Detailed 3D Model Driven Single View Scene Understanding

Maheen Rashid
Carnegie Mellon University
5000 Forbes Ave, Pittsburgh, PA 15213
maheenr@andrew.cmu.edu

Martial Hebert
Carnegie Mellon University
5000 Forbes Ave, Pittsburgh, PA 15213
http://www.cs.cmu.edu/~./hebert

Abstract

We present a data driven approach to holistic scene understanding. From a single image of an indoor scene, our approach estimates its detailed 3D geometry, i.e. the location of its walls and floor, and the 3D appearance of its containing objects, as well as its semantic meaning, i.e. a prediction of what objects it contains. This is made possible by using large datasets of detailed 3D models alongside appearance based detectors. We first estimate the 3D layout of a room, and extrapolate 2D object detection hypotheses to three dimensions to form bounding cuboids. Cuboids are converted to detailed 3D models of the predicted semantic category. Combinations of 3D models are used to create a large list of layout hypotheses for each image - where each layout hypothesis is semantically meaningful and geometrically plausible. The likelihood of each layout hypothesis is ranked using a learned linear model - and the hypothesis with the highest predicted likelihood is the final predicted 3D layout. Our approach is able to recover the detailed geometry of scenes, provide precise segmentation of objects in the image plane, and estimate objects' pose in 3D.

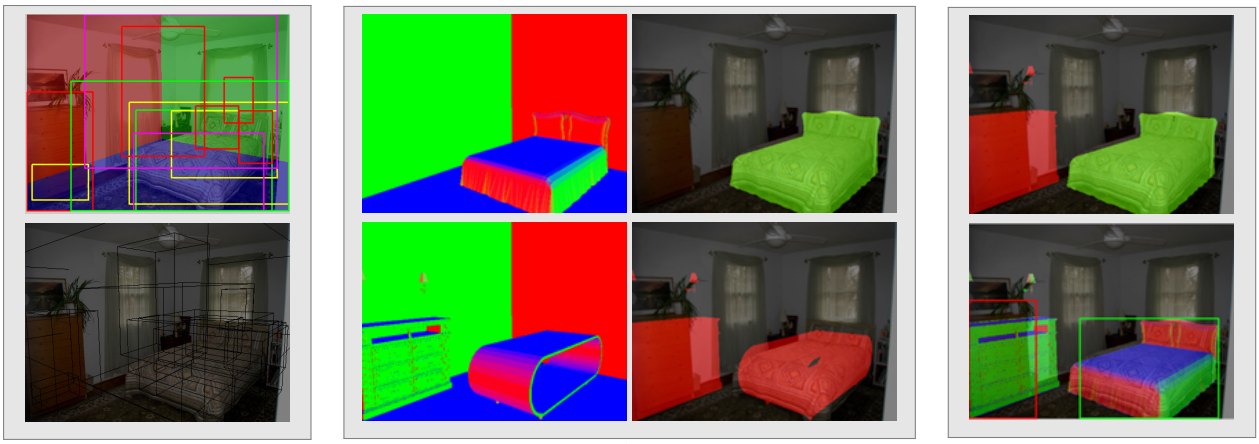
1. Introduction

The problem of single view 3D scene understanding has motivated a large amount of computer vision research. [11, 15, 27] use the Manhattan world assumption to jointly predict room layout and clutter in indoor scenes. [28] introduces a purely geometric approach that extracts 3D planes from a single image by using depth ordering, vanishing points and line direction. [14, 24, 9, 26] use depth estimation to provide predictions of scene's layout in 3D. In [21, 19, 20], Markov Chain Monte Carlo based sampling techniques are used to jointly solve for scene and objects' layout as well as camera parameters, while [25] proposes an efficient branch and bound solution to scene's 3D inference. The aim of this work is to predict the complete geometry of a scene from a single image and has applicability in the fields of both computer vision and graphics. For example, single image geometry is used by Karsch et al. [13] to realistically alter scenes' lighting and objects. In computer vision, Gupta *et al*. [10] uses an estimate of indoor scene geometry to predict how different surfaces in a scene are likely to be used by a human agent.

One theme of research has been to use 3D models to precisely model objects in images [17, 29, 4]. Lim *et al*.'s work [17] is a typical recent example which demonstrates the strength of using 3D models for the purpose of instance based object detection. By combining key-point correspondences between images and CAD models of IKEA furniture, and 2D appearance based features, they successfully perform instance detection and fine grained pose estimation of scene objects. However the information provided by a detection, or multiple detections, is not used to extract further information from the scene with respect to the room layout, or objects volume and 3D location.

