

Amodal Detection of 3D Objects: Inferring 3D Bounding Boxes from 2D Ones in RGB-Depth Images

Zhuo Deng Longin Jan Latecki
Temple University, Philadelphia, USA

dengzhuobhu@gmail.com, latecki@temple.edu

Abstract

This paper addresses the problem of amodal perception of 3D object detection. The task is to not only find object localizations in the 3D world, but also estimate their physical sizes and poses, even if only parts of them are visible in the RGB-D image. Recent approaches have attempted to harness point cloud from depth channel to exploit 3D features directly in the 3D space and demonstrated the superiority over traditional 2.5D representation approaches. We revisit the amodal 3D detection problem by sticking to the 2.5D representation framework, and directly relate 2.5D visual appearance to 3D objects. We propose a novel 3D object detection system that simultaneously predicts objects' 3D locations, physical sizes, and orientations in indoor scenes. Experiments on the NYUV2 dataset show our algorithm significantly outperforms the state-of-the-art and indicates 2.5D representation is capable of encoding features for 3D amodal object detection. All source code and data is on <https://github.com/phoenixnn/Amodal3Det>.

1. Introduction

Object detection is one of the fundamental challenges in computer vision, the task of which is to detect the localizations of all object instances from known classes such as chair, sofa, etc in images. Traditionally, detected object instances are represented by 2D bounding boxes around visible counterparts on images. Although 2D rectangles can roughly indicate where objects are placed on image planes, their true locations and poses in the physical 3D world are difficult to determine due to multiple factors such as occlusions and the uncertainty arising from perspective projections. However, it is very natural for human beings to understand how far objects are from viewers, object poses and their full extents from still images. These kind of features are extremely desirable for many applications such as robotics navigation, grasp estimation, and Augmented Reality (AR) etc. In order to fill the gap, a variety of efforts were made in the past decade including inferring 3D object

localizations from monocular imagery [6, 13, 20, 3], and 3D object recognitions on CAD models [29, 27]. But these works either rely on a huge number of ideal 3D graphics models by assuming the locations are known or are inclined to fail in cluttered environments where occlusions are very common while depth orders are uncertain.

The recent advent of Microsoft Kinect and similar sensors alleviated some of these challenges, and thus enabled an exciting new direction of approaches to 3D object detection [18, 17, 12, 11, 19, 25, 24]. Equipped with an active infrared structured light sensor, Kinect is able to provide much more accurate depth locations of objects associated with their visual appearances. The RGB-Depth detection approaches can be roughly categorized into two groups according to the way to formulate feature representations from RGB-Depth images.

In general, 2.5D approaches start by exploiting proper feature representations on color and depth images for object detection and building models to convert 2D results to 3D space. While 3D approaches start by putting detection proposals directly in 3D space for extracting features from 3D point cloud within 3D windows. The competition to determine whether 2.5D or 3D approaches represent the right direction for 3D amodal object detection is super intense: [25] utilized 3D sliding window to directly infer detections in 3D space and demonstrate its merits for dealing with occlusions, viewpoints etc over 2.5D approaches. Then 2.5D approach [11] outperformed [25] by starting with well established 2D reasoning and aligning CAD models with 2D detections. The most recent work [24] outperformed [11] by a significant margin by introducing a 3D ConvNet model to encode 3D geometric features directly. So far, 3D centric sliding shapes leads the 3D detection performance on the challenging NYUV2 RGB-Depth dataset [22].

Although utilizing 3D geometric features for detection is promising, in practice the reconstructed 3D shapes are often incomplete (when projecting pixels of one single depth map back to 3D space), noisy and sparse (due to occlusions, reflections and absorptions of infrared lights). Hence, the quality of obtained surfaces is very different from that of

CAD models with 360° panoramas, which makes fitting 3D bounding boxes to 3D points a very challenging task. In particular, when the majority of an object area on the depth map is in a "black hole", the recovered 3D shape hardly delivers salient features. However, light signals recorded in the 2D image plane are dense and structured, and humans still can perceive the objects and estimate their 3D locations from such images. Therefore, it should be possible to mimic the human 3D perception and leverage 2.5D image features directly using current deep learning techniques. As the proposed approach demonstrates this is indeed the case.

In this paper, we revisit the 3D amodal object detection problem from the 2D point of view. We start with 2D bounding box proposals obtained from extended multiscale combinatorial grouping (MCG) class independent object proposals [1, 12]. We design a novel 3D detection neural network based on Fast-RCNN framework that naturally integrates depth information with the corresponding visual appearances to identify object classes, orientations and their full extents simultaneously in indoor scenes, where 2D bounding boxes around superpixels together with RGB-Depth images are taken as inputs. To sum up, the highlights of the main contributions of this work are as follows:

- To the best of our knowledge, we are the first to reformulate the 3D amodal detection problem as regressing class-wise 3D bounding box models based on 2.5D image appearance features only.
- Given color, depth images and 2D segmentation proposals, we designed a novel 3D detection neural network that predicts 3D object locations, dimensions, and orientations simultaneously without extra step of training SVMs on deep features or fitting 3D CAD models to 2D detections.
- We do not make any Manhattan world assumption like 3D detectors do [25, 24] for orientation estimation, since objects in rooms are often cluttered and in disorder, reflecting various lifestyles and such assumptions may have dangerous consequences for autonomous systems like mobile robots.
- In addition, in order to benefit the future amodal 3D detection research, we improved the 3D ground-truth bounding boxes for the NYUV2 dataset by fixing many errors such as wrong labeling, partial extents, false negatives etc.

