PanoContext: A Whole-room 3D Context Model for Panoramic Scene Understanding

Yinda Zhang Shuran Song Ping Tan[†] Jianxiong Xiao

Princeton University [†]Simon Fraser University
http://panocontext.cs.princeton.edu

Abstract. The field-of-view of standard cameras is very small, which is one of the main reasons that contextual information is not as useful as it should be for object detection. To overcome this limitation, we advocate the use of 360° full-view panoramas in scene understanding, and propose a whole-room context model in 3D. For an input panorama, our method outputs 3D bounding boxes of the room and all major objects inside, together with their semantic categories. Our method generates 3D hypotheses based on contextual constraints and ranks the hypotheses holistically, combining both bottom-up and top-down context information. To train our model, we construct an annotated panorama dataset and reconstruct the 3D model from single-view using manual annotation. Experiments show that solely based on 3D context without any image-based object detector, we can achieve a comparable performance with the state-of-the-art object detector. This demonstrates that when the FOV is large, context is as powerful as object appearance. All data and source code are available online.

1 Introduction

Recognizing 3D objects from an image has been a central research topic since the computer vision field was established [1]. While witnessing the rapid progress on bottom-up object detection methods [2–6] in the past decade, the improvement brought by the top-down context cue is rather limited, as demonstrated in standard benchmarks (e.g. PASCAL VOC[3]). In contrast, there are strong psychophysical evidences that context plays a crucial role in scene understanding for humans [7, 8].

We believe that one of the main reasons for this gap is because the field of view (FOV) for a typical camera is only about 15% of that of the human vision system[1]. This problem is exemplified in Fig. 1. The narrow FOV hinders the context information in several ways. Firstly, a limited FOV sees only a small fraction of all scene objects, and therefore, observes little interplay among them. For example, on average, there is only 1.5 object classes and 2.7 object instances per image in PASCAL VOC. Secondly, the occurrence of an object becomes unpredictable with a small FOV. For example, a typically bedroom should have at least one bed, which can serve as a strong context cue. But in a bedroom picture of small FOV (Fig. 1), there might or might not be a bed,

[1] The approximate FOV of a single human eye is about 95°. Two eyes give us almost 180° FOV. Considering the movement of eyeballs (head rotation excluded, peripheral vision included), the horizontal FOV of the human vision system is as high as 270°. However, the FOV of a typical camera is much smaller. For example, on standard full-frame cameras, the horizontal FOV is only 39.6° (or 54.4°) with a standard 50mm lens (or with a 35mm wide-angle lens).

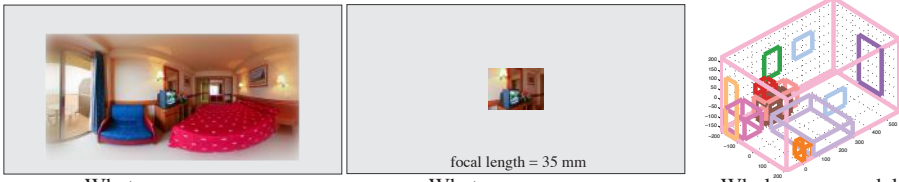


Fig. 1. Comparison of field-of-view. A camera with narrow FOV might not see a bed in a bedroom which complicates the context model.

depending on the direction the camera looks at. Given a much limited FOV, it is unfair to ask computer vision algorithms to match the performance of human vision. Therefore, we advocate the use of panoramic images for scene understanding, which nowadays can be easily obtained by camera arrays (e.g. Google Streetview), special lenses (e.g. `0-360.com`), smartphones (e.g. `cycloramic.com`) or automatic image stitching algorithms (e.g. [9–11]).

In this paper, we present a whole-room 3D context model to address the indoor scene understanding problem from a single panorama. The output is a 3D cuboid room layout with recognized scene objects represented by their 3D bounding boxes. An example of input and output are provided in Fig. 2. Our method consists of two steps: bottom-up hypotheses generation and holistic hypotheses ranking. It starts by generating hypotheses for the room layout and object bounding boxes in a bottom-up fashion using a variety of image evidences, e.g. edge, segmentation, and normal direction estimation. 3D scene hypotheses are formed from these hypotheses guided by context. A trained Support Vector Machine (SVM) [12] ranks these 3D scene hypotheses and chooses the best one. Finally, we locally refine good scene hypotheses by further maximizing their SVM scores. The SVM is trained utilizing both image information and room structure knowledge from our training data, which consists of high-resolution panorama images with detailed object annotations and the 3D ground truth reconstructed using the 2D annotations.

In a panorama, characteristic scene objects such as beds and sofas are usually visible despite occlusion, so that we can jointly optimize the detection of room layout and object to exploit the contextual information in full strength. The whole-room contextual information is critical in many key steps of our system. During *hypothesis generation*, the object categories are predicted based on its relative location in the room. We sample the number of object instances in a room according to the typical distribution of each object category, guided by the pairwise position relationship among objects. During *hypothesis ranking*, we firstly align each hypothesis with the 3D rooms from the training set to tell if it is valid. This non-parametric room alignment captures high order relationship among all objects, which cannot be represented well by pairwise constraints. Secondly, we also build a room model for each hypothesis. This room model includes color and texture statistics for the foreground and background. Since we know all the objects and room layout in 3D, we can calculate these statistics easily from image regions unoccluded by objects.We use this model to judge how well a hypothesis explains the image evidences. Thirdly, we reconstruct each room layout and object hypothesis to 3D space by assuming no floating object. A wrong room layout hypothesis is typi-

Input: a single-view panorama Output: object detection Output: 3D reconstruction

Fig. 2. Input and output. Taken a full-view panorama as input, our algorithm detects all the objects inside the panorama and represents them as bounding boxes in 3D, which also enables 3D reconstruction from a single-view.

cally ranked low by the SVM, since it often produces unreasonable 3D bounding boxes of objects. This implicit 3D interaction between objects and room layout enables us to identify many bad hypotheses. During *final adjustment*, we also use the object number distribution and pairwise context model to guide the search.

As demonstrated in our experiments, we can recognize objects using only 3D contextual information (without a classifier to discriminate object categories based on image feature), and still achieve a comparable performance with the state-of-the-art object detector [2], which learns a mapping from image region feature to object category. This shows that context is as powerful as object appearance and much more useful than we previously thought. The root of context model being under-utilized is partially because the regular FOVs are too small.

In the following section, we will describe our algorithm in greater details. We will also talk about the construction of a 3D panorama data set and present experiments to evaluate the algorithm in Sec. 3. In Sec. 5, we will discuss the relation of our proposed method with existing ones.

2 PanoContext: A whole-room 3D context model

As shown in Fig. 2, our input is a panorama covering 360° horizontal and 180° vertical FOV represented in equirectangular projection. Our output is a 3D box representation of the scene. We adopt the Manhattan world assumption, assuming that the scene consists of 3D cuboids aligned with three principle directions[2].