At the same time a body of research has focused on improving single view scene understanding by modelling contextual relations between scene elements in 3D rather than 2D. In [5], for example, a scene classifier, 2D object detectors, and a room layout estimator are simultaneously employed to provide an estimate of the scene's geometry, and its objects' semantic labels. The system uses a dictionary of recurring object configurations in 3D to bias its predictions and hallucinate detections where appropriate. Like other similarly themed works [12, 19, 15], 3DGP uses bounding cuboids rather than precise 3D models to represent objects. Its use of 3D geometric phrases along with its use of probabilistic co-occurrence modelling makes it a non-parametric system.

These examples illustrate how both these bodies of research provide complementary information. On the one hand, Lim *et al*.'s [17] work with IKEA CAD models demonstrates the strength of using precise detailed 3D object detections. However, the wealth of information gained by performing a successful match is not used for the purpose of extracting further information about the scene's 3D geometry. On the other hand, while the use of 3D context in single view geometric scene understanding has been



(a) (b) (c)

Figure 1: From a single image we (a) estimate room layout and 2D object locations to generate bounding cuboids. Cuboids are converted to detailed 3D models from which a list of geometrically plausible layout hypothesis is created as shown in (b). The likelihood of the generated hypothesis is ranked, and the final room prediction is made as shown in (c). The recovered 3D layout is detailed and geometrically and semantically viable.

demonstrated to be useful, as in [5], it does not provide 3D information about objects beyond coarse cuboids or planar segments' based representation.

In a recent, non parametric approach to 3D scene understanding, Satkin *et al*. [22] proposes the use of 3D nearest neighbours. Unlike previously discussed approaches, Satkin *et al*. provide a tool to match entire configurations of 3D scenes to images resulting in a detailed prediction of its 3D geometry. The use of 3D models allows the generation of potentially infinite viewpoints so that image viewpoints that are unfamiliar can still be successfully matched. However, in the same vein, the 3DNN algorithm performs poorly when the 2D scene features an unfamiliar geometric layout. In addition, the algorithm exclusively uses geometric features to rank matches. This can cause geometrically similar but semantically different scenes to be matched - for example a 3D model of large table being matched to a scene containing a boxy bed.

In this paper, we propose a data-driven approach that harnesses appearance based models to provide detailed geometric and meaningful semantic 3D understanding of a scene from a single image. The contributions of this paper are two-fold.

Firstly, we demonstrate the use of detailed 3D models in a non parametric and data driven approach at both identifying as well as locating objects from a single image in 3D. We therefore demonstrate the utility of 3D models beyond the task of instance specific object detection to the larger scale problem of detecting never-before-seen objects and scene understanding in 3D. In addition we move beyond the use of uninformative bounding cuboids to the intrinsically semantically meaningful, and geometrically informative representation of 3D models.

Secondly, we further the work of one of the most recent and state of the art works in the area of scene understanding, of Satkin *et al*. [22], by creating a flexible yet robust framework for the incorporation of both semantic and appearance cues for the purpose of achieving better holistic 3D scene understanding.

2. Approach

The high level aim of our algorithm is to achieve single view 3D scene understanding that is both geometrically and semantically meaningful by combining 3D models with appearance based models.

The first step in this process is to estimate the layout of a room - that is, the 3D position of its walls and floor along with the camera parameters. Following this step we use an object detector - specifically the Deformable Part Model detector of [7], to get a 2D prior on the position and semantic category of the objects in the scene. We use the estimated room layout and camera parameters to transform the object detector's output of 2D bounding boxes to 3D cuboids.

The extrapolated 3D cuboids provide us with a prior on the locations and dimensions of objects in the image. We place 3D models of the predicted object category within each bounding cuboid. For every placement of a 3D object (which we call swapping) we calculate its geometric likelihood - that is the similarity between the image and the projected 3D model.

This geometric likelihood is evaluated using the method of [22] where various geometrically meaningful image features, such as surface normal and clutter estimates, are combined in a learned linear model to output a single number as

a geometric similarity score. In other words, the geometric similarity score is $w^T L$, where each row of L is the output of a geometrically meaningful image feature [1], and w is a learned weight vector. This geometric similarity score is used at various points in our algorithm.

