Instance Segmentation using Depth and Mask RCNNs

Mohamed Masoud
masoud@stanford.edu

Rewa Sood
rrsood@stanford.edu

June 9, 2018

1 Abstract

There are currently many models that produce accurate instance segmentation results using RGB images. However, few models incorporate depth information. These models used different methods and architectures to incorporate the estimated or measured depth for object detection, semantic segmentation and instance segmentation. This project analyzes whether adding measured depth in addition to RGB information improves instance segmentation results and how well transfer learning works with a very small dataset. For that purpose, we modified the Mask RCNN network to include features from depth images in the object detection pipeline. Despite the marginally better results of the proposed RGB-D model over the RGB-only model, it is difficult to make a definitive conclusion as to whether the depth information significantly helps instance segmentation due to the small dataset size. However, transfer learning produces more accurate results, implying that a larger dataset could improve the results further.

2 Introduction

Instance segmentation is an important part of applications such as automated driving. The last few years has seen rapid development of fast scene understanding algorithms. However, many algorithms do scene understanding tasks on 2D data sets. Recently, Facebook AI Research (FAIR) has released code for a Mask R-CNN backbone which is state-of-the-art for 2D scene understanding tasks like instance segmentation. In this project, we investigate whether incorporating the depth features would further enhance the object detection component of the instance segmentation and accordingly would improve the Mask R-CNN overall performance. We compare the RGB-D Mask RCNN model to a baseline of RGB only model and a transfer learning model that fine tunes a pretrained Mask RCNN model [4] on the large COCO dataset [2] with the small NYU dataset RGB images [1]. The input to our method is the annotated NYU dataset for both transfer learning and the RGB-D model. The outputs are bounding boxes, instance masks, and class membership information. Improving instance segmentation would help applications such as automated driving become even safer than they already are because these applications would be able to differentiate between drivable road and other objects more effectively.

3 Related Work

There are some scene understanding studies using RGB-D images focused on object detection and semantic segmentation. [8] introduced a method to infer the overall 3D structure of the image scene and then parse its object for detection and instance segmentation. This study used the same NYU RGBD dataset. Furthermore, in [9], the authors constructed a large 3D RGB-D dataset with detailed annotations for scene understanding. [10], designed a generalized modified architecture of RCNN and a three channel representation of the depth images for object detection and segmentation. [11] uses fully convolutional networks for semantic segmentation. [3], introduced a method to estimate depth from the RGB image. The RGB-D is then used to train a modified Fast R-CNN for object detection [5]. Our work is guided by [3] to modify Mask R-CNN architecture for our study of RGB-D instance segmentation.

4 Dataset

We are using the dataset from NYU [1]. The NYU dataset contains 1449 densely labeled pairs of aligned

Kinect depth and RGB images. The dataset was split into 80/10/10 train/validation/test sets. The original image dimensions were rescaled to 256x256x3 to speed up runtime. Since we are doing transfer learning using the COCO dataset with the same architecture, we only use the masks that match the COCO dataset classes. The NYU dataset provides labels for 895 object classes while the COCO dataset has only 80 classes. We created class mappings that would capture low level features for the corresponding classes. The other significant part of the dataset are the instance maps and labels associated with these aligned images. The dataset provides an efficient way to extract the following triplet: the RGB image, the gray scale depth image and pixel level labels/masks from one data file 1. The easy extraction of this triplet made the dataset a great candidate with which to train the Mask RCNN, since all the masks are readily available. However, the size of the labeled dataset imposed a major challenge to the study. The small amount of data made overfitting inevitable, even for the transfer learning experiment. Additionally, the NYU dataset is labeled such that there is one bounding box per type of object, regardless of the relative locations of the objects. for example, in Figure 1, all four pictures have the same bounding box. The main issue with this labeling method is that some of the background features are also included with the pertinent features. This mixing of features confuses the network as to which components actually belong to the object. Additionally, this labeling scheme is vastly different from that used in the COCO dataset. This difference makes it difficult for the network to learn during transfer learning.