2. Related Works

Object detection is one of the oldest and most fundamental problems in computer vision. A huge number of works [28, 4, 5, 10] were proposed in the past few decades

to infer bounding boxes around visible object parts within image planes. As human beings can effortlessly infer objects as a whole and complete, [16] took one step further towards obtaining similar levels of perception ability by addressing the full extent object inference problem on 2D image planes. Although this kind of object representation is richer than traditional modal inferences, it is still far from human perception level in the physical world and from the requirements for some robotic applications where robots are expected to interact with the environments. In order to fill the gap, an increased number of 3D object detection related research has been proposed, especially after active sensors become available in the consumer market. In the following, we briefly review the 3D detection algorithms for RGB-D images.

2.5D approaches in RGB-D images: 2.5D approaches generally refer to methods where depth images are treated in a similar fashion as color images in traditional 2D detection task. [17] adapted the DPM algorithm to RGB-D images by utilizing the 3D euclidean distances from depth map. Handcrafted features were extracted from both color images within the output bounding boxes of existing 2D detectors and projected 3D point clouds within their associated foreground object segmentation masks. Their object locations are parametrized using 3D ellipsoids. [18] firstly generated 3D candidate cuboids by adapting the CPMC algorithm, and then incorporated 2D appearance features, object-object and object-scene context relationships into a Conditional Random Field (CRF) model for semantic labels inference.

Recently, feature engineering has been gradually replaced by deep Convolutional Neural Networks (CNNs) in 2D image based object detection. The most popular representative works are R-CNN [10], Fast-RCNN [8] and Faster-RCNN [9]. Inspired by [25], [19] adopt an exhaustive 3D sliding window strategy in the 3D point cloud where cross-modality features were extracted by feeding projected 2d bounding boxes to pretrained R-CNNs and the following bimodal deep Boltzman Machine trained separately. Detections were then determined by an ensemble of trained exemplar SVMs. Different from the previous sliding window framework, [12] built their detectors on the top of precomputed object segmentation candidates. They extended the R-CNN [10] to utilize depth information with HHA encoding (Horizontal Disparity, Height above ground and Angle of local surface normal w.r.t gravity direction). However, the outputs of their system were still limited to 2D bounding boxes. [11] extended [12] by firstly estimating 3D coarse poses for each detected 2D object and then aligning 3D CAD models to 3D points projected back from depth image that belongs to segmentation mask with Iterative Closest Point (ICP) algorithm.

The difference of the proposed method from the previous

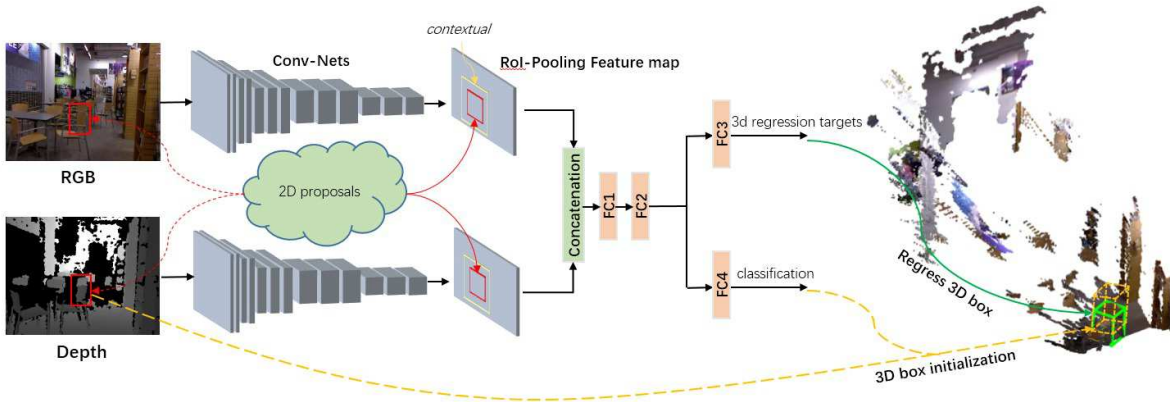


Figure 1. Overview of the proposed 3D object detection system. For each 2D segment proposal, we firstly initialize the localization of a 3D box (yellow dash box) based on depth information and its size according to classwise prior knowledge. Then object class and 3D regression offsets are jointly learned based on 2D features only, with the goal of obtaining the final 3D detection (green solid box) by adjusting the location, dimension, and orientation of the initial 3D box.

works above in three-folds: 1) no extra training examples or 3D CAD models are leveraged. 2) the model is trained end-to-end instead of piecewise. 3) no need for fitting point clouds lifted from depth map, which is often noisy and incomplete due to occlusions.