At this stage in our algorithm we have attempted to evaluate and re-rank the likelihood of each object detection. However in order to provide a holistic understanding of the scene, we need to take in to account the interactions between objects. We constrain the interactions between objects to be geometrically realistic, i.e., objects are constrained to not intersect one another. Using this constraint and a set of the swapped in 3D models we create a large set of hypotheses that may provide a realistic understanding of the scene. For a final prediction of the scene we train a simple linear regressor that attempts to maximize the accuracy of the retained object detections.

In the following subsections we explain each stage of our pipeline in detail. In Section 3 we evaluate the various aspects of our algorithm.

2.1. Layout Estimation

An accurate room layout hypothesis would not only lead to an accurate recovery of the intrinsic and extrinsic camera parameters but would not violate the volumetric constraints observed in the real world. In other words, a correct room layout hypothesis would not only prevent accurate 3D detections of objects from intersecting the walls or floor of the scene but also assist in obtaining accurate object detections. This observation has been made in previous literature [11, 19, 15] and used to optimize and correct room layout predictions based on the position and volume of clutter.

Consistent with this observation, we use the method of [22] to predict the layout of the room. The approach outlined by Satkin *et al*. not only incorporates geometric features in its final prediction of the room layout, but uses detailed 3D models ot re-rank the top N room layout hypothesis of [11]. It is therefore more robust than methods that use only the top ranking hypothesis of a room layout estimator such as [8, 10, 23].

Note that while we use the layout estimation approach of [22] and [11], it is possible to use any method that gives an estimate of the room layout and camera parameters, for example [16, 21, 27].

2.2. From Bounding Boxes to Cuboids

We use 2D object detections as the starting point for creating hypotheses of objects' location, geometry and semantic category in 3D. DPM harnesses crucial object specific appearance information from large datasets and can predict objects accurately despite partial occlusion and across a wide range of viewpoint and inter category appearance variation [7]. For example, a *bed* detection is indicative of what a bed looks like in the image plane. Each bounding box and semantic label provides us with a cue of where in space the detected objects lie, and what their dimensions would be.

At the same time, by extrapolating 2D object detections to three dimensions, we are able to better evaluate the geometric likelihood of detected objects and prune away incorrect detections. For example, while it is possible for both correct and incorrect detections to overlap in 2D, it is not possible for correctly detected objects to occupy the same volume in 3D. By extrapolating 2D detections to 3D, we can use the geometric constraint of non-intersecting objects to evaluate how correct a 2D object detection is.

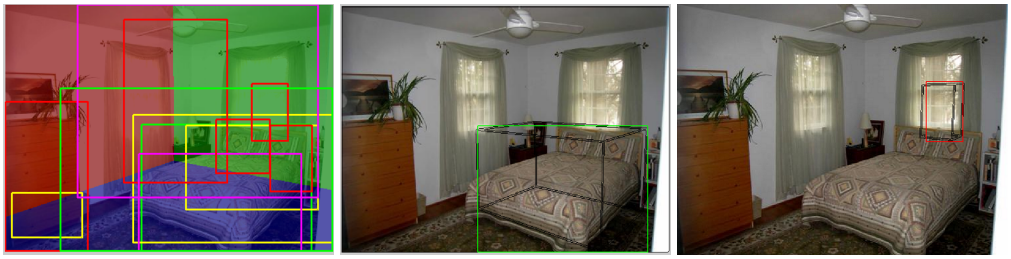
In order to utilize the appearance and semantic information each 2D detection provides us, as well as create a framework that distinguishes between correct and incorrect object detections, we transform each detection's 2D bounding box to a 3D cuboid. Even with known intrinsic and extrinsic camera parameters, this problem is inherently ill-posed. However, the semantic label and dimensions of each detection can be used alongside recovered room layout and camera parameters estimates to transform each 2D detection to a 3D cuboid. Due to its intelligent utilization of 3D models for cuboid estimation, we use the method presented in [5] to transform each 2D detection to a 3D cuboid. This method hallucinates a cuboid of probable dimensions (calculated by analysing the dimensions of IKEA models) that projects to fit in to the 2D object detector's bounding box.

While each generated cuboid respects the relative dimensions commonly seen for its semantic category in the real world, its depth from the camera has not yet been recovered. To get an estimate of its 3D dimensions that is meaningful, and to impose an absolute scale on all cuboids generated for an image, we fix the depth of each cuboid to lie on the estimated floor plane. Following, we discard cuboids that are hidden behind the walls of the 3D scene. In practice, these pruned away detections correspond to object detections on the walls and ceiling of the room. This can be seen in Figure 2, which shows examples of generated cuboids.

