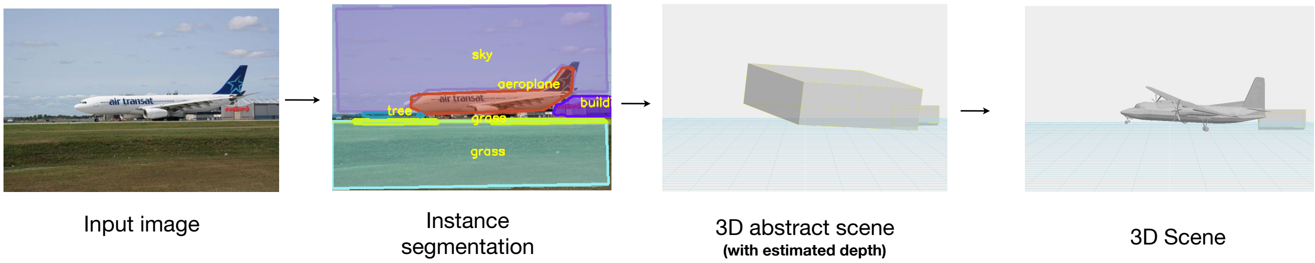
3D Scene Generation From Real-world Images

Flora Tasse
Streem, Inc.
flora@streem.pro

Pavan Kamaraju
Streem, Inc.
pavan@streem.pro

Ghislain Fouodji
Streem, Inc.
ghislain@streem.pro



Input image

Instance segmentation

3D abstract scene
(with estimated depth)

3D Scene

A 3D scene generated by our proposed network. From left to right: (1) the input RGB image, (2) instance segmentation, (3) a preliminary 3D scene generated by projecting 2D predictions to 3D using existing depth estimation methods, (4) 6-DOF pose and 3D object retrieval for relevant objects. Note: Training data included SUN2012 [13] and Pascal 3D+ [4].

Abstract

Scene understanding is critical in Computer Vision applications, particularly in settings like Augmented/Virtual Reality where interactions between the virtual environment and the physical world are needed. We present a novel end-to-end learning method for generating semantic-rich 3D *scenes from images. Scene understanding techniques have often been limited to the* 2D *space with object detection and segmentation. While new methods infer the 3D models and* 6*-DOF object pose from images, they either support very few object categories or assume object locations are known. Our solution learns to detect, classify, segment, retrieve 3D models and poses from an image to generate* 3D *scenes. Our proposed network architecture builds on top of MaskR-CNN to infer* 3D *information. Our shape retrieval has a Top-*1 *accuracy of about* 40% *on Pascal* 3D+*, on par with prior work on single object retrieval and pose estimation.*

1. Introduction

Recent advancements in object detection and instance segmentation help us not only in recognizing objects e.g. chairs, cups etc., but also understanding the scene in 2D. Moving this understanding from 2D to 3D requires retrieving 3D models and placing these objects in 3D space with correct pose and scale. Such 3D scene representation is critical in areas like Robotics and Augmented/Mixed Reality.

Recent works in shape retrieval, [9, 8, 2, 11, 12], use convolutional neural networks (CNN's) to retrieve a 3D model with pose using a learned similarity between RGB image features and pre-computed features from synthetic renderings of 3D models. However, recovering both the 3D model and pose from a similarity measure can be computationally expensive when applied at scale since it requires very large number of shape renderings. Additionally, it limits the estimated pose to three degrees of freedom. [5] address this in their recent work with a method that first predicts a pose through 2D projections of 3D bounding box corners and then use this pose as a prior for 3D model retrieval.

We take an alternative approach and present a scalable end-to-end solution to (1) detect the objects, (2) predict and retrieve the 3D model for the detected objects and (3) predict the 6-DOF pose for detected objects. We achieve this by building on top of MaskRCNN [6]. We add three extra output heads to MaskRCNN network to recover 6-DOF pose and feature vectors for the detected objects. We also attach a second network that learns the similarity between the feature vectors and depthmap descriptors (rendered and pre-computed offline from 3D models) to predict the 3D model of the detected object. This provides an end-to-end solution from detection to retrieval with pose. In comparison, [5] work assume object detection as a separate problem and focus on shape retrieval and pose estimation, trained separately. Similarly, [12] use pre-computed bounding box proposals to recover the 3D representation of a scene.

