulty of a scene classification task increases as the categories become more fine-grained: there are more categories available and it is harder to find a combination of features which reliably distinguishes them. Accuracy on the 16-way mid-level scene classification on the SUN database is also lower than the 15-way classification on the 15-scene database, even though these tasks involve nearly the same number of categories. This is because the mid-level categories used in the SUN hierarchy are heterogeneous and not always well-separated; examples of these categories include "shops and restaurants", "offices, labs, construction, and factories (workspaces)", "rooms in a home", and "educational, religious, or cultural spaces". This is a contrast to the "basic level" categories of the 15-scene database, which are, by comparison, homogeneous and very clearly separated (e.g. "office", "city", "forest").

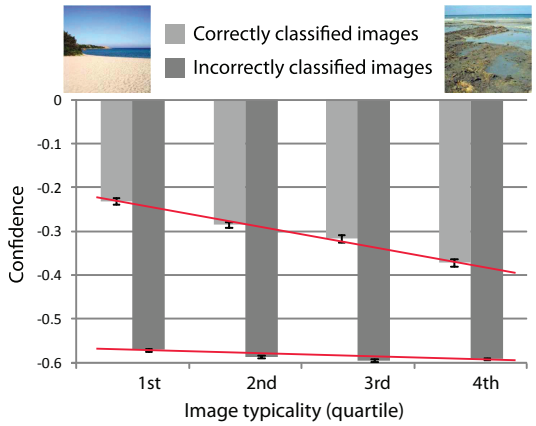


Fig. 26 Confidence of the SVM classifier as a function of image typicality. Images are sorted according to their typicality score from least typical (4th quartile) to most typical (1st quartile).

5.3 Recognition and typicality of scenes

What is the scene classification performance as a function of scene typicality? Fig. 25 shows that classification performance for individual images varies with their typicality scores, using the combined kernel with all features: the most typical images were classified correctly about 50% of the time, and the least typical images were classified correctly only 23% of the time. Images were divided into four groups corresponding to the four quartiles of the distribution of typicality scores across the database. These groups contained 5020, 4287, 5655, and 4908 images (groups are listed in order from fourth quartile – lowest typicality – to first quartile). A one-way ANOVA comparing these quartile groups shows a significant effect of image typicality quartile on classification accuracy [7]; Bonferroni-corrected post-hoc tests show that the differences between each quartile are significant.

Image typicality is also related to the confidence of the SVM classifier. The confidence reflects how well the classifier believes the image matches its assigned category[8]. Fig. 26 shows the SVM confidence as a function of image typicality for correctly- and incorrectly-classified images. Confidence increases with increasing typicality, but this pattern is stronger in correctly-classified images. [9]

In summary, scenes which people rate as more typical examples of their category are more likely to be correctly classified by the algorithms based on global image descriptors. Although we cannot claim that the features used in these algorithms are the same features which humans use to perform the same classification task, this nevertheless indicates that more typical examples of a scene category contain more of the diagnostic visual features that are relevant for scene categorization. It also shows that typical images of scene categories can be reliably identified by state of the art computer vision algorithms.

[7] $F(3,19846) = 278, p < .001$.

[8] Due to the difficulty of the one-versus-all classification task, confidence was low across all classifications, and even correctly-classified images had average confidence scores below zero.

[9] A 4×2 ANOVA gives significant main effects of image typicality ($F(3,19842) = 79.8, p < .001$) and correct vs. incorrect classification ($F(1,19842) = 6006, p < .001$) and a significant interaction between these factors ($F(3,19842) = 43.5, p < .001$).

Fig. 27 These scenes of a beach, a village, and a river are all from a single image (Fig. 29).

5.4 Scene detection

Imagine that you are walking in a street: scene recognition will tell you that you are in the street, and object recognition will allow you to localize people, cars, tables, etc. But there are additional detection tasks that lie in between objects and scenes. For instance, we want to detect restaurant terraces, or markets, or parking lots. These concepts also define localized regions, but they lack the structure of objects (a collection of parts in a stable geometric arrangement) and they are more organized than textures.

Here, we refer to these scenes within scenes as "subscenes" to distinguish that them from global scene labels. A single image might contain multiple scenes (e.g. Fig. 27 and 29), where a scene is a bounded region of the environment that has a distinct functionality with respect to the rest. For instance, a street scene can be composed of store fronts, a restaurant terrace, and a park. To be clear, we are not describing a part-based model of scenes – subscenes are full fledged, potentially independent scenes that create their own context. The objects and the actions that happen within subscenes have to be interpreted in the framework created by each local scene, and they might be only weakly related to the global scene that encompasses them. However, the dominant view in the literature is that one image depicts one scene category (with some exceptions [49,7]). Also, while our approach takes scene representations and makes them more local, complementary work from [37] comes from the other direction and detects object arrangements called "visual phrases". Furthermore, scene detection is also related to scene viewpoint recognition [51].