Our method first generates whole-room hypotheses and then ranks them holistically. The challenge for hypotheses generation is to maintain high recall using a manageable number of hypotheses, while the challenge for holistic ranking is to have high precision. To generate hypotheses, we first estimate vanishing points by Hough Transform based on the detected line segments (Sec. 2.1). We then generate 3D room layout hypotheses from line segments and verify them with the computed geometric context and orientation map on the panorama (Sec. 2.2). For objects, we generate 3D cuboid hypotheses using rectangle detection and image segmentation (Sec. 2.3). Next, we use sampling to generate whole-room hypotheses, each of which has a 3D room and multiple 3D objects

[2] We focus on indoor scenes only, although our algorithm may be generalized to outdoor scenes as well. We assume an object can either stand on the ground, sit on another object, or hang on a wall (i.e. no object floats in space). We also assume that the height of camera center is 1.6 meters away from the floor to obtain a metric 3D reconstruction.

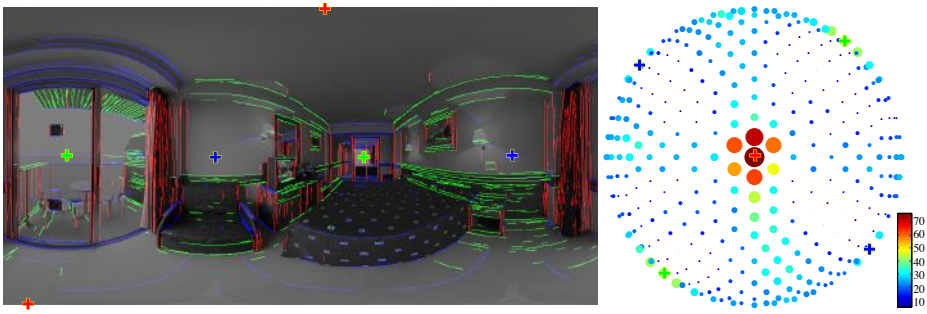


Fig. 3. Hough transform for vanishing point detection. The **left** image shows the detected edges and vanishing points. The colors indicate the edge directions. The **right** image shows the votes on each bin of the half sphere. The sizes and colors both indicate the number of votes.

inside (Sec. 2.4). To choose the best hypothesis that is coherent with image evidence and structurally meaningful (i.e. satisfying all context constraints), we extract various features and train a SVM to rank these hypotheses holistically (Sec. 2.5). Finally, we locally adjust the top hypothesis and search for a solution that maximizes the SVM score by adding, deleting and swapping an object.

2.1 Vanishing point estimation for panoramas

We detect line segments on the panorama and use them to vote for the vanishing directions (Fig. 3). To take full advantage of previous line segment detection working on standard camera photo, we convert a panorama image to a set of perspective images, and run the state-of-the-art Line Segment Detection (LSD) algorithm [13] in each perspective image, and warp all detected line segments back to the panorama.

A line segment in 3D space corresponds to a section of a great circle on the panorama sphere and displays as a curve in panorama image. For each line $\mathbf{l}$, we use $\mathbf{n}$ to denote the normal direction of the plane where its great circle lies in. The vanishing direction $\mathbf{v}$ associated with the line $\mathbf{l}$ should be perpendicular to $\mathbf{n}$. We use a Hough Transform [14] to find all vanishing directions. We uniformly divide the unit sphere into bins by recursively dividing triangles of a icosahedron. A line segment $\mathbf{l}$ will vote for all bins whose center $\mathbf{n}_b$ satisfies $\mathbf{n}_b \cdot \mathbf{n} = 0$. We then find three mutually orthogonal bins with maximal sum of votes as three vanishing directions. After that, we snap all line segments to align with their vanishing directions.

2.2 Room layout hypothesis generation

Because the room layout is essential to generate good object hypotheses in 3D, we first obtain some good room layouts to reduce the burden of 3D object detection in the next step. We randomly generate many room layout hypotheses and keep those consistent with a pixel-wise surface normal direction estimation on panorama.

A 3D room layout can be generated by sampling line segments as room corners [15]. Geometrically, five lines determine a cuboid in 3D space except some degenerative

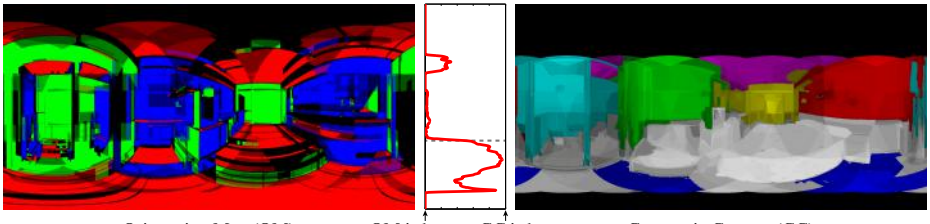


Fig. 4. OM vs. GC. Here shows the OM and GC for the panorama image in Fig. 1. The curve at the center shows the accuracy comparison between OM and GC as the vertical view angle changes (data is from all training images). We can clearly see that OM is better at the upper part while GC is better at the lower part, and there is a clear threshold to combine them.

cases. We classify each line segment with two labels from top/bottom, front/back, and right/left according to its association to the vanishing directions, and randomly sample five non-degenerative lines to form a room layout hypothesis. To reduce the number of hypotheses while keeping the good ones, we use the surface normal consistency with a pixel-wise surface direction estimation from panorama to rank these hypotheses and choose the top 50 (since the recall starts to saturate around 50 in Fig. 11).

Orientation Map (OM) [16] and Geometric Context (GC) [15] provide pixel-wise surface normal estimation for ordinary perspective images. We convert a panorama into several overlapping perspective images, and apply OM and GC on these images respectively and project results back to the panorama. From our training data with manually marked ground truth wall orientations, we observe that GC provides better normal estimation at the bottom (probably because the model was trained using images looking slightly downwards), and OM works better at the top half of an image (probably less cluttered.), as shown in Fig. 4. Therefore, we combine the top part of OM and the bottom part of GC to evaluate the room layout. As can be seen from Fig. 11(left), the recall rate is significantly improved by combining OM and GC. Fig. 7 shows some good room layout hypotheses, which are generally very close to the ground truth.

2.3 3D object hypotheses generation

After generating a set of good 3D room layout hypotheses, the next step is to generate 3D cuboid hypotheses for major objects in the room. To obtain high recall for hypotheses generation, we use two complementary approaches: a detection-based method to build cuboid from detected rectangular surface and a segmentation-based method by fitting cuboid to 2D projections.