3D approaches in RGB-D images: 3D approaches make use of depth map in a different way in that 3D points are reconstructed first, and the main processing is based on analyzing point clouds. [25] extended the traditional 2D sliding window strategy to 3D by putting 3D boxes within an estimated room box. A bunch of exemplar SVMs were trained with synthetic depths rendered from 3D CAD models, and then applied to each 3D detection window in a 3D indoor scene. 3D handcrafted features were built directly on discretized 3D space. Although the approach showed encouraging detection performance, the required computations are extremely expensive. [24] improved [25] dramatically with respect to both performance and efficiency by proposing 3D region candidates and extracting 3D features directly from 3D convolutional neural networks. Similar to [25], [21] designed handcrafted 3D features on point clouds for both 3D cuboid detection and Manhattan room layout prediction. In favor of better 3D features analysis, both [24] and [21] utilized enhanced depth map derived by fusing multiple nearby depth map frames to denoise and fill in missing depth. In contrast, our method naturally models the relationships between 2.5D features and 3D object localizations and full-extents in single frame RGB-D data.

3. 3D Object Detection in RGB-D Images

3.1. Amodal 3D Object Detection

Given a pair of color and depth images, the goal of the amodal 3D object detection is to identify the object

instance locations and its full extent in 3D space. As is well-known typical indoor environments in real life are very complicated, because objects may be heavily occluded and appear in a wide range of configurations. Encouraged by the success of 3D CAD model retrieval, the available depth map makes encoding 3D geometry features directly for detection very promising. However, the quality of depth map is far from perfect in reality due to measurement errors, and more importantly, the geometry of object instances is incomplete and its variations are determined by the camera view, e.g., see examples shown in Fig. 4. This may seriously limit the representation power from direct 3D reconstruction. Therefore, in this section we revisit the task of RGB-D amodal object detection and stick to the 2.5D representation by making the assumption that underlying relationships between 2.5D feature representations and 3D object locations and orientations exist. In the following, we explore how to effectively utilize RGB and depth for this task.

2D RoI proposals: Information contained in color and depth images are demonstrated to be complimentary to each other by varieties of RGB-D research works. While color encodes distinctive visual appearance features, depth conveys the geometric structures of objects. However, in 3D detection, one additional dimension significantly enlarges the search space. Since starting with well established 2D reasoning is arguably more efficient and accurate than starting from 3D reasoning [11]. We obtain the ROI proposals by using the adapted MCG algorithm in RGB-D images [12].

3D box proposal and regression:

Lifting 2D inferred object proposals to 3D bounding boxes by fitting a tight box around the 3D points projected from pixels in the instance segmentation [18, 11, 24] is not

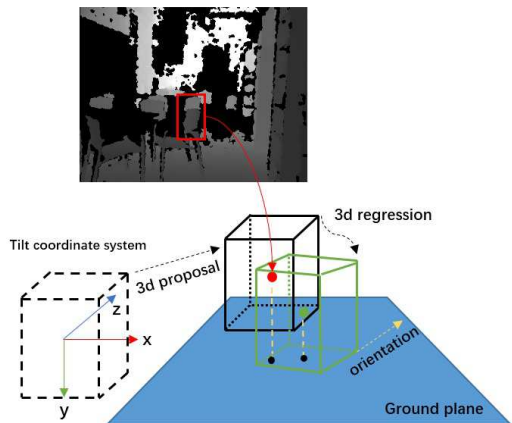


Figure 2. An example for the process of 3D box proposal and regression. The 3D box in dash line represents box initialized with class-wise averaged dimension in tilt coordinate system. The black solid line 3D box is translated from the dash-line box based on 2D segment. Finally, we regress the 3D box based on the features of the 2D segment to obtain the green 3D box. The yellow vector determines the orientation angle of 3D box to the principal axis (z-axis) in $\theta \in [-\pi/2, \pi/2]$, e.g., $\theta = 0$ if the yellow vector aligns with z-axis.

robust for 3D object detection due to both imperfect segmentations and noisy depth data. On the other hand, significantly extended 3D search space makes it inefficient to explore solutions in a brutal-force way [25]. One of the main contributions of this paper is initializing 3D proposals from 2D segments and reformulating the 3D amodal detection problem as regressing class-wise 3D bounding box models based on 2.5D visual appearance features only. As is shown in Figure 2, for each 2D segment proposal, we compute one 3D box counterpart as the 3D proposal. Then 3D proposals are transforming towards 3D ground truth according to learned high level 2.5D representations.

In this paper, the 3D bounding box is parametrized into one seven-entry vector $[x_{cam}, y_{cam}, z_{cam}, l, w, h, \theta]$. $[x_{cam}, y_{cam}, z_{cam}]$ is its centroid under camera coordinate system. $[l, w, h]$ represents its 3D size, and θ is the angle between principal axis and its orientation vector under tilt coordinate system (see Figure 2). The tilt coordinate system is converted from original camera coordinate system by aligning point clouds with gravity direction without any rotation around the y-axis:

$$XYZ_{cam} = R_{tilt}^{-1} * XYZ_{tilt} \quad (1)$$

$$R_{tilt} = R_x * R_z, \quad (2)$$

where R_{tilt} is the transform matrix between tilt and camera system, and R_x and R_z are rotation matrices around x-axis and z-axis, respectively.