2.3. Object Swapping

By generating 3D cuboids we have extrapolated the semantic and appearance information each 2D object detection provides us to three dimensions. However, just as many object detections may be incorrect, many of the generated cuboids would also be incorrect. These incorrect cuboids may occupy an inappropriate volume (be too large or small for the object it is meant to bound), be positioned in an area that is free space, or be positioned in the right place with the correct volume, but predict an incorrect category. In addition, generated cuboids' location may be offset from the position of the object it is meant to bound.

[1] As in [22] we use features predicting the presence of objects [18, 12, 11], surface normals [16, 9], edges, and oriented edges [3].



(a) Object detections and estimated room layout (b) The cuboid generated for a *Bed* detection (c) Discarded *Table* detection and cuboid

Figure 2: 3D cuboids are generated from object detections' bounding boxes. Cuboids lying behind walls are discarded

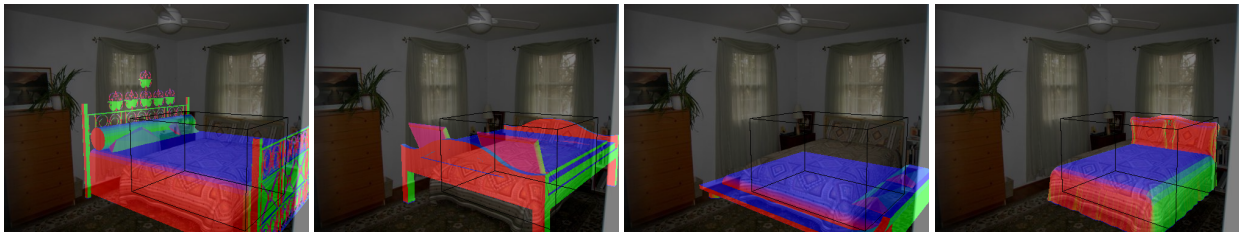


Figure 3: Four rotation transformations are applied to every swapped in model. Our dataset features many unique styles of each category.

At the same time, the generated 3D cuboid provides two cues that can be used for their further evaluation: the location of the predicted object, and the dimensions of the predicted object. However, it does not provide information about the detailed 3D appearance of the object or its orientation relative to the camera.

With these limitations in mind, the aim of the swapping algorithm is two fold: first, to evaluate the geometric and semantic likelihood of a cuboid being correct; second, to transform each generated cuboid to a detailed 3D model.

In this process we iterate over a large dataset of 3D models of the predicted category of each cuboid, placing each 3D model within the cuboid and recording its geometric likelihood score from [22]. This geometric score allows us to evaluate the similarity of the original image and the 3D model across a number of geometrically meaningful image features. For example, clutter estimates of the 2D scene are compared with a 2D clutter mask of the predicted 3D model. During the swapping process, the geometric likelihood of each object is evaluated independently of all other swapped in 3D models. In other words, the inserted model is the only object present in the 3D room.

Category	Number
Bed	288
Couch	421
Table	566
Chair	321
Side table	125

Table 1: Number of Unique 3D Models Used for Object Swapping

In order to use the information provided by the dimensions of the cuboid, we set a threshold on the overlap between the dimensions of the cuboid and the swapped in 3D model. We measure overlap by using the floor plan overlap of [22]. In this the footprint of both the generated cuboid and inserted 3D model on the visible floor are calculated separately. The pixel-wise overlap between the two is then used for evaluating overlap. We use an overlap of 40% in all our experiments.

To account for inaccuracies in the location of 3D cuboid as well as viewpoint variations, we allow for the flexibility of iterating over a number of transformations for each swap. For example, for every swap we can place the object so that the bottom left corner of the inserted objects bounding box and the cuboid are in alignment. In practice, we evaluate four transformations for every swapped in model: four 90^0 rotations placing the swapped object in the centre of the generated cuboid.

Finally, to account for inter class appearance variations we iterate over a large dataset of 3D models that features objects of various styles. This dataset is obtained using models from [2]. However, any source of 3D models can be used. Table 1 outlines the number of unique 3D models used in the swapping process for each of the 5 semantic categories, while Figure 3 shows some styles of the *bed* category in our dataset as well as an illustration of the transformations applied during the swapping process.