Detection-based cuboid generation: We project the input panorama orthographically to six axis-aligned views, and run a rectangle detector in each projection respectively (Fig. 5(top)). Our rectangle detector is similar as Deformable Part Model [2] but without spring-like constraints. We define a part at each corner and the middle of each edge of the rectangle. We use the SUN primitive dataset [17] containing 382 annotated cuboid images, and transform each cuboid surface to an axis aligned rectangle to train each part detector independently. During testing, we first compute the response maps of all part

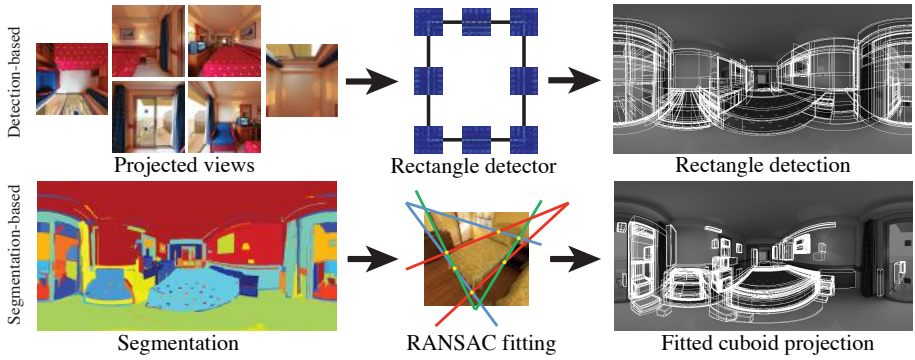


Fig. 5. Two ways to generate object hypotheses: detection and segmentation.

detectors, and sum up them according to the models. We set a low threshold to ensure high recall. We then generate cuboid hypotheses from the 3D rectangles.

Segmentation-based cuboid generation: Some objects, such beds and sofas, do not have strong edges, and cannot be reliably detected by the rectangle detection. Therefore, we generate additional cuboid hypotheses from image segmentation (Fig. 5(bottom)) by selective search [4]. Specifically, for each segment, we evaluate how well its shape can be explained by the projection of a cuboid. We create many cuboids by randomly sampling 6 rays at the segment boundary passing through the three vanishing points. Among these cuboids, we choose the best one whose projection has the largest intersection over union score with the segment.

2.4 Whole-room scene hypotheses generation

After obtaining a hypothesis pool for room layout and objects, we generate a pool of whole-room hypotheses, each consisting of a room layout with several cuboid objects inside. To achieve high recall with a manageable number of hypotheses, we classify the semantic type of each cuboid and use pairwise context constraints to guide our sampling.

Semantic label: Intuitively, the semantic object type is strongly correlated with the cuboid shape and its 3D locations in the room. We train a random forest classifier to estimate the semantic type of a cuboid according to its size, aspect ratio and relative position in the room. And we achieve the multiple-label classification accuracy at around 70%. This shows that the context between room and objects is very strong.

Pairwise constraint: There are strong pairwise context constraints between scene objects, e.g. nightstands are usually nearby a bed, and a TV set often faces a bed. For two object types, we collect all instances of the object pair, one for each type, coexisting in a room from our training database. We then take the displacement between two objects in a pair as a sample, and capture the pairwise location constraint by all collected samples. Such a set of samples are plotted in Fig. 6(b). When testing the validity of a pair of objects, we compute their displacement and search for the K nearest neighbors in the sample set. The mean distance to the K nearest neighbors will be transferred to a probability by a sigmoid function.

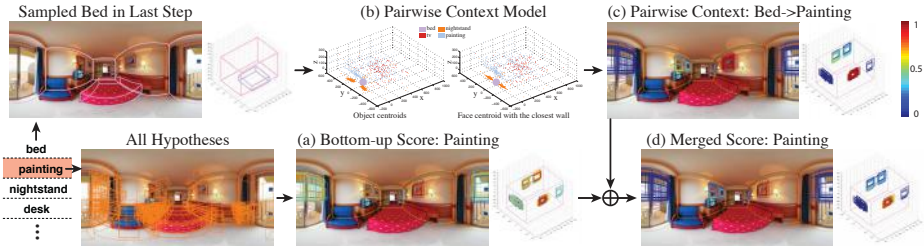


Fig. 6. Example sampling pipeline. Here we show an example of sampling a painting guided by context, given that a bed is already sampled in previous steps. (a) the bottom-up scores of painting hypotheses, (b) pairwise context statistics to show how objects of different categories locates around a bed, (c) the pairwise context constraint from the sampled bed, (d) the scores for merging bottom-up scores and pairwise context.

Whole-room sampling: We generate a complete scene hypothesis as following,

1. Randomly select a room layout according to their consistency with GC and OM (higher consistency with higher probability).
2. Decide the number of instances for each type and the sampling order for instances according to statistic prior. Fig. 6 shows an example of order list on left side.
3. Start from the first object, randomly choose a cuboid according to the bottom up score, e.g. rectangle detection score, semantic classifier score. Hypothesis with higher score would be sampled with higher probability.
4. Go to the next object, we evaluate the pairwise context constraint for all the cuboid hypotheses with all the previously selected objects, and merge it the bottom-up scores. A new object will be randomly selected according to merged scores. For example, the unary bottom-up score is effective in pruning invalid hypotheses (Fig. 6(a)), and pairwise score can further enhance it (Fig. 6(c)). As shown in Fig. 6(d), the rectangles on head of the bed are further highlighted, and those on windows are depressed. We can see the hypotheses around the true painting are all with high score, and thus we have a higher chance to get a correct object.
5. Given all the sampled object so far, repeat the previous step until all the instances have been sampled.

Comparing with completely random sampling, our method can avoid obviously unreasonable scene hypotheses, and thus ensure high recall with a manageable number of samples. Fig. 7 shows some sampling results.

2.5 Data-driven holistic ranking

After generating a long list of whole-room hypotheses, we train a SVM model to rank them and choose the best hypothesis, holistically for the whole-room.

Linear SVM: Our goal is to learn a mapping from a panorama $\mathbf{x}$ to a scene parsing result $\mathbf{y}$. Because $\mathbf{y}$ is a structural output, we formulate the problem as a 0-1 loss structural SVM [18], i.e. a binary linear SVM[3]. We define a feature vector $\mathbf{f}(\mathbf{x}, \mathbf{y})$ for a panorama

[3] We use a 0-1 loss structural SVM because the ranking among the bad hypotheses is unimportant, and we only want to find the best one. Our experiment also shows that it is very slow to train a general structural SVM using the standard

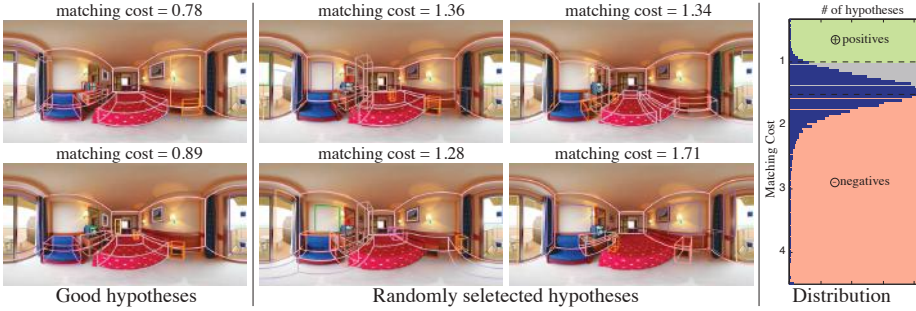


Fig. 7. Whole-room hypotheses. On the left we show some good hypotheses selected based on matching cost. At the center we show some random hypotheses to visualize the hypothesis space.