3D box proposals are derived from corresponding 2D segment proposals. For box size in 3D proposals, we simply use averaged class-wise box dimensions estimated from

training set as base 3D box size. It is better than fitting 3D points projected back from 2D segment pixels, which would significantly increase variance of box dimensions for regression. It is inspired by the cues of *familiar size* in human 3D perception [7, 16]. For example, when people are looking for an object like a bed, they have a rough object dimensions in their mind, which constrains the detection of bed instances. The center of proposed 3D box $[x_{ini}, y_{ini}, z_{ini}]$ is initialized based on 3D points projected from 2D segment pixels. Since depth maps are usually noisy and have missing data, we set z_{ini} to z_{med} which is the median depth value of segment points for the sake of robustness. In the case that the whole segment is a "black hole", we use interpolated depth map instead. x_{ini} and y_{ini} are computed as described in Eq. (3): f is focal length of RGB camera, (o_x, o_y) is the principal point, (c_x, c_y) is the center of 2D box proposal.

$$\begin{cases} x_{ini} = z_{med} * (c_x - o_x) / f \\ y_{ini} = z_{med} * (c_y - o_y) / f \end{cases} \quad (3)$$

In contrast to [24], we do not make any Manhattan world assumption, since objects in rooms may appear in diverse orientations. In this work, the orientation angle θ is explicitly introduced as a parameter of 3D bounding box model. We define the orientation vector of a 3D box as the vector perpendicular to its longer edge in xz -plane (the yellow vector in Fig. 2). The initial orientation angle θ_{ini} is set to zero for all 3D box proposals, i.e., parallel to the x-axis in the tilt coordinate system, which is the case when box orientation vector aligns with camera principal axis. The range of θ is $[-\pi/2, \pi/2]$.

The 3D box regressor net will reshape the proposed raw 3D shape model based on the learned 2.5D appearance features. We represent the regression offsets as a 7-element vector $[\delta_x, \delta_y, \delta_z, \delta_l, \delta_w, \delta_h, \delta_\theta]$ for each positive example and ground truth boxes during training stage. Instead of finding the closest matching of major directions between detected box and ground-truth boxes [24] for computing box dimension differences, we can directly compare corresponding length, width and height parameters and normalize them by the size of the detected box, which is possible due to our parameterization of 3D bounding boxes. Similar to [8], the target for learning is then normalized by statistical information from proposed boxes.

Multi-task Loss: Each training example is associated with a ground-truth class c and corresponding ground-truth 3D bounding box. To jointly train for classification and bounding box regression, the loss function is defined as follows:

$$L(p, c, t_{3d}^c, v_{3d}) = L_{cls}(p, c) + \lambda(c \geq 1)L_{3d}(t_{3d}^c, v_{3d}), \quad (4)$$

where t_{3d}^c expresses the regression offsets w.r.t ground truth locations, v_{3d} are regression targets, p is the predicted probability of the object class, L_{cls} is defined as softmax function, and L_{3d} is L1 smooth loss as defined in [8].

Post processing: We apply typical Non-Maximum Suppression (NMS) scheme to the 2D detected boxes. No NMS is used in 3D. In contrast to [24], we do not perform any further pruning of the results, e.g., based on object size statistics.

3.2. Convolutional Network Architecture

There have been many deep convolutional network models proposed recently for 2D image based recognition. In this paper, we adopt the Fast-RCNN [8] as the raw base model due to both of its one single stage training architecture and high efficiency by sharing features computation. As is shown in Figure 1, color and depth images go through two VGG16 [23] Conv-Nets for computing shared feature maps, respectively. Features extracted from RoI pooling layer based on 2D object proposals and their enlarged contextual patches are concatenated for multiple tasks learning.

Mini-batch sampling



Figure 3. Red: two examples of 2D ground truth bounding boxes from [26]. Green: 2D RoI proposals. If compared 2D RoI proposals directly to red bounding boxes, the two positive chair examples are wrongly treated as negative ones. To solve this problem, we added yellow (dashed) $gt2d_{sel}$ boxes for mini-batch sampling.

For training deep neural network models, a small set of examples is randomly selected from training set to update model parameters at each iteration for the sake of computation efficiency. It is very important to properly define and select positive and negative examples from RoI pool for image based object detections. Typically, one RoI is treated as positive if it has intersection over union (IoU) overlap with ground truth box greater than 0.5, and negative if IoU is between 0.1 and 0.5. However, directly applying this rule to mini-batch sampling using 2D annotations provided by [26] would cause a serious problem. [26] provides two kinds of 2D ground truth bounding boxes for NYUV2 dataset: 1) projected 2D bounding boxes by fitting visible point clouds, and 2) projected 2D bounding boxes from amodal 3D bounding boxes. Using either kind for mini-batch sampling with 2D representations, the detection performance degrades dramatically since the true positive segments may be treated as negative ones if comparing them directly to the 2D ground truth provided by [26], as is illustrated in Fig. 3.