2.4. Hypothesis generation and final scene prediction

The Object Swapping algorithm evaluates the geometric likelihood of each swap independently of all other swaps. However, in order to provide an accurate holistic understanding of the scene in 3D it is necessary to harness the information gained in the object swapping process for the purpose of providing a prediction of object configurations that are geometrically realistic (objects do not intersect one another or the walls) and semantically correct (the 3D prediction of a image containing a bed and side table should feature these 3D objects).

In order to generate such a final prediction of the scene's configuration in 3D we generate a list of swapped object combinations with the following constraints:

- No two objects on a list should intersect one another in 3D.
- No two objects on a list should have been swapped in to the same cuboid.

The first condition ensures that the real world constraint of solid objects not intersecting one another is respected. The second constraint is imposed to respect the information gained from 2D object detection - each object detection translates to a maximum of one 3D object being associated with it. The top N object swaps with the highest geometry score per cuboid are used to generate the list of candidate object configurations.

As explained in Section 2.3 the geometric likelihood of each swapped in 3D model is evaluated with no other objects present in the room. Hence, while we have an indicator of the geometric likelihood of each object in a list, we do not yet have an indicator of the geometric likelihood of the list as a whole. We therefore insert all 3D models on a list at the position they were originally swapped in at in the 3D room, and obtain a score indicating this 3D scene's geometric similarity to the image (again using the previously discussed method of [22]). This is necessary to capture how object relations in 3D affect the similarity of the 3D scene configuration and the 2D image, for example, due to occlusions.

At this point, for each image we have generated a number of layout hypotheses or lists. Each one of these generated lists contains a combination of 3D models that are geometrically plausible but not necessarily correct. For example, in the middle panel in Figure 1 both the top and bottom row show lists with a correct 3D model(*bed* in the top row list, and *sidetable* in the bottom row list) while the list shown in bottom row also contains a 3D model that is inaccurate (the *table* positioned in place of the bed). In fact, each generated list would feature some combination of correct and incorrect 3D models and the list containing the most number of correct 3D models, and the least number of incorrect 3D models would be closest to the ground truth for that image. In other words, the best layout hypotheses for an image would be the list that is most accurate.

We therefore train a linear regressor capable of ranking the accuracy of lists for each image. For each image in the training data we calculate the accuracy of each of its lists by using the following formula

$$List\ Accuracy = \alpha_1 t + \alpha_2 f \tag{1}$$

where t is the proportion of true positive 2D object detections retained for that training image, f is the proportion of false positives discarded, and α_1, α_2 are weight terms set according to the ratio of true and false positive 2D object detections in the entire training dataset.

The features used for training this linear model - and for prediction during test time - are the geometry similarity score of the list, the semantic labels of the 3D models contained in that list, and their corresponding object detections' confidence scores.

This trained model is used to predict the accuracy of each generated list for a test image. The list with the highest predicted accuracy is taken to be our final 3D scene layout prediction.

3. Evaluation

We now evaluate our system. Firstly, we evaluate its ability to recover the poses of objects. Performance at recovery of geometry of the scene is then compared. Finally we move on to compare its performance at the task of accurate delineation of objects in the image plane. Both 3DNN [22], and 3DGP [5] are used as baselines.

All experiments are performed on the CMU 3D Annotated Scene Database [1], which contains 526 images of living rooms and bedrooms. We use the pre-trained DPM detectors from [5] for five categories: *Bed, Sofa, Chair, Side Table* and *Table*. These are trained on the PASCAL VOC dataset [6] alongside a new *furniture* dataset [5]. For experiments involving 3DGP, we report results using M1 marginalization from the paper [5], for which the best geometry recovery scores were reported.

For our own system, we use the top 1 model per cuboid for the generation of layout hypotheses as described in Section 2.4. Since the geometric features used for predicting geometric similarity from [22] are also trained on the same dataset using five-fold cross validation, we use Leave One Out training for linear regression, and restrict the training data for each model to be within the same fold. This is done to prevent data contamination. To show the effect of using more confident DPM detections as input - we vary the DPM confidence threshold and show results for two different thresholds.