$\mathbf{x}$ and its hypothesis $\mathbf{y}$. The binary label l indicates whether $\mathbf{y}$ is close enough to the manually annotated ground truth $\mathbf{y}^*$, i.e. $l = [\Delta(\mathbf{y}, \mathbf{y}^*) < \epsilon]$. During training, for each panorama $\mathbf{x}_n$, we sample M hypotheses $\{\mathbf{y}_n^m\}^{m=1,\cdots,M}$. We use all N panorama from our training set to train the binary SVM by MN pairs of $\{\langle \mathbf{f}(\mathbf{x}_n, \mathbf{y}_n^m), l_n^m \rangle\}_{n=1,\cdots,N}^{m=1,\cdots,M}$. Since we typically have hundreds of panoramas and thousands of hypotheses, there are about a million training data for the SVM During testing, the hypothesis with maximal SVM score is chosen as the result.

Matching cost: $\Delta(\mathbf{y}, \mathbf{y}^*)$ measures the difference between a whole-room hypothesis $\mathbf{y}$ and its ground truth $\mathbf{y}^*$. We first register the two scenes by matching their vanishing directions and room centers. For all pair of cuboids of the same semantic type, one from each scene, we compute their distance as the the average 3D distance between corresponding vertices. We then search for the bipartite matching minimizing the distance for each semantic label. $\Delta(\mathbf{y}, \mathbf{y}^*)$ is the sum of all bipartite matching cost plus the constant penalty for unmatched cuboids in both scenes. We use this score to decide the labels for the training data. Because it is hard to find a good threshold, we choose two conservative thresholds to make sure all positives are good and all negatives are bad. We drop all other data in between as we cannot tell their quality reliably.

Holistic features: The feature vector $\mathbf{f}(\mathbf{x}, \mathbf{y})$ is a concatenation of object level feature $\mathbf{f}^{\text{object}}$ and room level feature $\mathbf{f}^{\text{room}}$. Thus, it encodes both bottom-up image evidence and top-down contextual constraints. The relative importance between all the information is learned by the SVM in a data-driven way using the training data. $\mathbf{f}^{\text{object}}$ measures the reliability of each single object. On each object hypothesis, rectangle detector score, segmentation IOU score, sum/mean/std on each channel of OM and GC, entropy of color distribution, and 2D projected size will be extracted and concatenated into a column vector. We concatenate the sum/mean/max/min of features of all instances in a category as the feature for the category. For categories with no object, we set the features zeros. We concatenate the features for all object categories to a single vector as $\mathbf{f}^{\text{object}}$. Since the number of categories is fixed, the total dimension is also fixed.

Non-parametric room alignment: The room level feature $\mathbf{f}^{\text{room}}$ checks whether the hypothesized structure, i.e. the room layout and the arrangement of all objects, can be

cutting plane algorithm, and the general structural SVM is very sensitive to the loss function. A 0-1 loss structural SVM is basically a binary linear SVM that can be trained efficiently and is robust to the loss-function.

found in reality. We propose a non-parametric data-driven brute-force context model by aligning a hypothesis $\mathbf{y}$ with all manual 3D annotations $\{\mathbf{y}_1^*, \mathbf{y}_2^*, \cdots, \mathbf{y}_N^*\}$ in the training set. After registering two scenes, we can efficiently compute the distances between all pairs of cuboids in the two scenes. The distance is defined as a combination of center distance, volume intersection over union, and semantic type consistency. Since our training data has limited size, we apply various transformations $\mathbf{T}$ on the ground truth rooms to increase the database diversity. Specifically, we increase/decrease the size of the room, while keep the relative positions of all objects in room unchanged, or keep their absolute distance to a wall fixed. We further allow a universal scaling on the whole scene. The room level feature $\mathbf{f}^{\text{room}}$ is defined as the accumulated sums and products of the 10 smallest matching costs between a hypothesis with all rooms in training data under these transformation.

Room-only color model: To consider all the objects together, we divide image region to foreground (pixels covered by objects) and background (other pixels). In each regions, we extract the similar feature as defined in $\mathbf{f}^{\text{object}}$. This provides context information integrating both bottom-up and top-down information, and it is part of $\mathbf{f}^{\text{room}}$.

Local adjustment: The top hypothesis returned by the SVM could be limited by the size of our hypothesis pool. So we apply a local refinement to some high score hypotheses for a result with higher SVM score. Specifically, we delete, add, or swap an object using the pairwise context constraints, or completely re-sample some new rooms. If this generates a result with higher SVM score, we will accept this new one and perform another local refinement around it.

3 Experiments

3.1 Annotated 3D panorama dataset

We collected 700 full-view panoramas for home environments from SUN360 database [19], including 418 bedrooms and 282 living rooms. We split our dataset into two halves for training and testing respectively. The data is manually annotated in house by five persons. After that, an author went through each image to correct mistakes and ensure consistency among the annotation.

To annotate panorama, we designed a WebGL annotation tool in browser, which renders a 360° panorama as a texture warped inside a sphere with the camera located at the center (Fig. 8). To annotate an object, the user first chooses one of the nine predefined viewpoints of a cuboid (shown in the black box in Fig. 8), because a different viewpoint requires a different set of vertices to be specified. When the user is marking these vertices, the interface will highlight the corresponding vertex on a 3D cuboid on the right. We ask the annotator to mark the 3D bounding box as tight as possible and align it with major orientations of the object. A key novelty of our tool is to first let the user to choose one of the nine predefined viewpoints for each cuboid object, and click each visible vertices guided by the instruction. We found that this interface is much easier to use than [17], where the viewpoint is implicit. For rectangular objects, we annotate its four corners using a polygon tool. To label the room layout, we design a specialized primitive which requires the user to click on the eight corners of the room.

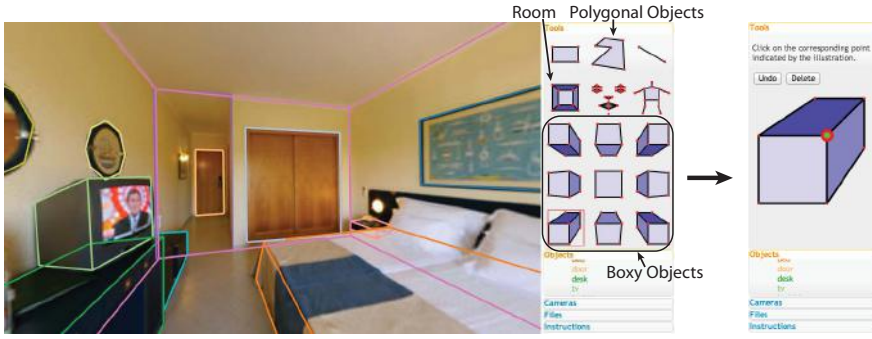


Fig. 8. Panoramic annotation tool. To annotate a 3D cuboid, the user picks a viewpoint from [Tools], and clicks on the vertices of the boxes on the panorama. The screen displays an indication about what is the next corner to click on, as shown on the right.

We further convert 2D annotations to 3D scene models. We assume each object as a perfect cuboid which only rotates horizontally around a vertical axis. Furthermore, we do not allow any floating object, such that each object can only be standing on the ground, sitting on another object, or attaching to a wall. The task of generating 3D scene models amounts to find a perfect cuboid for each object by minimizing the reprojection error between the 3D cuboid and annotations under those constraints. This can be done by a single non-linear optimization.