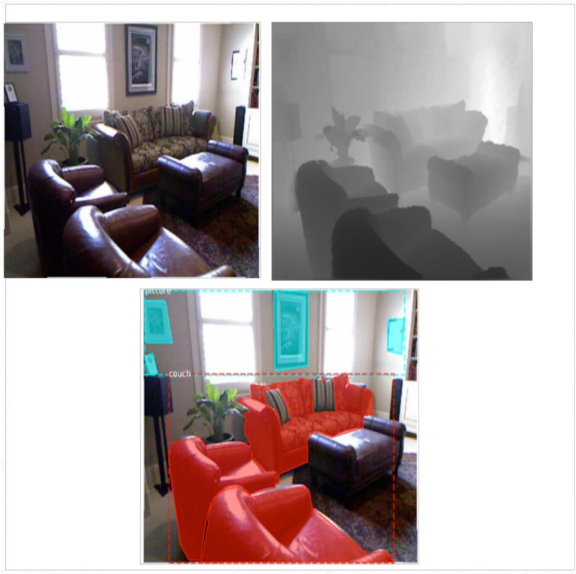


Figure 1: *This figure shows an example of the NYU data: RGB, Depth, and Ground Truth labels*

5 Methods

This dataset provides a unique approach to exploring the RGB-D plus Mask RCNN problem. The NYU dataset encodes depth in a 2D image that mirrors its associated RGB image. We are proposing an architecture that trains one Mask R-CNN pipeline: a concatenation of the RGB and depth image features. The approach we took involved instance level feature concatenation, where for each instance, a fixed-length feature vector was extracted by the fully convolution network layer and the depth and RGB feature vectors were concatenated. We only trained one RGB RPN and used it for both the RGB and depth pipelines. The architecture used in this study is a modification of the FAIR's Mask R-CNN 2.

The RGB image backbone and RPN pipeline is a complete Mask-RCNN pipeline that takes the single scale of the entire RGB image through the Faster-RCNN ResNet-50 with Feature Pyramid Network (FPN) backbone layers [7]. The extracted RGB image convolution feature map output of the backbone layers is then fed into the Region Proposal Network (RPN) model. The RPN generates the bounding box proposals and then detection target layers subsample these proposals through non-max suppression and generate masks for each Region of Interest (RoI) proposal. The depth image pipeline is similar to the RGB pipeline, where the depth image pipeline takes an RGB representation of the gray scale depth image into a separate ResNet-FPN backbone. For each ROI generated by the RGB network RPN model, both extracted RGB and depth convolution feature maps pass separately through the FPN feature pyramid, ROIAlign pooling, and two fully connected (FC) convolution layers. The separate RGB and depth output encodings are then squeezed and concatenated to form the class logits for the softmax classifier output layer and to feed the bounding box regression output layer. The FPN, ROIAlign pooling, FC

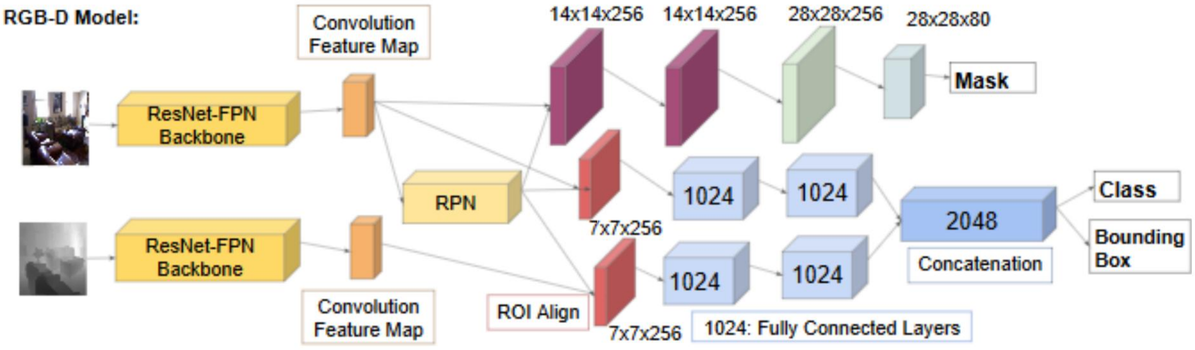


Figure 2: *RGB-D model*

layers, and concatenation form the network head, which is the only part trained during transfer learning.