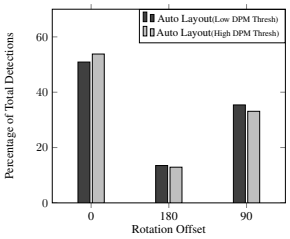


Figure 4: Bed Orientation Recovery

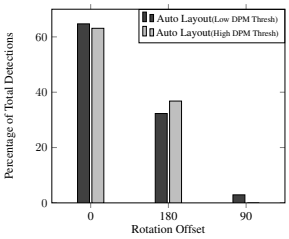


Figure 5: Sofa Orientation Recovery

3.1. Orientation

An advantage of using detailed 3D models rather than cuboids is that it allows us to recover the orientation of objects - that is, knowing which way they are facing. This allows us to recover the geometry of a 3D scene in a manner that is more meaningful especially with respect human affordance and interaction with the recovered 3D scene.

During the swapping process (Section 2.3), we evaluate each 3D model's geometric likelihood for four 90^o rotations with respect to the orthogonal walls of the room. Orientation of ground truth objects is also labelled using one of these four orientations with respect to the ground truth walls. We show the recovery of orientation for the categories of *bed* and *sofa*.

Figure 4 shows the orientations recovered as a percentage of total correct detections for the *bed* category. Note that the percentage of detections that are offset by 90^o is quite low, which indicates that even though our approach might confuse the head and foot of a bed, the overall geometric alignment of the *bed* is in general quite good. A similar trend is observed for the *sofa* category (Figure 5).

It is important to note that these results have been recovered after using only the top 1 3D model for each cuboid, and without any parametric constraints on object orientation or co-occurrences. In this sense our approach has great potential for even better recovery of orientation. For example, a simple heuristic for the recovery of object pose in 3D could be to restrict 3D models of beds to be swapped so that the head of the bed is aligned with the wall of the room. Similarly, 3D models of sofas can be restricted to face towards detections of tables.

3.2. Geometry Recovery

We evaluate the ability of our system to accurately recover the geometric layout of a scene using the metrics presented in [23, 22]. Specifically, we evaluate performance on the following metrics:

- **Surface Normals for All Pixels**: The dot product between the predicted and ground truth surface normals is evaluated at all pixels.
- **Surface Normals for Object Pixels**: The dot product between the predicted and ground truth surface normals is evaluated at only those pixels where objects exist in the ground truth.
- **Surface Normals for Matched Objects**: The dot product between the predicted and ground truth surface normals is evaluated at pixels where objects exist in both the prediction and the ground truth. For all other pixels the metric reports a 0. This is a stricter metric for evaluation since objects must be predicted in the correct location in order to score well.
- **Floor Plan Overlap**: The pixel wise overlap of the foot print of the predicted and ground truth objects on the visible floor is used.

Figure 6 shows that our system is able to out perform 3DGP across all these metrics. These results indicate that the use of detailed models greatly assists in the accurate recovery of the 3D geometry of the room.

3.3. Object Detection in 2D

The use of 3D models for scene understanding has advantages that extend beyond accurate recovery of scene layout in 3D. In particular, precise segmentation masks for detected objects can be recovered from projection of the predicted 3D models.

Figure 7 shows how modeling 3D objects as detailed models rather than cuboids is able to provide precise segmentation masks for objects in the image plane. The pixel-wise overlap detection threshold is steadily increased, and at each threshold the percentage of detections made is recorded. While 3DGP which uses bounding cuboids is not able to sustain detections across higher thresholds, both 3DNN and our own approach that use 3D models are able to sustain detections even at thresholds as high as 80 % overlap.

At the same time, by using the location, size and semantic label of 2D object detections' bounding boxes, our system is able to detect more objects than 3DNN. This difference is particularly stark for the *Sofa* category where our system is able to detect 8% more couches.

3.4. Qualitative Results

Figure 8 shows some qualitative results of our system. Our approach is able to recover the precise geometry of objects in 3D, and is also semantically informative. In addition, the last row shows examples where despite an incorrect room layout estimate, our system is able to use cues from the image plane to precisely predict objects location, orientation, and style in 3D.

In the supplementary material we show qualitative comparisons of our results against [22] and [5], as well as some failure examples of our system.