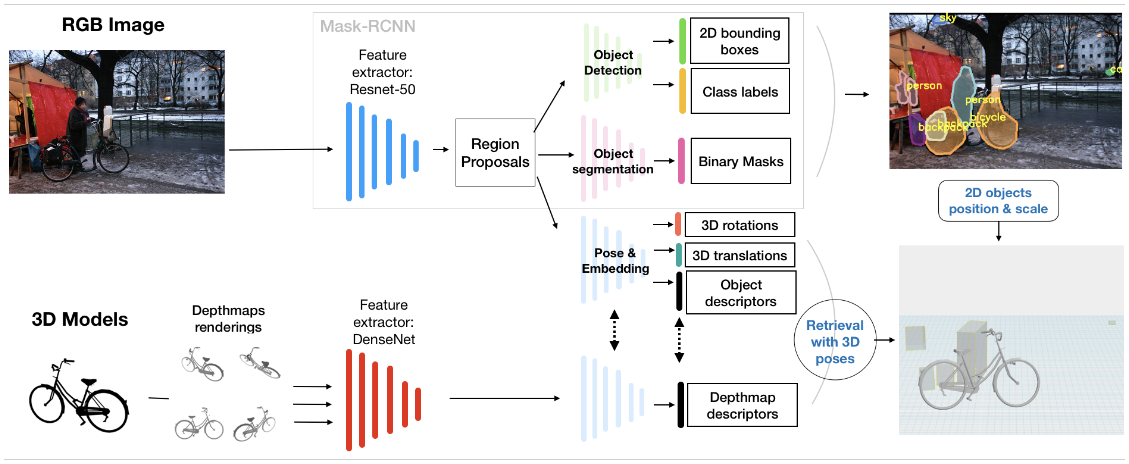


Figure 1. Our network architecture. (Top) We modify MaskRCNN, the state-of-art in instance segmentation by adding 3 additional output layers: 3D rotations, 3D translations, and 500-D object descriptors. (Bottom) We attach a second network based on DenseNet [7] classifier, where the classification output is replaced with a regressor that learns depthmap descriptors. The last layers in both networks have shared weights such that the network learns common features in images and depthmaps that enhances similarity learning.

2. 3D scene generation

We describe our approach for detecting, segmenting, and predicting pose of objects detected in an image below.

MaskRCNN is the current state-of-art in instance segmentation, with the ability to predict bounding boxes, classes and segmentation masks. It relies on the well-studied Feature Pyramid Network to extract image features at multiple scales and use an ROI (Region-Of-Interest) pooling method to produce object features for individual region proposals. The original MaskRCNN network generates 3 outputs from these object features: 2D bounding boxes, class labels and 2D binary masks per object. We modified (See Sec. 2.1 for the modifications) the original network architecture (See Fig.1 for overview) and trained an open-source implementation of MaskRCNN [1] on the diverse SUN 2012 dataset[3], with 400 classes. 3D models were retrieved from the PASCAL 3D+ dataset.

2.1. Network architecture

In addition to the three heads in the MaskRCNN network (2D bounding boxes, categories, segmentation), we add extra output layers: (1) 3D object rotations (2) 3D translations and (3) 500-D vectors representing the objects descriptors.

We use the quaternion representation of 3D rotation, commonly used for pose regression [10]. Each rotation is represented by a unit quaternion $q = w, x, y, z$, with the additional constraint $w >= 0$. 3D translation is simply represented by 3 scalars t_x, t_y, t_z representing the X, Y and Z axes. Thus 6-DOF pose is a 7D vector consisting of both rotation and translation parameters.

To retrieve the right 3D shape, we compute descriptors for 3D shapes and 2D objects such that given the descriptor of an object in an image, we can provide the relevant 3D shape for it. Hence, we add a descriptor head to the MaskRCNN network to extract them. We also add a separate network that generates descriptors for synthetically generated depthmaps of 3D shapes. The aim is to train both networks such that the distance between object descriptors and shape descriptors represents the similarity between an object in a scene and its shape.