It is important to note that the concatenation of the encoded RGB and depth features is performed on the ROI level. Conceivably, the ROI proposals derived from the depth images could be different from the proposals derived from the RGB images because the two types of images contain different information. Therefore, in order to avoid matching the ROI proposals between the RGB and the depth images, we use only the RGB ROI proposals for both the pipelines. Additionally, since the RGB and depth features describe the same image, it is reasonable to use RGB bounding box proposals for the depth features as well.

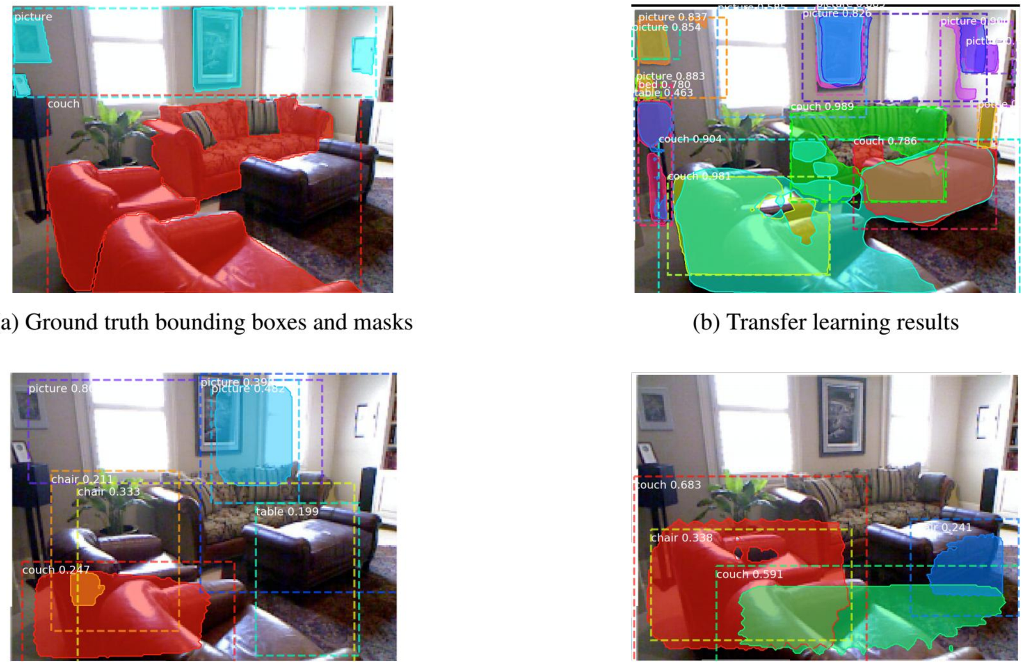
Another important note is that the depth mask branch is omitted. The reason is that we are focusing our study on the object detection part of the instance segmentation and the boundaries of objects are fuzzy in depth masks.

6 Experiments and Results

We compared three different models: the Mask-RCNN architecture using only RGB images, the modified architecture using RGB and depth images, and transfer learning, where the network heads were fine tuned, using a pretrained COCO dataset model. For the first two models, we train the entire network with the small NYU training dataset. Additionally, the images were first resized to 256x256 from their original size to help decrease the amount of time required for the network to train and to make sure the images were square for all models. Each model was trained using Resnet-50 instead of Resnet-101 so that there were fewer layers to train. The results from each of these models with no hypertuning are shown in Figure 3. The network was trained for 20 epochs with a batch size of 2 and a learning rate of 0.01.

We studied three hyperparameters for tuning: learning rate, regularization strength, and data augmentation. The learning rate was originally set at 0.01. Keeping the rest of the parameters the same, we changed this rate to 0.001. Looking at the loss curves from the runs that used the original learning rate, we noticed that the loss initially drops significantly and subsequently plateaus out 4. Thus, we tried lower learning rate of 0.001 in pursuit of better convergence. Again, keeping all other parameters the same, we modified the regularization strength by increasing the weight decay from 1e-4 to 1e-3. We chose to increase the regularization to try to help the network overfit less. For data augmentation, we randomly flipped the image over the vertical axis with a probability of 50%. Regularization and data augmentation produced loss curves similar to those in 4.