To fix the problem, we added new 2D ground truth box named $gt2d_{sel}$ to the training set for determining positive and negative examples from proposed 2d segments only.

We stress that the amodal 2D bounding boxes provided by [26] can be still used as targets for the 2D box regression task.

Each mini-batch consists of 256 randomly selected 2D box proposals from $N = 2$ images (128 RoIs per image). The ratio of positive and negative examples is set to 1 : 3.

For data augmentation, we flip horizontally images and their corresponding 3D bounding boxes. No other extra data is used during training.

4. Improved 3D annotations on NYUV2

The labeled NYU Depth V2 dataset [22] is a most popular but very challenging dataset in the RGBD scene understanding research community. The original version provides 1449 RGB-Depth indoor scene images with dense 2D pixelwise class labels. To enrich the labeling features and encourage 3D object detection research, in the SUN RGBD dataset [26] (superset of NYUV2) Xiao et al. added extra 3D bounding boxes and room layouts to ground truth annotations. Since depth maps are imperfect in reality due to measurement noise, light reflection and absorption, and occlusion etc, they also refined the quality of depth maps by integrating multiple RGB-D frames from the NYUV2 raw video data.

However, the extended 3D annotations in [26] have some notable issues: 1) 3D boxes were labeled independently from the original 2D object instances in NYUV2. This inconsistency leads to many salient objects being unlabeled or mislabeled, which causes unnecessary false negatives during the detection task. 2) 3D amodal annotations are mixed with modal ones. Amodal bounding boxes cover the full extent of objects, while modal ones only encompass the visible parts (e.g., see the beds in Figure 4). This is a very undesirable feature for the "amodal" detection as perused in this paper following the approaches in [2, 16, 24]. 3) Inconsistent labelings among scenes that have overlapping areas. 4) Inaccurate 3D extents or locations of object instances.

In order to provide better 3D labelings for amodal 3D object detection research, we provide improved ground truth 3D bounding boxes annotations for 19 indoor object classes (bathtub, bed, bookshelf, box, chair, counter, desk, door, dresser, garbage bin, lamp, monitor, nightstand, pillow, sink, sofa, table, tv, toilet) by complying with the following stricter principles: 1) Amodal for all: all the 3D bounding boxes should encompass the whole 3D instance of the object, even if only object parts are visible. 2) Place tight boxes around 3D object extents with reasonable orientations. 3) Comply with the physical configuration rules. For example, table and chair rest on the floor, and the height of door should not be too short. 4) Labeling is as consistent as possible with the NYUV2 2D object instances.

In the improved annotation set, we provide 3D amodal bounding boxes, 2D amodal bounding boxes cropped by