As scenes are more flexible than objects, it is unclear what the right representation will be in order to detect them in complex images. Here, we use the scene classification framework to directly classify image crops into subscene categories. We refer to this task as "scene detection". Our terminology is consistent with the object detection literature

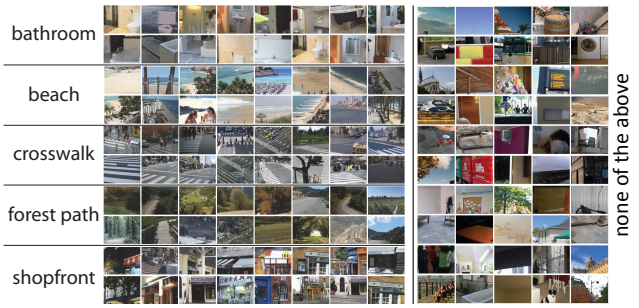


Fig. 28 Example subscene patches annotated by observers. Left: random samples of consistently annotated crops for several classes. Right: samples of crops that are not scenes, having been annotated "none of the above".

Fig. 29 There are many complementary levels of image understanding. One can understand images on a continuum from the *global* scene level (left) to the *local* object level (right). Here, we introduce the intermediate concept of *local subscenes* (middle), and define a task called Scene Detection.

(a) beach (b) village (c) harbor&dock (d) river

Fig. 30 Masks generated from the labels of Mechanical Turk workers. The brightness of each region is proportional to the degree of consistency in annotations.

[14] where object *classification* or *recognition* involves classifying entire images, while object *detection* requires localizing objects within an image.

We choose a set of 47 scene categories that commonly co-occur within images (e.g. ocean with coast, alley with crosswalk, shelving with office area, river with harbor, village with plaza, etc.), and use our SUN database as training examples. It may seem odd to train a localized detector from entire images, but we do not expect scenes to vary significantly when they photographed as single images or in a larger context: this is similar to training an object detector with more close-up views of objects.

To test scene detection, we create a database of 1000 images that have spatially localized subscenes (called the Scene Detection database). We use pairs of category related keywords to search for relevant images from online sources such as Flickr, Picasa, Google, and Bing. We manually filter the results to ensure that images 1) are large enough such that crops are still of sufficient resolution 2) depict relevant scenes and 3) are not distorted with respect to lighting, viewpoint, or post-processing.

Unlike objects, scenes don't necessarily have clear, segmentable boundaries. The key idea to reliably annotate ground truth subscenes is to examine only the local image region without distraction from the surrounding context (for instance "sibling" or "parent" scenes of different categories). We do this by cropping out image regions at 3 different scales and assigning scene labels to those crops in isolation. An annotator does not know what the entire image looks like (the crop may in fact be almost the entire image, or it may be a small region). We use a somewhat sparse set of 30 partially overlapped crops for each image in the Subscene Database. For each crop, we annotate its categories in an Amazon Mechanical Turk task performed by three different workers.

The annotators are presented with visual examples and text guidelines for all 47 categories, as well as a "none of the above" option for crops that do not fit any category or are not scenes. We gave the workers the following instructions: Choose a place name from the list that best describes the image, and if the image does not fit any description, select "none of the above". To ensure accurate annotation, we had each crop labeled three times. We define an annotation to be "consistent" if at least two of the three annotators agreed *and* no annotators selected "none of the above". 962 different Mechanical Turk workers annotated subscenes, taking an average of 30 seconds per annotation (or two and a half minutes to label the five subscenes each task presents). Fig. 30 and Fig. 28 visualize the annotation results. The annotations are generally quite accurate, especially where the workers were consistent with each other. After the annotation, an author went through the results and removed any obvious mistakes. The error rate in the experiment was less than one percent.

We use the same algorithms as scene classification to train a detector. All testing is done on the scene detection dataset. To evaluate the detection performance, we use a simple criterion to plot the precision-recall curves: a detection is considered correct if the predicted label matches the human label. To generate testing images, we use sparse sliding windows because there usually aren't exact boundaries for subscenes. We uniformly generate about 30 crops at 3 different scales – the same windows that we have ground truth annotations for. We do not perform non-maximum suppression as in the object domain. Therefore, for each crop, there are 47 class prediction scores. We take the negative of the minimum score among all 47 class prediction scores as the score for the "non-scene" class, i.e. the 48th class.