3.2 Evaluation

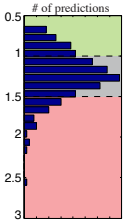
Some results for both bedroom and living room are shown in Fig. 10, where we can see that the algorithm performs reasonably.

Matching cost to the ground truth: The most straightforward way for evaluation is to compare the prediction with the ground truth, using the matching cost that we defined to choose label for the SVM training in Sec. 2.5. The average matching cost is 1.23, which is much better than the pure bottom-up hypotheses generation (average cost is 1.55). We show the histogram for the distributions of the matching cost in Fig. 9(a).

Semantic image segmentation: We also covert the 3D understanding results into a semantic segmentation mask on the panoramas and compare the results with the ground truth as in PASCAL VOC segmentation challenge [3]. During conversion, we use the 3D bounding boxes of the objects to create a mask with occlusion testing. To avoid the artifact of panorama projection, instead of comparing panorama segmentation results, we uniformly sample rays of different orientation on the sphere and compare the label of the prediction and ground truth. Fig. 9(b) shows the accuracy.

4 How important is larger FOV and context?

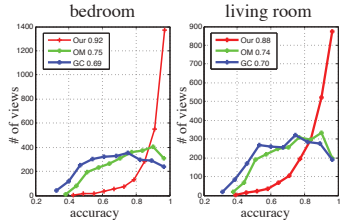
To justify the importance of FOV for context model, we conduct five types of comparison. First, we show that a larger FOV provides stronger context to recover room



(a) Cost

category	accuracy	category	accuracy
background	86.90	wardrobe	27.44
bed	78.58	tv	34.81
painting	38.70	door	19.40
nightstand	39.66	chair	9.61
window	35.58	sofa	11.10
mirror	38.15	cabinet	5.46
desk	29.55	average	35.00

(b) Semantic segmentation accuracy (bedroom)



(c) Compare with regular FOV.

Fig. 9. Evaluation. (a) shows the matching cost distribution for the top hypotheses of our results (for bedroom). (b) shows the accuracy for semantic segmentation. (c) shows the distribution of views across different surface orientation prediction accuracy (see the supp. material for more).

layout.Second, we show that whole-room context provides stronger cue for recognizing objects than an object detector which classifies an image patch (i.e. small FOV). Third, we decompose our system and disable some key usages of global context information, to see the importance of context during sampling and ranking. Fourth, we vary the effective range of FOV in the context model to demonstrate the larger FOV enables stronger context. Finally, we combine our context model with standard object detectors, to demonstrate the complementary natural of context and local object appearance.

Is larger FOV helpful for room layout estimation? We warp the panorama images into perspective images using $54.4°$ FOV, run [16] and [15] on these images to obtain surface orientation estimation for regular FOV images. Then our result and the ground truth on panorama are warped to these perspective views for comparison. From the comparison shown in Fig. 9(c), we can see that by using panorama, our algorithm significantly outperforms these results on regular FOV images.

Is context as powerful as local image appearance for object detection? Using only our top 1 prediction for each panorama images, we can compute the precision and recall for each object category. Therefore, we compare with the state-of-the-art object detector. We train DPM [2] using the SUN database [20]. To test it on panorama, we warp a panorama into many regular perspective images and run the trained DPM on it. Fig. 12(a) shows the result of the comparison. We can see that our model performs better than DPM at many object categories. This demonstrates that by using only 3D contextual information without any image feature for categorization, we can still achieve a comparable performance with object detectors using only image features.

Is context important in sampling and ranking? We disable the pairwise context model for sampling and the room alignment matching cost for ranking respectively and together to show the power of each one. In Fig. 12(a), the detection performances for nightstand and tv keep on decreasing when disabling more context model. These objects are usually in a common size and shape, and thus cannot be discriminated easily with bottom up image evidence. Context information can be especially useful under this situation. However, for painting and door, the performance does not change much. The reason could be that these objects usually have strong relations with the walls, and we did not turn off the wall-object context, so the pairwise or high level context does not matter much. Such strong context between wall and objects further shows the advantage of using panorama, in which all the walls, floor, and ceiling are visible.