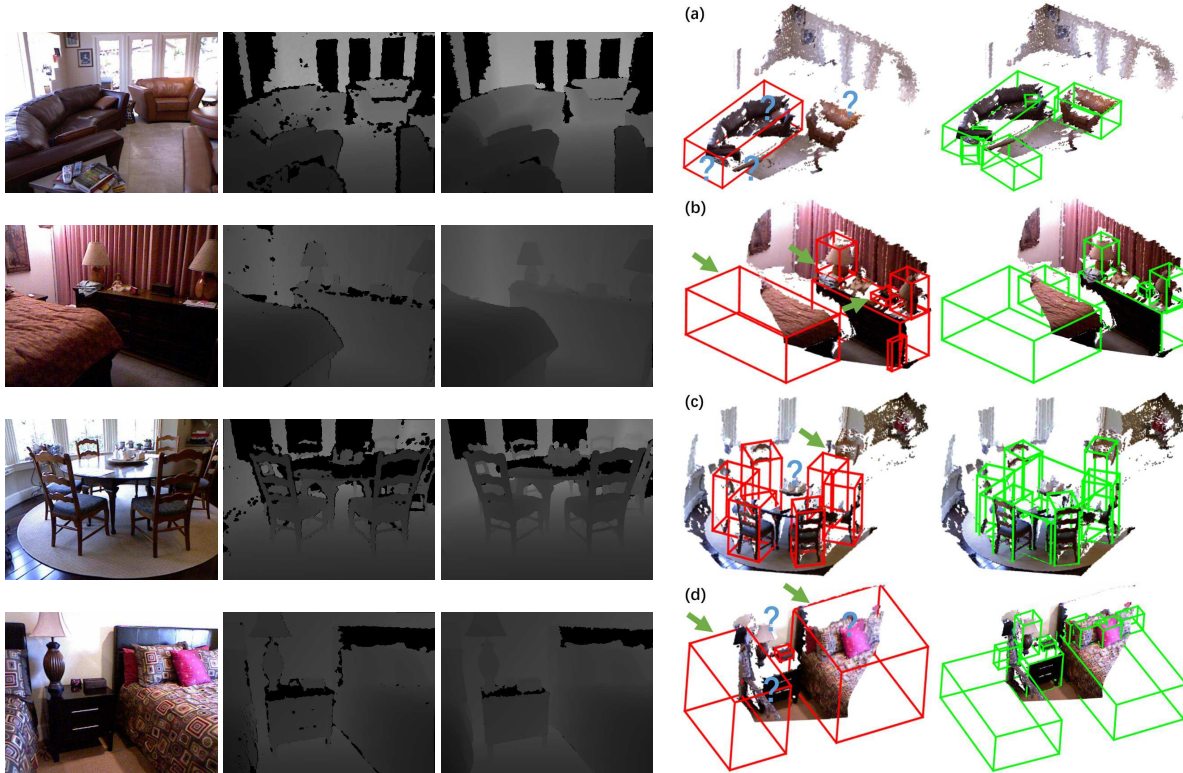


Figure 4. Examples of improved 3D annotations for 19 amodal 3D object detection classes and comparisons with annotations in SUN RGBD dataset [26]. Column 1: color images. Column 2: original single frame raw depth maps. Column 3: refined depth maps by integrating multiple depth maps within nearby video frames. Column 4: Ground truth 3D bounding boxes (red color) from [26]. Blue question marks represent missing object annotations. Green arrows point to problematic object annotations. Column 5: our improved 3D annotations (green color). As is shown in Column 4, notable issues include missing bounding boxes for salient objects, e.g., 2 sofas, 1 table and 1 pillow are missing in (a), 1 table is missing in (c), 1 lamp, 1 nightstand and several pillows are missing in (d), modal boxes for partial visible objects are incomplete, e.g., all bounding boxes for beds in (b) and (d), inaccurate 3D extensions and locations, e.g., 1 chair in (c) is mis-located, 1 lamp in (b) is floating above table surface, 1 box object in (b) has very loose bounding box. In comparison to examples shown in Column 4, we provide much more reasonable annotations for amodal 3D object detection research purpose. In this paper, we use original single frame depth maps as in column 2 as input instead of refined ones that were adopted in [24].

image plane and rotation matrix R_{tilt} for gravity alignment etc. Some examples and comparisons with annotations in [26] are shown in Figure 4. The improved annotations will be released on the authors’ website.

5. Experiments

In this section, we compare our algorithm with the currently best performing 3D detector [24] on the NYUV2 dataset [22] with the improved 3D bounding box annotations as described in Sec 4. Control experiment analysis and related discussions are also provided for better understanding the importance of each component in the designed 3D detection system. In the standard NYUV2 dataset split, the training set consists of 795 images and test set contains 654 images. We follow this standard for all the experiments. For our algorithm, we use the single frame depth map provided by the NYUV2 instead of the refined version in SUN-RGBD dataset.

3D Amodal Detection Evaluation

In order to compare the proposed approach to *deep sliding shapes* [24], we perform evaluation on 19 object classes detection task. We evaluate the 3D detection performance by using the 3D volume intersection over union (IoU) metric firstly defined in [25]. A detected bounding box is considered as a true positive if the IoU score is greater than 0.25. In the experiment, we set λ to 1 in the loss function. We use momentum 0.9, weight decay 0.0005, and “step” learning rate policy in Caffe, where base learning rate is 0.005, and γ is 0.1. We run SGD for 40000 mini-batch iterations during the training stage. In order to reduce the internal covariate shift, we normalized activations by adding BatchNorm Layers [14] to the 3D detection network.

In Table 1, we quantitatively compare with the state-of-the-art 3D approach algorithm “deep sliding shape” [24] on a 19-class detection task on the NYUV2 RGB-D dataset. Our method significantly outperforms [24] by a




















Methods																				mAP
[24](old gt3d)	64.4	82.3	20.7	4.3	60.6	12.2	29.4	0.0	38.1	27.0	22.1	0.7	49.5	21.5	57.8	60.5	49.8	8.4	76.6	36.1
[24]	62.3	81.2	23.9	3.8	58.2	24.5	36.1	0.0	31.6	27.2	28.7	2.0	54.5	38.5	40.5	55.2	43.7	1.0	76.3	36.3
Ours	36.1	84.5	40.6	4.9	46.4	44.8	33.1	10.2	44.9	33.3	29.4	3.6	60.6	46.3	58.3	61.8	43.2	16.3	79.7	40.9

Table 1. 3D Object Detection Performance Comparisons on 19 Classes on NYUV2 dataset. 1st row is evaluated using 3D annotations in [26]. The others are evaluated using the improved 3D annotations (see Sec 4).




