2.2. Loss function

To jointly learn 3D object poses and their descriptors, we use two loss functions to obtain a total loss function

$$L_i = L_i^{pose} + L_i^{dtr}$$

.

Similar to prior work for rotation regression [10], we use the mean squared error of unit quaternions for computing the pose loss (L_i^{pose}).

$$L_i^{pose} = ||q_i - q_i^{gt}||_2 + 0.1||t_i - t_i^{gt}||_2$$

where (q_i, t_i) is the predicted quaternion and translation of an object i, and (q_i^{gt}, t_i^{gt}) is the ground-truth.

For descriptor loss (L_i^{dtr}), we use the well-known triplet loss function which maximizes the distance between negative pairs while reducing the distance between similar pairs:

$$L_i^{dtr} = ||d_i - s_i^{gt}||_2 + ||d_i - s_j^{gt}||_2 + m$$

where m is the margin (by default $m = 1$) and s_j the descriptor of a random object with a shape different from i.

2.3. Shape retrieval

Once the above network has been trained with our loss function, we can generate descriptors for arbitrarily generated depthmaps and save them offline for fast retrieval with

Table 1. View Estimation using ground-truth detections on Pascal 3D+ Val.

Category Agnostic	aero	bike	boat	bottle	bus	car	chair	table	mbike	sofa	train	tv	mean
MedErr ([5])	10.9	12.2	23.4	9.3	3.4	5.2	15.9	16.2	12.2	11.6	6.3	11.2	11.5
MedErr (Ours)	52.7	74.0	62.8	11.6	22.2	54.8	43.0	21.1	47.9	37.8	19.9	23.3	33.6
$Acc_{\pi/6}$ ([5])	0.80	0.82	0.57	0.90	0.97	0.94	0.72	0.67	0.90	0.80	0.82	0.85	0.81
$Acc_{\pi/6}$ (Ours)	0.31	0.19	0.28	0.86	0.69	0.40	0.36	0.71	0.34	0.37	0.64	0.61	0.47

Table 2. Retrieval results: Top-1 retrieval accuracy using ground-truth detections and poses on Pascal 3D Val.

Method	aero	bike	boat	bottle	bus	car	chair	table	mbike	sofa	train	tv	mean
Top-1-Acc ([5])	0.53	0.38	0.51	0.37	0.79	0.44	0.32	0.43	0.48	0.33	0.66	0.72	0.497
Top-1-Acc (Ours)	0.46	0.54	0.53	0.20	0.44	0.31	0.28	0.62	0.55	0.51	0.43	0.40	0.397

object descriptor queries at runtime. Similarly to [5], we use pose prior to improve retrieval accuracy. When given an object descriptor and its predicted query, we restrict retrieval to only those depthmaps with a similar pose, and return the shape whose depthmap descriptor is closest to the query.

The above provides a framework for jointly estimating the 6-DOF pose and relevant models for objects in a scene.

3. Results

Here, we present pose estimation and shape retrieval results on the Pascal3D dataset, a popular dataset for pose estimation. It was used for shape retrieval by [5] for the first time and thus we compare our work mainly against them with quantitative results on the validation dataset.

Our results in comparison with [5] on pose estimation and shape retrival are shown in Table 1 and Table 2 respectively. In line with previous pose estimation evaluation on this dataset, we only compute metrics for non-occluded and non-truncated objects. The median viewpoint estimation error ($MedErr$), computed from the angle difference between predicted and the ground-truth pose, increases from 11.5 to 33.6, and the Top-1 retrieval accuracy $Top-1-Acc$ decreases from 50% to 40%. There is a net decrease in performance when using our end-to-end solution as our neural network has a lot more tasks to learn when compared to learning only pose or shape retrieval.

It is interesting to see that when occluded and truncated objects are also taken into account, $MedErr$ is 30% and $Top-1-Acc$ is 34.7%, indicating that our network is able to use cues from surrounding objects to perform pose and retrieval. Note that all previous work including [5] do not present performance data for occluded/truncated objects.

Moreover, unlike prior work, our network performs instance segmentation on top of pose estimation and retrieval. This is the first time this has been done to our knowledge.