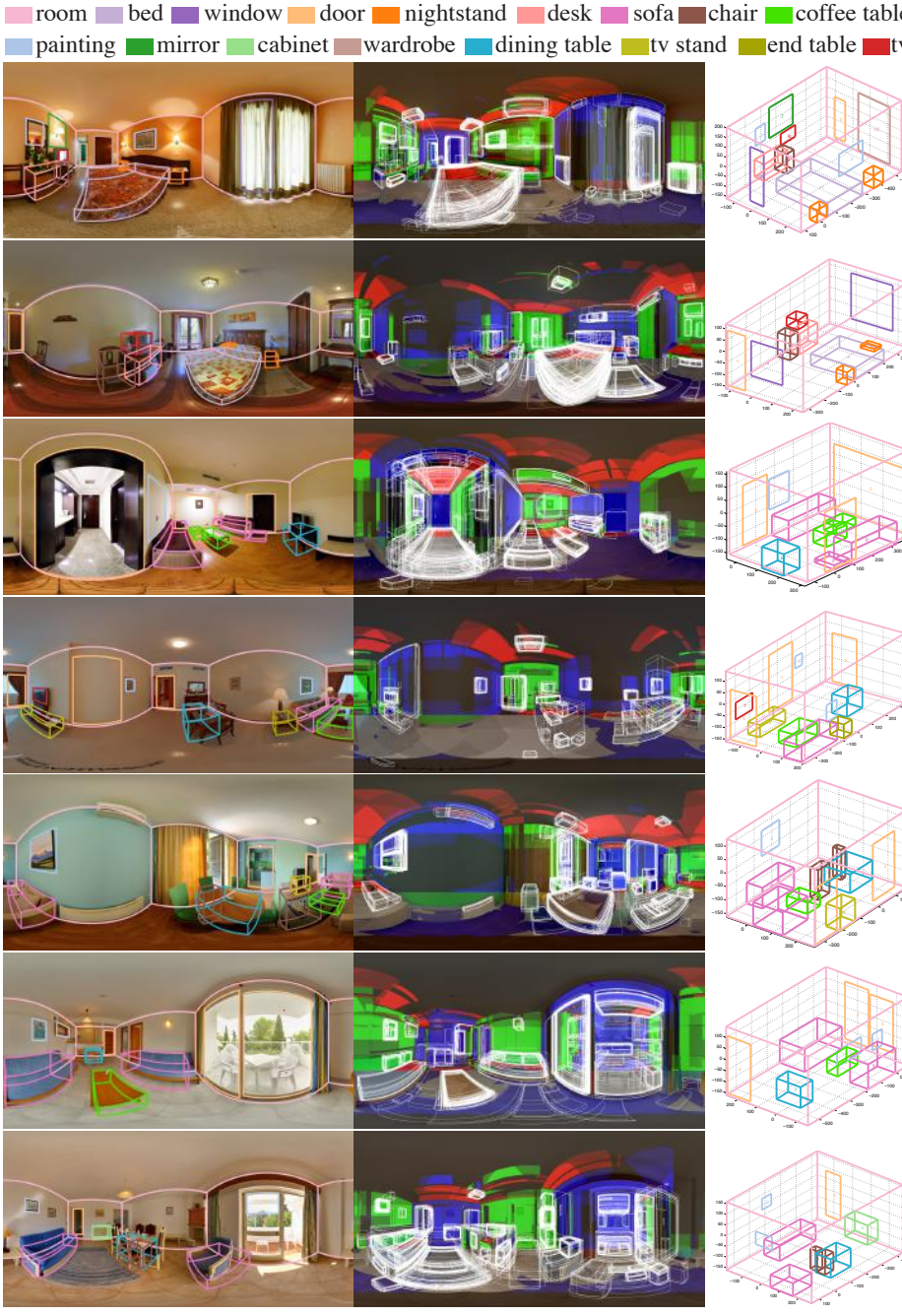


Fig. 10. Example results. The first column is the input panorama and the output object detection results. The second column contains intermediate steps for generating cuboid hypotheses from bottom-up sampling as well as the combination of OM and GC. The third column is the results visualized in 3D. (Best view in color. More results are available in the supplementary material.)

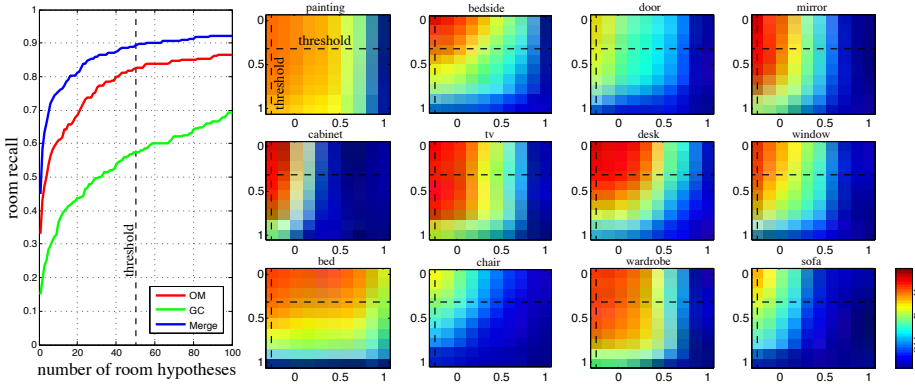


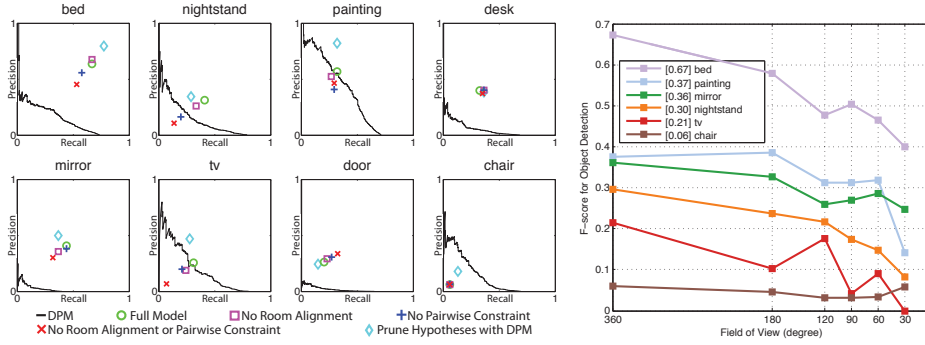
Fig. 11. Recall. Left: The room recall verse the number of top room hypotheses ranked by the OM, GC and our combination of OM and GC, which shows that merging OM and GC significantly improves the result and justifies that 50 is a good threshold. Right: The object recall w.r.t. the rectangle detection score (horizontal axes) and random forest score (vertical axes).

Is larger FOV better for context? We narrow down the FOV that is allowed to for pairwise context model. Pair of cuboids far in term of FOV cannot affect each other by pairwise context model during the whole-room hypothesis sampling. Fig. 12(b) shows the F-score ($\sqrt{\text{precision} \times \text{recall}}$) of object detection w.r.t. different FOVs. We can see that the F-score curves are all in a decreasing tendency when the FOV is getting smaller. It shows that the big FOV is essential in providing more context information.

Is context complementary with local object appearance? We combine to our model with object detector to answer this question. We run DPM on each object hypothesis, and prune those with low score during the scene hypothesis sampling. The result is shown in Fig. 12(a). We can see that for categories on which both DPM and context perform well, merging them will achieve higher performance, like bed, tv, painting. It proves that the context information is complementary to image evidence. For categories that DPM does not work well, the improvement benefit from merging is very limited as expected, like mirror, desk. For the objects without much context, e.g. chair, though the performance is improved, it is still not comparable with DPM, which probably implies that context could hurt the detection performance for objects with flexible locations. Note that this is just a simple test to show the effect of merging DPM with our system, there are actually many parts in our model which can be improved by a strong image feature based detector. The confidence score from detector can be used as a powerful bottom up feature of object hypotheses during the sampling and holistic ranking.

5 Related works

There are many exceptional works that inspired the design of our algorithm. The surface orientation and depth estimation from a single image is studied in [21–28]. The state-of-the-art of single view room layout estimation can be found in [15, 16, 29–51]. Our work extends them to full-view panorama to fully exploit the contextual information. There are also many great works that model context and object relations [52–63]



(a) Precision-recall comparison with DPM (b) F-score with different FOV

Fig. 12. Object detection. (a) the performance of our system by partially disable some key usage of context, and the comparison with DPM. (b) F-score of our system with decreasing field of view. The performance becomes worse when the FOV is getting smaller.

and parse a scene [64, 65, 30] in a unified way. Although they have some success on reasoning about 3D, their main focus is still on 2D, while our context model is fully in 3D. For scene parsing grammar, several approaches such as And-Or graph, stochastic grammar, or probabilistic languages have been proposed [66–73]. Our data-driven sampling and discriminative training provides a simple but powerful way to combine the bottom-up image evidence and top-down context information. Same with our assumptions, 3D cuboids are also a popular representation for scene understanding [15, 17, 34, 47, 48, 74–76]. For object recognition datasets, there are several main-stream datasets that contain object annotation in regular pictures [3, 19, 77, 20, 78–81]. Our panorama dataset is the first annotated panorama dataset for scene understanding, and we also provide ground truth in 3D. For using panoramas in computer vision tasks, there are several projects focus on scene viewpoint recognition, localization, image extrapolation and warping [19, 82–84]. Recently, the rapid increase of popularity of RGB-D sensors enables many seminar works on scene understanding in 3D [85–94, 74]. We expect our approach can also be naturally extended into RGB-D panoramas or RGB-D scanned 3D rooms [95].