Methods																				mAP
img	27.9	64.5	24.5	1.5	33.1	46.0	20.3	1.7	28.7	32.1	24.6	3.0	43.4	27.7	49.6	46.7	27.6	1.3	66.0	30.0
img+HHA	33.1	83.9	29.8	6.0	43.1	46.3	25.3	1.87	30.9	32.9	24.3	4.1	58.3	40.3	54.8	59.6	39.6	3.5	69.5	36.2
img+d	38.9	85.2	37.5	11.4	46.5	47.1	29.9	4.2	43.3	37.3	30.8	1.3	59.8	44.1	57.7	63.8	39.4	11.6	75.5	40.1
img+d+ct	36.1	84.5	40.6	4.9	46.4	44.8	33.1	10.2	44.9	33.3	29.4	3.6	60.6	46.3	58.3	61.8	43.2	16.3	79.7	40.9
img+d+ct-3dreg	8.3	5.0	14.3	2.1	14.1	3.6	0.6	0.7	4.1	29.5	27.1	2.4	23.0	31.4	20.5	34.5	4.6	1.7	67.6	15.5

Table 2. Ablation study on NYUV2 dataset. "img": use color image only as input to our detection network. "HHA": depth embedding of [12]. "d": normalized depth map. "ct": context information. "3dreg": 3d regression offsets. "+": with. "-": without.

clear margin **4.6%** measured by mean Average Precision score (mAP). In particular, we achieve much better detection performances on difficult object categories reported in [24] such as door, tv, box, monitor. The reason is that in [24] the 3D box proposals network (RPN) relies on the quality of recovered 3D point cloud. But, in practice, the depth data from Kinect alike sensors are noisy and incomplete. Therefore, if the point cloud is sparse or empty for object instances such as tv or monitor, then the corresponding 3D anchor boxes are treated as negative 3D proposals and discarded. In contrast, our approach is more robust in such cases, since our 3D box initialization uses median value of segment pixel depths and 3D regression are based on learned 2.5D features (see Sec. 3.1), and hence neither depend on density nor geometries of 3D point clouds.

In addition, we list the results of [24] evaluated on the 3D annotations of [26] as a reference. Their results based on the improved 3D annotations are slightly better, which might be due to the fact that wrong labelings have been corrected in the new annotations.

We also provide qualitative results in Figure 5 and 6. True positive detections in Figure 5 indicate that 2.5D representation features are useful for detecting 3D objects with various of orientations, sizes and locations in complex indoor scenes. In Figure 6, we list several failure cases including wrong box size, inaccurate locations, wrong box orientations, and mis-classifications.

Ablation Study

To understand the importance of each component of our system, we conduct control experiments and list detection results in Table 2. We are reaching the following conclusions: 1) Color images contain rich 2D features for inferring object 3D full-extents. 2) The features encoded in depth map are complimentary to those in color images. 3) We nor-

malized the depth map by truncating depth value beyond 8 meters. It achieves 2.9% improvement than using HHA embedding as Horizontal disparity, Height above ground and Angle of local surface normal with inferred gravity direction. 4) Contextual information slightly improves the performance by 0.8%.

In order to demonstrate effectiveness of 3D regression learned by the proposed system, we remove 3D offsets and evaluate the initial 3D boxes in "img+d+ct-3dreg". The performance degrades dramatically by 25.4%.

Computation Speed

Our 3D detection system is developed based on the open source Caffe CNN library [15]. The training of 3D detector takes around 15 hours on an Nvidia Titan X GPU using CUDA 7.5 and cuDNN v4 support. The GPU usage is around 9 GB. During testing, the detection net takes 0.739s per RGB-D image pair, which is nearly **20x** faster than the Object Recognition Network (ORN) in [24].

6. Conclusion

We present a novel amodal 3D object detection system that directly learns deep features in RGB-D images without performing any 3D point reconstruction. Our system learns 2.5D visual appearance features from pairs of color and depth images. Experiments demonstrate that the 2.5D visual features are correlated to 3D object sizes, locations, and orientations. Our approach significantly outperforms the best performing 3D detector [24], which is truly a 3D approach, since it analyzes 3D point clouds.

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant IIS-1302164. We also thank Nvidia Corporation for the GPU donation.



Figure 5. Examples of detected true positive 3D amodal bounding boxes on NYUv2. 3D detections are rendered in 3D space in green. The corresponding object 2D locations are marked with red bounding boxes.