In order to evaluate the detection performance, we use a simple criteria. As we have human labels for all the possible crops that we will consider during the detection stage, we consider a detection correct if the predicted label matches the human label. This task is harder than the recognition task as many crops are not considered to contain clearly

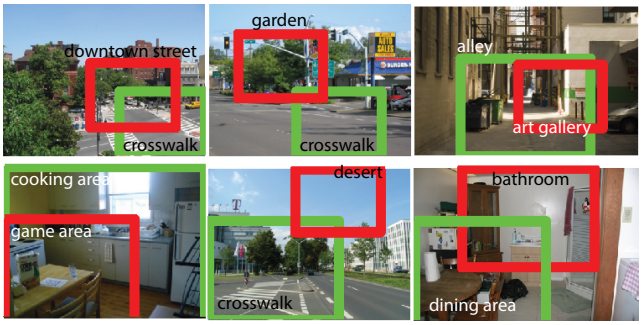


Fig. 31 Subscene localization result. This figure shows the two most confident detections for several images. The detections with a red bounding box are incorrect detections and the green bounding boxes denote correct detections.

Fig. 32 Scene detection results for some classes and their recall precision curves. Green boxes indicate correct detection, and red boxes indicate wrong detection.

defined scenes. In the test set there are a total of 26,626 crops, and 12,145 of those correspond to non-scenes. Fig. 31 shows some examples with multiple scenes detected. To evaluate the performance, we use precision-recall curves: for each class and for each decision threshold we compute how many of the crops labeled according to each category are retrieved and with what precision. The average precision over all classes for the combined model is 19.224. Fig. 32 shows examples of precision-recall curves for a few selected classes (performances are typical). These results show that although it is hard to specify the spatial extent of a subscene, the scene detector is still able to detect areas that are consistent with human annotation. Such results would be useful for general scene parsing tasks, such as object detection.

6 Conclusion

To advance the field of scene understanding, we need datasets that encompass the richness and variety of environmental scenes and knowledge about how scene categories are organized and distinguished from each other. In this work, we propose a large dataset of 908 scene categories. We evaluate state-of-the-art algorithms, and study several questions related to scene understanding. All images, object labels, scene definitions, and other data, as well as the source code, are publicly available online. Future works include going beyond 2D images and reasoning about scenes in 3D [53, 54, 43].

Acknowledgements We thank Yinda Zhang for help on the scene classification experiments. This work is funded by Google Research Award to J. X., NSF grant 1016862 to A.O, NSF CAREER Award 0747120 to A.T., NSF CAREER Award 1149853 to J.H, as well as ONR MURI N000141010933, Foxconn and gifts from Microsoft and Google. K.A.E was funded by a NSF Graduate Research fellowship.

References

1. Ahonen, T., Matas, J., He, C., Pietikäinen, M.: Rotation invariant image description with local binary pattern histogram fourier features. In: SCIA (2009)
2. Arbelaez, P., Fowlkes, C., Martin, D.: The berkeley segmentation dataset and benchmark (2007)
3. Arbelaez, P., Maire, M., Fowlkes, C., Malik, J.: Contour detection and hierarchical image segmentation. PAMI (2011)
4. Barnard, K., Duygulu, P., de Freitas, N., Forsyth, D., Blei, D., Jordan, M.: Matching words and pictures. JMLR (2003)
5. Barriuso, A., Torralba, A.: Notes on image annotation (2012)
6. Biederman, I.: Recognition-by-components: A theory of human image understanding. Psychological Review (1987)
7. Boutell, M.R., Luo, J., Shen, X., Brown, C.M.: Learning multi-label scene classification. Pattern Recognition (2004)
8. Bunge, J., Fitzpatrick, M.: Estimating the number of species. Journal of the American Statistical Association (1993)
9. Dalal, N., Triggs, B.: Histogram of oriented gradient object detection. In: CVPR (2005)
10. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: A Large-Scale Hierarchical Image Database. In: CVPR (2009)
11. Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., Darrell, T.: Decaf: A deep convolutional activation feature for generic visual recognition. arXiv (2013)
12. Ehinger, K.A., Xiao, J., Torralba, A., Oliva, A.: Estimating scene typicality from human ratings and image features. In: CogSci (2011)
13. Epstein, R., Kanwisher, N.: A cortical representation of the local visual environment. Nature (1998)
14. Everingham, M., Gool, L.V., Williams, C.K.I., ands, J.W., Zisserman, A.: The pascal visual object classes (voc) challenge. IJCV (2009)
15. Fei-Fei, L., Fergus, R., Perona, P.: Learning generative visual models from few training examples. In: CVPR Workshop on Generative-Model Based Vision (2004)
16. Fei-Fei, L., Perona, P.: A bayesian hierarchical model for learning natural scene categories. In: CVPR (2005)
17. Fellbaum, C.: Wordnet: An Electronic Lexical Database. Bradford Books (1998)
18. Felzenszwalb, P., Girshick, R., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part based models. PAMI (2007)