6 Conclusion

Small field-of-view in standard cameras is one of the main reasons that contextual information is not as useful as it should be. To overcome this limitation, we propose a whole-room 3D context model that takes a 360° panorama as input and outputs a 3D bounding box of the room and detects all major objects inside. Experiments show that our model can recognize objects using only 3D contextual information, and still achieves a comparable performance with the state-of-the-art object detector using image features for categorization. We showcase that the root of context model being under-utilized is partially because regular FOVs are too small, and that context is more powerful than we thought. We believe that this is a useful message, because our community is not fully aware that we make recognition more difficult than it should be.

References

1. Roberts, L.G.: Machine perception of 3-D solids. PhD thesis, Massachusetts Institute of Technology (1963)
2. Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. PAMI (2010)
3. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The PASCAL visual object classes (voc) challenge. IJCV (2010)
4. Uijlings, J.R.R., van de Sande, K.E.A., Gevers, T., Smeulders, A.W.M.: Selective search for object recognition. IJCV (2013)
5. Wang, X., Yang, M., Zhu, S., Lin, Y.: Regionlets for generic object detection. In: ICCV. (2013)
6. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. arXiv preprint arXiv:1311.2524 (2013)
7. Biederman, I.: On the semantics of a glance at a scene. (1981)
8. Torralba, A.: Contextual influences on saliency. (2004)
9. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. IJCV (2004)
10. Brown, M., Lowe, D.G.: Recognising panoramas. In: ICCV. (2003)
11. Brown, M., Lowe, D.G.: Automatic panoramic image stitching using invariant features. IJCV (2007)
12. Fan, R.E., Chang, K.W., Hsieh, C.J., Wang, X.R., Lin, C.J.: Liblinear: A library for large linear classification. JMLR (2008)
13. von Gioi, R.G., Jakubowicz, J., Morel, J.M., Randall, G.: LSD: a Line Segment Detector. Image Processing On Line (2012)
14. Hough, P.V.: Machine analysis of bubble chamber pictures. In: International Conference on High Energy Accelerators and Instrumentation. Volume 73. (1959)
15. Hedau, V., Hoiem, D., Forsyth, D.: Recovering the spatial layout of cluttered rooms. In: ICCV. (2009)
16. Lee, D.C., Hebert, M., Kanade., T.: Geometric reasoning for single image structure recovery. In: CVPR. (2009)
17. Xiao, J., Russell, B.C., Torralba, A.: Localizing 3D cuboids in single-view images. In: NIPS. (2012)
18. Joachims, T., Finley, T., Yu, C.N.J.: Cutting-plane training of structural svms. Machine Learning (2009)
19. Xiao, J., Ehinger, K.A., Oliva, A., Torralba, A.: Recognizing scene viewpoint using panoramic place representation. In: CVPR. (2012)
20. Xiao, J., Hays, J., Ehinger, K.A., Oliva, A., Torralba, A.: SUN database: Large-scale scene recognition from abbey to zoo. In: CVPR. (2010)
21. Delage, E., Lee, H., Ng, A.Y.: Automatic single-image 3D reconstructions of indoor manhattan world scenes. In: ISRR. (2005)
22. Coughlan, J.M., Yuille, A.: Manhattan world: Compass direction from a single image by bayesian inference. In: ICCV. (1999)
23. Hoiem, D.: Seeing the world behind the image: spatial layout for 3D scene understanding. PhD thesis, Carnegie Mellon University (2007)
24. Saxena, A., Sun, M., Ng, A.: Make3D: Learning 3D scene structure from a single still image. PAMI (2009)
25. Hoiem, D., Efros, A.A., Hebert, M.: Automatic photo pop-up. In: TOG. (2005)
26. Hoiem, D., Efros, A.A., Hebert, M.: Putting objects in perspective. IJCV (2008)
27. Hoiem, D., Efros, A.A., Hebert, M.: Closing the loop in scene interpretation. In: CVPR. (2008)

28. Hoiem, D., Efros, A.A., Hebert, M.: Geometric context from a single image. In: ICCV. (2005)
29. Gupta, A., Satkin, S., Efros, A.A., Hebert, M.: From scene geometry to human workspace. In: CVPR. (2011)
30. Han, F., Zhu, S.C.: Bottom-up/top-down image parsing by attribute graph grammar. In: ICCV. (2005)
31. Zhao, Y., chun Zhu, S.: Image parsing with stochastic scene grammar. In: NIPS. (2011)
32. Wang, H., Gould, S., Koller, D.: Discriminative learning with latent variables for cluttered indoor scene understanding. In: ECCV. (2010)
33. Yu, S., Zhang, H., Malik, J.: Inferring spatial layout from a single image via depth-ordered grouping. In: IEEE Workshop on Perceptual Organization in Computer Vision. (2008)
34. Hedau, V., Hoiem, D., Forsyth, D.: Thinking inside the box: Using appearance models and context based on room geometry. In: ECCV. (2010)
35. Lee, D.C., Gupta, A., Hebert, M., Kanade, T.: Estimating spatial layout of rooms using volumetric reasoning about objects and surfaces. In: NIPS. (2010)
36. Pero, L.D., Guan, J., Brau, E., Schlecht, J., Barnard, K.: Sampling bedrooms. In: CVPR. (2011)
37. Yu, L.F., Yeung, S.K., Tang, C.K., Terzopoulos, D., Chan, T.F., Osher, S.: Make it home: automatic optimization of furniture arrangement. TOG (2011)
38. Pero, L.D., Bowdish, J.C., Fried, D., Kermgard, B.D., Hartley, E.L., Barnard, K.: Bayesian geometric modelling of indoor scenes. In: CVPR. (2012)
39. Hedau, V., Hoiem, D., Forsyth, D.: Recovering free space of indoor scenes from a single image. In: CVPR. (2012)
40. Schwing, A.G., Hazan, T., Pollefeys, M., Urtasun, R.: Efficient structured prediction for 3D indoor scene understanding. In: CVPR. (2012)
41. Xiao, J., Hays, J., Russell, B.C., Patterson, G., Ehinger, K., Torralba, A., Oliva, A.: Basic level scene understanding: Categories, attributes and structures. Frontiers in Psychology (2013)
42. Guo, R., Hoiem, D.: Beyond the line of sight: labeling the underlying surfaces. In: ECCV. (2012)
43. Satkin, S., Hebert, M.: 3DNN: Viewpoint invariant 3D geometry matching for scene understanding. In: ICCV. (2013)
44. Satkin, S., Lin, J., Hebert, M.: Data-driven scene understanding from 3D models. In: BMVC. (2012)
45. Choi, W., Chao, Y.W., Pantofaru, C., Savarese, S.: Understanding indoor scenes using 3D geometric phrases. In: CVPR. (2013)
46. Del Pero, L., Bowdish, J., Kermgard, B., Hartley, E., Barnard, K.: Understanding bayesian rooms using composite 3D object models. In: CVPR. (2013)
47. Zhao, Y., Zhu, S.C.: Scene parsing by integrating function, geometry and appearance models. In: CVPR. (2013)
48. Schwing, A.G., Fidler, S., Pollefeys, M., Urtasun, R.: Box in the box: Joint 3D layout and object reasoning from single images. (2013)
49. Schwing, A.G., Urtasun, R.: Efficient exact inference for 3D indoor scene understanding. In: ECCV. (2012)
50. Chao, Y.W., Choi, W., Pantofaru, C., Savarese, S.: Layout estimation of highly cluttered indoor scenes using geometric and semantic cues. In: ICIAP. (2013)
51. Furlan, A., Miller, D., Sorrenti, D.G., Fei-Fei, L., Savarese, S.: Free your camera: 3D indoor scene understanding from arbitrary camera motion. In: BMVC. (2013)
52. Rabinovich, A., Vedaldi, A., Galleguillos, C., Wiewiora, E., Belongie, S.: Objects in context. In: ICCV. (2007)