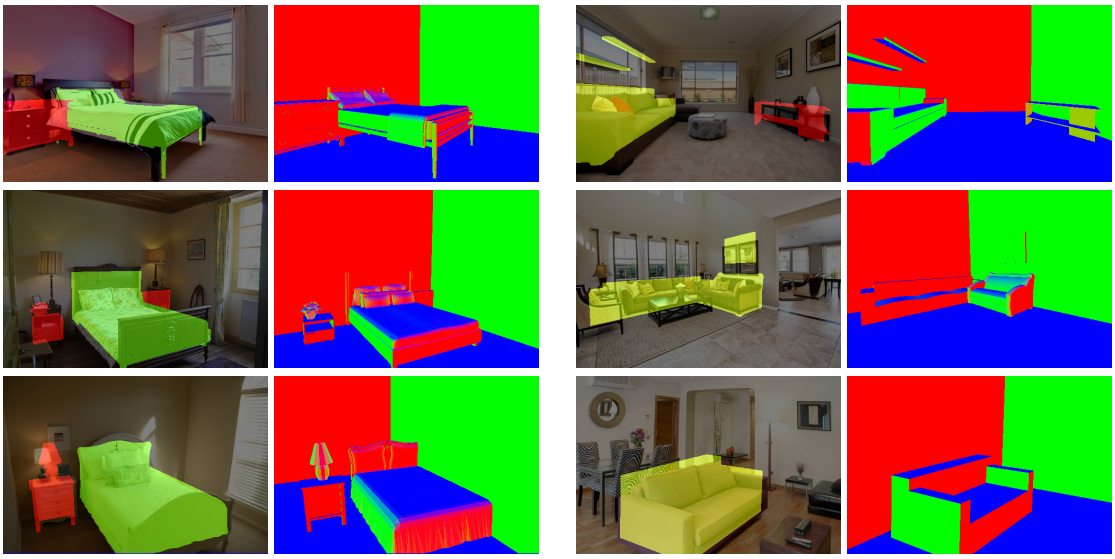


Figure 8: Our system is able to recover the precise pose, geometry and semantic meaning for scenes

4. Conclusion

While the problem of single view scene understanding is ill-posed, solving it can help answer virtually any question about 3D world. It can allow us to recover free space and the geometric and semantic layout of a scene.

In this paper we propose an approach to single view scene understanding that utilizes 3D models in a data driven framework. It successfully utilizes appearance and semantic information, alongside geometrically meaningful features and constraints, to provide a holistic understanding of a scene in 3D. The recovered 3D scene respects real world properties of objects - both geometric (objects may not intersect one another) and scalar (all objects must lie on the floor plane). The recovered scene is also detailed and semantically meaningful. For a detected sofa our system can answer questions about not only its pose, location, and dimensions in 3D - but also its style.

We have demonstrated the strength of utilizing 3D models beyond the scope of instance detection and in the realm of scene understanding; we show that 3D model representation leads to better geometry recovery, object detection, and precise image segmentation than the state of the art in scene understanding that utilizes cuboid representation [5]. We have further shown that the use of appearance and semantic priors allows our system to perform significantly better at precise object detection than methods that do not (3.3). Furthermore, the system is simple and flexible. It can be used as part of another system, or built on to achieve better and more robust scene understanding. It is therefore of value to the computer vision research community.

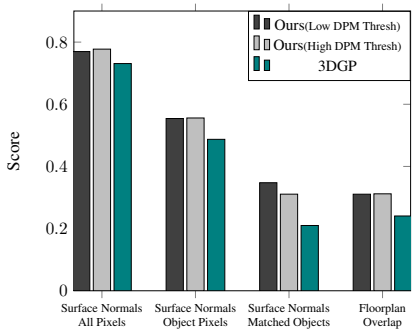


Figure 6: Averaged Geometry Recovery

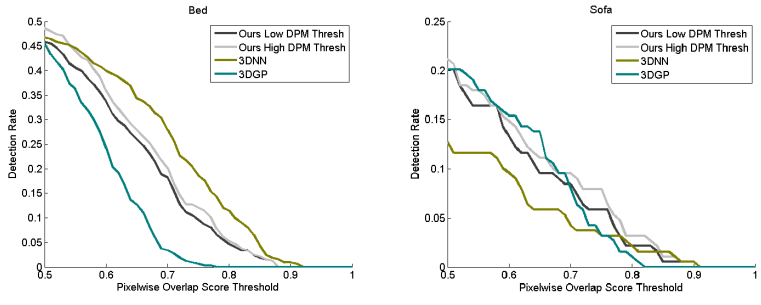


Figure 7: Pixelwise overlap detection. By using 3D models instead of cuboids, our system is able to sustain detections over higher thresholds.