19. Griffin, G., Holub, A., Perona, P.: Caltech-256 object category dataset. Tech. rep. (2007)
20. Hays, J., Efros, A.A.: IM2GPS: estimating geographic information from a single image. In: CVPR (2008)
21. Hoiem, D., Efros, A., Hebert, M.: Recovering surface layout from an image. IJCV (2007)
22. Jolicoeur, P., Gluck, M., Kosslyn, S.: Pictures and names: Making the connection. Cognitive Psychology (1984)
23. Kosecka, J., Zhang, W.: Video compass. In: ECCV (2002)
24. Lalonde, J.F., Hoiem, D., Efros, A.A., Rother, C., Winn, J., Criminisi, A.: Photo clip art. SIGGRAPH (2007)
25. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: CVPR (2006)
26. Martin, D., Fowlkes, C., Tal, D., Malik, J.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: Proc. ICCV (2001)
27. Matas, J., Chum, O., Urban, M., Pajdla, T.: Robust wide-baseline stereo from maximally stable extremal regions. IVC (2004)
28. Ojala, T., Pietikäinen, M., Mäenpää, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. PAMI (2002)
29. Oliva, A., Torralba, A.: Modeling the shape of the scene: a holistic representation of the spatial envelope. IJCV (2001)
30. Philbin, J., Chum, O., Isard, M., Sivic, J., Zisserman, A.: Lost in quantization: Improving particular object retrieval in large scale image databases. In: CVPR (2008)
31. Razavian, A.S., Azizpour, H., Sullivan, J., Carlsson, S.: CNN features off-the-shelf: an astounding baseline for recognition. arXiv (2014)
32. Renninger, L., Malik, J.: When is scene recognition just texture recognition? Vision Research (2004)
33. Rosch, E.: Natural categories. Cognitive Psychology (1973)
34. Rosch, E., Mervis, C., Gray, W., Johnson, D., Boyes-Braem, P.: Basic objects in natural categories. Cognitive Psychology (1976)
35. Russakovsky, O., Deng, J., Huang, Z., Berg, A.C., Fei-Fei, L.: Detecting avocados to zucchinis: what have we done, and where are we going? In: ICCV (2013)
36. Russell, B.C., Torralba, A., Murphy, K.P., Freeman, W.T.: Labelme: a database and web-based tool for image annotation. IJCV (2008)
37. Sadeghi, M.A., Farhadi, A.: Recognition using visual phrases. In: CVPR (2011)
38. Sánchez, J., Perronnin, F., Mensink, T., Verbeek, J.: Image classification with the fisher vector: Theory and practice. IJCV (2013)
39. Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., LeCun, Y.: Overfeat: Integrated recognition, localization and detection using convolutional networks. arXiv (2013)
40. Shechtman, E., Irani, M.: Matching local self-similarities across images and videos. In: CVPR (2007)
41. Shotton, J., Winn, J., Rother, C., Criminisi, A.: TextonBoost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In: ECCV (2006)
42. Sivic, J., Zisserman, A.: Video data mining using configurations of viewpoint invariant regions. In: CVPR (2004)
43. Song, S., Xiao, J.: Sliding Shapes for 3D object detection in RGB-D images. In: ECCV (2014)
44. Spain, M., Perona, P.: Some objects are more equal than others: measuring and predicting importance. In: ECCV (2008)
45. Torralba, A., Fergus, R., Freeman, W.T.: 80 million tiny images: a large database for non-parametric object and scene recognition. PAMI (2008)
46. Torralba, A., Murphy, K., Freeman, W., Rubin, M.: Context-based vision system for place and object recognition. In: ICCV (2003)
47. Tversky, B., Hemenway, K.: Categories of environmental scenes. Cognitive Psychology (1983)
48. Vedaldi, A., Fulkerson, B.: VLFeat: An open and portable library of computer vision algorithms (2008)
49. Vogel, J., Schiele, B.: A semantic typicality measure for natural scene categorization. In: DAGM (2004)
50. Vogel, J., Schiele, B.: Semantic model of natural scenes for content-based image retrieval. IJCV (2007)
51. Xiao, J., Ehinger, K., Oliva, A., Torralba, A.: Recognizing scene viewpoint using panoramic place representation. In: CVPR (2012)
52. Xiao, J., Hays, J., Ehinger, K., Oliva, A., Torralba, A.: SUN database: Large-scale scene recognition from abbey to zoo. In: CVPR (2010)
53. Xiao, J., Owens, A., Torralba, A.: SUN3D: A database of big spaces reconstructed using SfM and object labels. In: ICCV (2013)
54. Zhang, Y., Song, S., Tan, P., Xiao, J.: PanoContext: A whole-room 3D context model for panoramic scene understanding. In: ECCV (2014)