53. Tu, Z.: Auto-context and its application to high-level vision tasks. In: CVPR. (2008)
54. Choi, M.J., Torralba, A., Willsky, A.S.: A tree-based context model for object recognition. PAMI (2012)
55. Choi, M.J., Torralba, A., Willsky, A.S.: Context models and out-of-context objects. Pattern Recognition Letters (2012)
56. Choi, M.J., Lim, J.J., Torralba, A., Willsky, A.S.: Exploiting hierarchical context on a large database of object categories. In: CVPR. (2010)
57. Desai, C., Ramanan, D., Fowlkes, C.C.: Discriminative models for multi-class object layout. IJCV (2011)
58. Ladicky, L., Russell, C., Kohli, P., Torr, P.H.: Graph cut based inference with co-occurrence statistics. In: ECCV. (2010)
59. Sudderth, E.B., Torralba, A., Freeman, W.T., Willsky, A.S.: Describing visual scenes using transformed objects and parts. IJCV (2008)
60. Sudderth, E.B., Torralba, A., Freeman, W.T., Willsky, A.S.: Depth from familiar objects: A hierarchical model for 3D scenes. In: CVPR. (2006)
61. Sudderth, E., Torralba, A., Freeman, W., Willsky, A.: Describing visual scenes using transformed dirichlet processes. In: NIPS. (2005)
62. Sudderth, E.B., Torralba, A., Freeman, W.T., Willsky, A.S.: Learning hierarchical models of scenes, objects, and parts. In: ICCV. (2005)
63. Sudderth, E.B., Jordan, M.I.: Shared segmentation of natural scenes using dependent pitman-yor processes. In: NIPS. (2008)
64. Li, C., Kowdle, A., Saxena, A., Chen, T.: Towards holistic scene understanding: Feedback enabled cascaded classification models. PAMI (2012)
65. Heitz, G., Gould, S., Saxena, A., Koller, D.: Cascaded classification models: Combining models for holistic scene understanding. In: NIPS. (2008)
66. Wu, T., Zhu, S.C.: A numerical study of the bottom-up and top-down inference processes in and-or graphs. IJCV (2011)
67. Battaglia, P.W., Hamrick, J.B., Tenenbaum, J.B.: Simulation as an engine of physical scene understanding. Proceedings of the National Academy of Sciences (2013)
68. Tenenbaum, J.B., Kemp, C., Griffiths, T.L., Goodman, N.D.: How to grow a mind: Statistics, structure, and abstraction. Science (2011)
69. Mansinghka, V.K., Kulkarni, T.D., Perov, Y.N., Tenenbaum, J.B.: Approximate bayesian image interpretation using generative probabilistic graphics programs. In: NIPS. (2013)
70. Han, F., Zhu, S.C.: Bottom-up/top-down image parsing with attribute grammar. PAMI (2009)
71. Tu, Z., Chen, X., Yuille, A.L., Zhu, S.C.: Image parsing: Unifying segmentation, detection, and recognition. IJCV (2005)
72. Li, L.J., Socher, R., Fei-Fei, L.: Towards total scene understanding: Classification, annotation and segmentation in an automatic framework. In: CVPR. (2009)
73. Li, L.J., Su, H., Xing, E.P., Li, F.F.: Object bank: A high-level image representation for scene classification & semantic feature sparsification. In: NIPS. (2010)
74. Lin, D., Fidler, S., Urtasun, R.: Holistic scene understanding for 3D object detection with rgbd cameras. In: ICCV. (2013)
75. Fidler, S., Dickinson, S.J., Urtasun, R.: 3D object detection and viewpoint estimation with a deformable 3d cuboid model. In: NIPS. (2012)
76. Xiao, J., Furukawa, Y.: Reconstructing the world's museums. IJCV (2014)
77. Russell, B.C., Torralba, A., Murphy, K.P., Freeman, W.T.: LabelMe: a database and web-based tool for image annotation. IJCV (2008)
78. Bell, S., Upchurch, P., Snavely, N., Bala, K.: OpenSurfaces: a richly annotated catalog of surface appearance. TOG (2013)
79. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: CVPR. (2009)

80. Shotton, J., Winn, J., Rother, C., Criminisi, A.: TextonBoost for image understanding: Multi-class object recognition and segmentation by jointly modeling texture, layout, and context. IJCV (2009)
81. Russell, B.C., Torralba, A.: Building a database of 3D scenes from user annotations. In: CVPR. (2009)
82. Ni, K., Kannan, A., Criminisi, A., Winn, J.: Epitomic location recognition. In: CVPR. (2008)
83. Zhang, Y., Xiao, J., Hays, J., Tan, P.: Framebreak: Dramatic image extrapolation by guided shift-maps. In: CVPR. (2013)
84. He, K., Chang, H., Sun, J.: Rectangling panoramic images via warping. TOG (2013)
85. Song, S., Xiao, J.: Sliding Shapes for 3D object detection in RGB-D images. In: ECCV. (2014)
86. Wu, Z., Song, S., Khosla, A., Tang, X., Xiao, J.: 3D ShapeNets for 2.5D object recognition and Next-Best-View prediction. ArXiv e-prints (2014)
87. Guo, R., Hoiem, D.: Support surface prediction in indoor scenes. (2013)
88. Gupta, S., Arbelaez, P., Malik, J.: Perceptual organization and recognition of indoor scenes from rgb-d images. In: CVPR. (2013)
89. Silberman, N., Hoiem, D., Kohli, P., Fergus, R.: Indoor segmentation and support inference from rgbd images. In: ECCV. (2012)
90. Jiang, H., Xiao, J.: A linear approach to matching cuboids in RGBD images. In: CVPR. (2013)
91. Kim, B., Kohli, P., Savarese, S.: 3D scene understanding by Voxel-CRF. In: ICCV. (2013)
92. Zhang, J., Kan, C., Schwing, A.G., Urtasun, R.: Estimating the 3D layout of indoor scenes and its clutter from depth sensors. In: ICCV. (2013)
93. Jia, Z., Gallagher, A., Saxena, A., Chen, T.: 3D-based reasoning with blocks, support, and stability. In: CVPR. (2013)
94. Zheng, B., Zhao, Y., Yu, J.C., Ikeuchi, K., Zhu, S.C.: Beyond point clouds: Scene understanding by reasoning geometry and physics. In: CVPR. (2013)
95. Xiao, J., Owens, A., Torralba, A.: SUN3D: A database of big spaces reconstructed using sfm and object labels. In: ICCV. (2013)