References

[1] CMU 3D-Annotated Scene Database. http://cmu.satkin.com/bmvc2012/. 5

[2] Google 3D Warehouse. http://sketchup.google.com/3dwarehouse. 4

[3] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik. Contour detection and hierarchical image segmentation. *PAMI*, 33(5):898–916, May 2011. 3

[4] M. Aubry, D. Maturana, A. Efros, B. Russell, and J. Sivic. Seeing 3d chairs: exemplar part-based 2d-3d alignment using a large dataset of cad models. In *CVPR*, 2014. 1

[5] W. Choi, C. Pantofaru, and S. Savarese. Understanding indoor scenes using 3d geometric phrases. In *CVPR*, 2013. 1, 2, 3, 5, 6, 7

[6] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html. 5

[7] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1627–1645, 2010. 2, 3

[8] D. F. Fouhey, V. Delaitre, A. Gupta, A. A. Efros, I. Laptev, and J. Sivic. People watching: Human actions as a cue for single-view geometry. In *ECCV*, 2012. 3

[9] D. F. Fouhey, A. Gupta, and M. Hebert. Data-driven 3D primitives for single image understanding. In *ICCV*, 2013. 1, 3

[10] A. Gupta, S. Satkin, A. A. Efros, and M. Hebert. From 3d scene geometry to human workspace. In *CVPR*, 2011. 1, 3

[11] V. Hedau, D. Hoiem, and D. Forsyth. Recovering the spatial layout of cluttered rooms. In *ICCV*, 2009. 1, 3

[12] V. Hedau, D. Hoiem, and D. Forsyth. Thinking inside the box: Using appearance models and context based on room geometry. In *ECCV*, 2010. 1, 3

[13] K. Karsch, V. Hedau, D. Forsyth, and D. Hoiem. Rendering synthetic objects into legacy photographs. In *SIGGRAPH Asia*, 2011. 1

[14] K. Karsch, C. Liu, and S. B. Kang. Depth extraction from video using non-parametric sampling. In *ECCV*, 2012. 1

[15] D. Lee, A. Gupta, M. Hebert, and T. Kanade. Estimating spatial layout of rooms using volumetric reasoning about objects and surfaces. In *NIPS*, 2010. 1, 3

[16] D. Lee, M. Hebert, and T. Kanade. Geometric reasoning for single image structure recovery. In *CVPR*, 2009. 3

[17] J. J. Lim, H. Pirsiavash, and A. Torralba. Parsing ikea objects: Fine pose estimation. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2013. 1

[18] D. Munoz, J. A. Bagnell, and M. Hebert. Stacked hierarchical labeling. In *ECCV*, 2010. 3

[19] L. D. Pero, J. Bowdish, D. Fried, B. Kermgard, E. Hartley, and K. Barnard. Bayesian geometric modeling of indoor scenes. In *CVPR*, 2012. 1, 3

[20] L. D. Pero, J. Bowdish, E. Hartley, B. Kermgard, and K. Barnard. Understanding bayesian rooms using composite 3d object models. In *CVPR*, 2013. 1

[21] L. D. Pero, J. Guan, E. Brau, J. Schlecht, and K. Barnard. Sampling bedrooms. In *CVPR*, 2011. 1, 3

[22] S. Satkin and M. Hebert. 3dnn: Viewpoint invariant 3d geometry matching for scene understanding. In *Proceedings of the International Conference on Computer Vision (ICCV)*, 2013. 2, 3, 4, 5, 6

[23] S. Satkin, J. Lin, and M. Hebert. Data-driven scene understanding from 3D models. In *BMVC*, 2012. 3, 6

[24] A. Saxena, M. Sun, and A. Y. Ng. Make3D: Learning 3D scene structure from a single still image. *PAMI*, 2009. 1

[25] A. G. Schwing and R. Urtasun. Efficient Exact Inference for 3D Indoor Scene Understanding. In *ECCV*, 2012. 1

[26] A. Shrivastava and A. Gupta. Building part-based object detectors via 3d geometry. In *International Conference on Computer Vision*, 2013. 1

[27] H. Wang, S. Gould, and D. Koller. Discriminative learning with latent variables for cluttered indoor scene understanding. In *ECCV*, 2010. 1, 3

[28] S. Yu, H. Zhang, and J. Malik. Inferring spatial layout from a single image via depth-ordered grouping. In *CVPR Workshop*, 2008. 1

[29] M. Zia, M. Stark, B. Schiele, and K. Schindler. Detailed 3d representations for object recognition and modeling. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(11):2608–2623, Nov 2013. 1