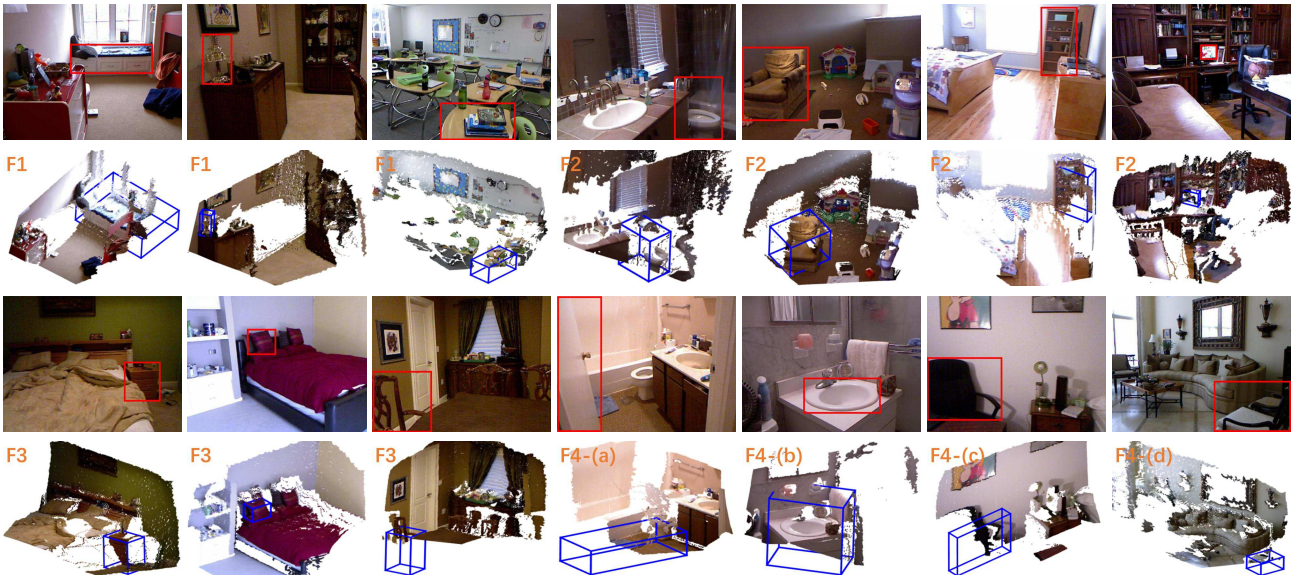


Figure 6. Examples of failure cases. 3D detections are rendered in 3D space in blue. The corresponding objects are marked with red bounding boxes. We show four types of failures. F1: box dimension errors. F2: orientation errors. F3: 3D location errors. F4: classification errors ((a) door detected as bathtub, (b) sink detected as toilet, (c) chair detected as tv, (d) chair detected as table).

References

- [1] P. Arbeláez, J. Pont-Tuset, J. T. Barron, F. Marques, and J. Malik. Multiscale combinatorial grouping. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 328–335, 2014.
- [2] T. P. Breckon and R. B. Fisher. Amodal volume completion: 3d visual completion. *Computer Vision and Image Understanding*, 99(3):499–526, 2005.
- [3] X. Chen, K. Kundu, Z. Zhang, H. Ma, S. Fidler, and R. Urtasun. Monocular 3d object detection for autonomous driving. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2147–2156, 2016.
- [4] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 886–893. IEEE, 2005.
- [5] 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.
- [6] S. Fidler, S. Dickinson, and R. Urtasun. 3d object detection and viewpoint estimation with a deformable 3d cuboid model. In *Advances in neural information processing systems*, pages 611–619, 2012.
- [7] J. Fredebon. The role of instructions and familiar size in absolute judgments of size and distance. *Perception & Psychophysics*, 51(4):344–354, 1992.
- [8] R. Girshick. Fast r-cnn. In *ICCV*, 2015.
- [9] R. Girshick. Fast r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1440–1448, 2015.
- [10] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.
- [11] S. Gupta, P. Arbeláez, R. Girshick, and J. Malik. Aligning 3d models to rgb-d images of cluttered scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4731–4740, 2015.
- [12] S. Gupta, R. Girshick, P. Arbeláez, and J. Malik. Learning rich features from rgb-d images for object detection and segmentation. In *European Conference on Computer Vision*, pages 345–360. Springer, 2014.
- [13] V. Hedau, D. Hoiem, and D. Forsyth. Thinking inside the box: Using appearance models and context based on room geometry. In *European Conference on Computer Vision*, pages 224–237. Springer, 2010.
- [14] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.
- [15] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 675–678. ACM, 2014.
- [16] A. Kar, S. Tulsiani, J. Carreira, and J. Malik. Amodal completion and size constancy in natural scenes. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 127–135, 2015.
- [17] B.-s. Kim, S. Xu, and S. Savarese. Accurate localization of 3d objects from rgb-d data using segmentation hypotheses. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3182–3189, 2013.
- [18] D. Lin, S. Fidler, and R. Urtasun. Holistic scene understanding for 3d object detection with rgb-d cameras. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1417–1424, 2013.
- [19] L. S. Liu W, Ji R. Towards 3d object detection with bimodal deep boltzmann machines over rgb-d imagery. In *CVPR*, 2015.
- [20] N. Payet and S. Todorovic. From contours to 3d object detection and pose estimation. In *2011 International Conference on Computer Vision*, pages 983–990. IEEE, 2011.
- [21] Z. Ren and E. B. Sudderth. Three-dimensional object detection and layout prediction using clouds of oriented gradients. *CVPR*, 2016.
- [22] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus. Indoor segmentation and support inference from rgb-d images. In *European Conference on Computer Vision*, pages 746–760. Springer, 2012.
- [23] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
- [24] S. Song and J. Xiao. Deep sliding shapes for amodal 3d object detection in rgb-d images. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.
- [25] X. J. Song S. Sliding shapes for 3d object detection in rgb-d images. In *ECCV*, 2014.
- [26] X. J. Song S, Lichtenberg SP. Sun rgb-d: A rgb-d scene understanding benchmark suite. In *CVPR*, 2015.
- [27] H. Su, S. Maji, E. Kalogerakis, and E. Learned-Miller. Multi-view convolutional neural networks for 3d shape recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 945–953, 2015.
- [28] P. Viola and M. Jones. Robust real-time object detection. *International Journal of Computer Vision*, 4, 2001.
- [29] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1912–1920, 2015.