There have been other work in 3D scene generation from images such as IM2CAD [9]. However IM2CAD is limited to indoor scenes and 8 object categories, with no instance segmentation. For objects with no available 3D shapes we provide cuboids as shapes.

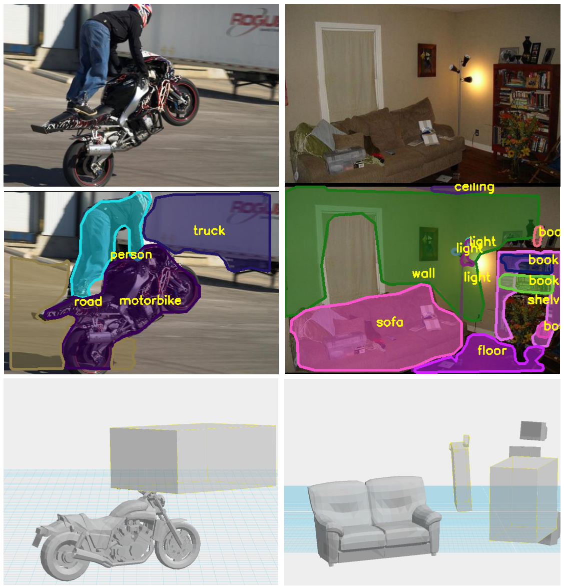


Figure 2. Examples of our 3D scene generation. Top to bottom: (1) input image, (2) instance segmentation, (3) 3D scene generated.

4. Conclusions and Future Work

3D scene generation from RGB images is a critical and challenging task and applicable in a wide variety of vision applications. We address this problem and present our ongoing work on the first end-to-end learning method for generating semantic rich 3D scenes from RGB images. Our solution builds on MaskRCNN to not only detect and classify objects but also retrieves a 3D model with 3-DOF pose. Our approach is scalable and matches the state-of-the-art for shape retrieval, detecting up to 400 different objects. We are further investigating the use of keypoints to improve pose.

References

[1] Waleed Abdulla. Mask r-cnn for object detection and instance segmentation on keras and tensorflow. `https://github.com/matterport/Mask_RCNN`, 2017.

[2] Mathieu Aubry and Bryan C. Russell. Understanding deep features with computer-generated imagery. In *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, 2015.

[3] Angel X. Chang, Thomas A. Funkhouser, Leonidas J. Guibas, Pat Hanrahan, Qi-Xing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. Shapenet: An information-rich 3d model repository. *CoRR*, abs/1512.03012, 2015.

[4] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision*, 111(1):98–136, Jan. 2015.

[5] Alexander Grabner, Peter M. Roth, and Vincent Lepetit. 3D Pose Estimation and 3D Model Retrieval for Objects in the Wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[6] K. He, G. Gkioxari, P. Dollr, and R. Girshick. Mask r-cnn. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2980–2988, Oct 2017.

[7] G. Huang, Z. Liu, L. v. d. Maaten, and K. Q. Weinberger. Densely connected convolutional networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2261–2269, July 2017.

[8] Moos Hueting, Viorica Ptrucean, Maks Ovsjanikov, and Niloy J. Mitra. Scene structure inference through scene map estimation. *VMV*, 2016.

[9] H. Izadinia, Q. Shan, and S. M. Seitz. Im2cad. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2422–2431, July 2017.

[10] Alex Kendall, Matthew Grimes, and Roberto Cipolla. Posenet: A convolutional network for real-time 6-dof camera relocalization. 2015.

[11] H. Su, S. Maji, E. Kalogerakis, and E. Learned-Miller. Multi-view convolutional neural networks for 3d shape recognition. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 945–953, Dec 2015.

[12] Shubham Tulsiani, Saurabh Gupta, David Fouhey, Alexei A. Efros, and Jitendra Malik. Factoring shape, pose, and layout from the 2d image of a 3d scene. In *Computer Vision and Pattern Regognition (CVPR)*, 2018.

[13] Jianxiong Xiao, Krista A. Ehinger, James Hays, Antonio Torralba, and Aude Oliva. Sun database: Exploring a large collection of scene categories. *Int. J. Comput. Vision*, 119(1):3–22, Aug. 2016.