The loss curves show that for both the training and validation losses and the bounding box and class losses, the RGB-D network achieves a marginally lower loss overall compared to the RGB model. In addition, the mAP scores for each hyperparameter experiment in Table 1 indicate that the RGB-D network produces similar, if not better, results to the RGB network. One possible explanation for the plateauing of the loss curves after epoch 2 is that there are not enough examples in the NYU dataset to define the multidimensional problem space. Based on Table 1, none of the hyperparameter changes improved



(a) Ground truth bounding boxes and masks

(b) Transfer learning results

(c) RGB results

(d) RGB-D results

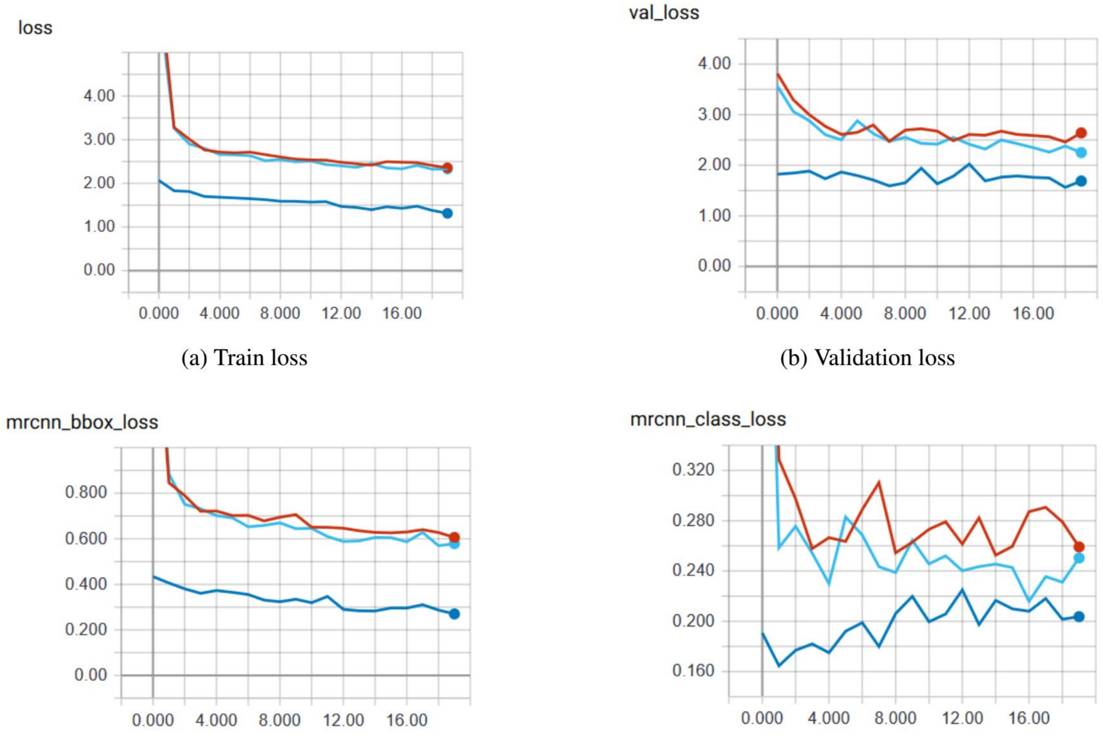
Figure 3: Ground Truth vs transfer learning, RGB, and Depth Results

	Original	Augment	Weight Decay	Learning Rate
RGB (%)	12.22	6.02	**10.85**	6.7
RGB-D (%)	**20.63**	**6.67**	10.23	**7.52**
Transfer (%)	36.19	32.3	36.75	36.49

Table 1: mAP Results for various experiments

the RGB and RGB-D models, however there seems to be some improvement in the transfer learning mAP scores for changes in weight decay and learning rate. One possible explanation for the reduction in performance for the RGB and RGB-D models is that there was not enough data to learn originally, therefore, the lower learning achieved slower but not better convergence. The L2 regularization did not help with the generalization problem; it may have caused the model to learn less from the already small dataset: a high bias and high variance problem. Interestingly, data augmentation produced the worst results by far. This outcome might be due to the fact that the NYU bounding boxes mark all instances of an object as one object. So, the original bounding box and objects versus the flipped version might produce very different features.

Looking at the different model results for a specific image from the test set provides some insights into how each model has been trained. Exploiting the COCO dataset pretrained features, the transfer learning model detects most of the objects in the image, even the unlabeled objects such as the object incorrectly labeled 'bottle' in 3b. This fine tuned model also finds each separate object: pictures and each separate couch. It misclassified certain objects and gets a bit confused due to the NYU data labeling: many pictures with high scores are detected on the



(a) Train loss

(b) Validation loss

(c) Bounding Box loss

(d) Class loss

Figure 4: Losses for RGB(red), RGB-D(Light blue), and Transfer learning(Dark blue)

top part of the image.

Comparatively, the RGB and RGB-D models perform less accurately. Both models' detections are moderately similar. However, there are some interesting distinctions. The RGB-D detections have higher accuracy and class scores in couch/chairs area where the depth features are more prominent. In contrast, the pictures area is lacking depth features and therefore the depth encoding does not contribute to the detection of pictures in this area.

7 Conclusion and Future Work

In our study, we modified the Mask-RCNN architecture to incorporate the Kinect depth measurements with the object detection part of Mask-RCNN instance segmentation. We compared the performance against a baseline of RGB-only model and a more accurate transfer learning model that fine tunes a pretrained COCO dataset using the same NYU data RGB images. The NYU depth V2 dataset is being used for this study. The annotated part of the NYU dataset is a small set of 1449 triplets [RGB, depth, and Mask images]. The results show a marginal enhancement of performance incorporating the depth features over the corresponding RGB-Only model.

The future work on the proposed model should address the challenges of the present study. Despite the readiness of extracting the masks, the NYU dataset introduces major challenges due to its small labeled data size and the flaws of the objects bounding box labeling. Princeton SUN RGB-D 9 dataset could be considered as a viable alternative. The future study would also benefit from more computational power and resources, as the proposed model and deep models like Mask-RCNN require a big computational budget. Addressing these challenges would greatly improve the scope and the results of the study.

8 Contribution

Both authors contributed equally to this work.

References

[1] https://cs.nyu.edu/s̃ilberman/datasets/nyu_depth_v2.html

[2] http://cocodataset.org/

[3] Yuanzhouhan Cao, Chunhua Shen and Heng Tao Shen. Exploiting Depth from Single Monocular Images for Object Detection and Semantic Segmentation, *IEEE Transactions on Image Processing*, eprint arXiv:1610.01706 (2016).

[4] Kaiming He, Georgia Gkioxari, Piotr Dollar and Ross B. Girshick. Mask R-CNN. 2017 IEEE International Conference on Computer Vision (ICCV), (2017), pages 2980-2988.

[5] R. Girshick, ”Fast R-CNN,” 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, 2015, pp. 1440-1448.

[6] Kaiming He, Shaoqing Ren, Ross Girshicka and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks, NIPS'15 Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1 Pages 91-99

[7] Tsung-Yi Lin, Piotr Dollr, Ross Girshick, Kaiming He, Bharath Hariharan, Serge Belongie. Feature Pyramid Networks for Object Detection, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[8] Nathan Silberman, Derek Hoiem, Pushmeet Kohli and Rob Fergus. Indoor Segmentation and Support Inference from RGBD Images ,European Conferene of Computer Vision (ECCV), 2012.

[9] Shuran Song, Samuel P. Lichtenberg and Jianxiong Xiao. SUN RGB-D: A RGB-D Scene Understanding Benchmark Suite ,2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

[10] Saurabh Gupta, Ross Girshick, Pablo Arbelaez and Jitendra Malik. Learning Rich Features from RGB-D Images for Object Detection and Segmentation ,2European Conference of Computer Vision (ECCV), 2014.

[11] Johnathan Long, Evan Shelhamer, Trevor Darrell. Fully Convolutional Networks for Semantic Segmentation, 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)