Improving Image Semantic Segmentation with RGB-Depth

Mengxin Zhang
mengxinz@andrew.cmu.edu

Xinyu Huang
xinyuh1@andrew.cmu.edu

Jiayuan Li
jiayuanl@andrew.cmu.edu

Jiqian Dong
jiqiand@andrew.cmu.edu

Abstract

This project mainly focuses on image semantic segmentation based for RGBD images. Since traditional image segmentation mainly focus 2D (RGB) images, it is sometimes hard to draw a smooth and clear decision boundaries for each object as the contrast in pixels are not always evident in RGB images. Therefore, we believe by adding some high contrast features such as depth information to this segmentation task, the boundaries between different objects can be drawn more clearly since there rarely exist 2 perfectly aligned objects. As more and more open source RGBD datasets become available for everyone, we are aiming to boost the accuracy of imagery semantic segmentation by transferring the current state-of-art frameworks to these RGBD images. Specifically, we train a pixel wise segmentation algorithm UNet on RGBD dataset and hope to test out an effective way of combining the original RGB features with the newly added depth information.

1 Introduction

Image segmentation is one of the areas of computer vision that is maturing very rapidly. Today, thanks to deep learning, there is a plethora of pre-trained models for solving the problem in RGB images (YOLO, RCNN, Fast RCNN, Mask RCNN, etc.). However, as the image is purely planar, the depth information is always lost when taking the photo, which further makes it hard to draw a clear decision boundary between 2 objects. The consequence of it is the accuracy of image segmentation drops significantly as the background becomes messier. In this project, we are aiming to achieve a better pixel wise segmentation of images by adding depth information into consideration and transferring the newly proposed CNN network (U-Net, Segnet, etc) into the segmentation of RGBD images.

RGBD camera, which have become cheaper and more popular these days, can be applied into the image segmentation domain. With the depth information, we can build better networks. Equipped with multiple well labeled RGBD datasets, we are diving into the semantic segmentation of RGBD images and will finally obtain the image masks for each object and compare the accuracy against the traditional 2D image segmentation network such as U-Net.

The challenges of this task include: 1) The number of classes in the dataset is large and there exist a lot of unlabeled pixels in the back grounds 2) The way to combine depth information with original RGB pixels remains is not thoroughly researched, the different scale, $0 - 255$ for RGB data and $0 - \infty$, may cause trouble when simply concatenating the channels.

2 Related work

Semantic segmentation is widely used in many fields, such as object segmentation, the road scenes recognition in self-driving, and the segmentation of biological images. We plan to use the existing CNN model based on U-Net [1] as our baseline model. The u-net is a "fully convolutional network" [2] architecture for fast and precise segmentation of images. It has the ability to work with very few training images and yields more precise segmentation, which can alleviate the problem of inadequate images in the dataset. The main idea in U-net is to supplement a usual contracting network by successive layers. Replacement of the last fully connective layers with deconvolutional layers is a popular approach in semantic segmentation to keep spatial information as much as possible, such that has the ability to yields precise pixel-wise segmentation. The interest in semantic segmentation is active [3] [4] [5]. Besides U-net, Segnet [3] is another popular semantic pixel-wise segmentation algorithm, also uses the similar encoder-decoder architecture. The reason why encoder-decoder architecture becomes popular is that the concatenation of the image before convolution and after convolution is able to keep both large-scale and tiny-scale spatial detail and significantly increase the performance of semantic segmentation. Therefore they modify the pooling operators by replacing them with upsampling operators,

32nd Conference on Neural Information Processing Systems (NIPS 2018), Montréal, Canada.

and finally combine the high resolution features from the contracting path with upsampled output to provide the segmentation map. The resulting network won the EM segmentation challenge at ISBI 2012 by a large margin.

3 Method

3.1 Network and structure

We used U-net with RGB images as our baseline model, and then used U-net on RGB-D images as our initial trial. We would like to test out whether the model itself is suitable in the situation, thus no image preprocessing are performed before hand. We applied the parameters as stated in the U-net original paper [1] and no parameter tuning has been performed at this stage.

An overview of U-net architecture is illustrated in Figure 1. Our network architecture largely assembles the figure except for the following aspects:

1. The image size is different as the images in our dataset is 480×640. At each stage in the network (where the image size basically does not change), rather than shrinking the image a little bit, we would keep the width and height by padding.
2. Rather than using ReLU as the activation function, we used ELU non-linearity which has been shown to have a better performance in CNN.
3. We applied batch normalization after each convolutional layer.

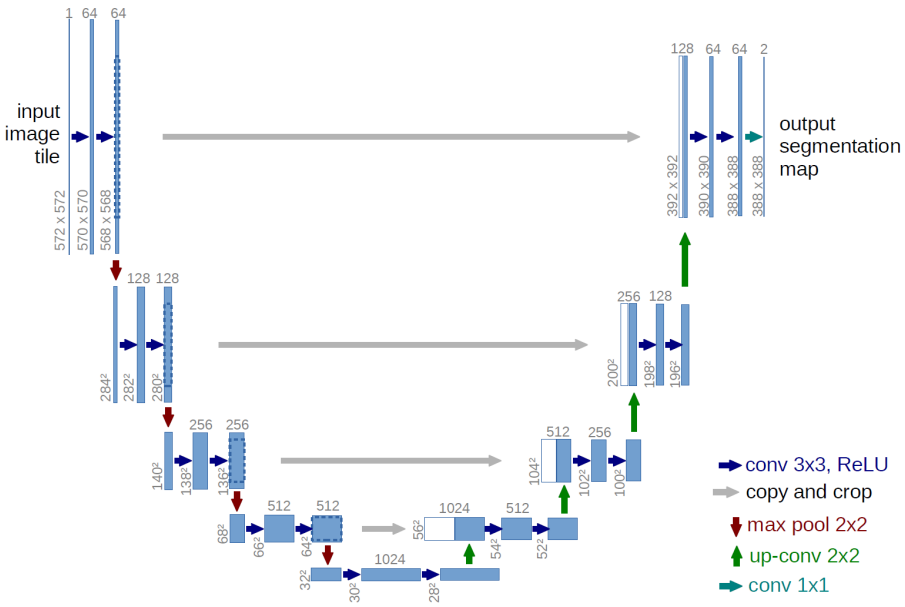

Figure 1: U-net architecture

3.2 Performance evaluation metrics

In this segmentation task, we use both pixel wise accuracy and the F scores at different intersection over union (IoU) thresholds as the metric for evaluating the model performance. The pixel wise accuracy is calculated as the fraction of truly classified pixels among all the pixels in the image. The F Score value is calculated based on the number of true positives (TP), false negatives (FN), and false positives (FP) resulting from comparing the predicted object to all ground truth objects, which can be further explained in the following equations. The Equation 2 is equivalent to F1 Score when β is set to 1.

$$IoU(A, B) = \frac{A \cap B}{A \cup B}. \tag{1}$$

$$F_\beta(t) = \frac{(1+\beta^2) \cdot TP(t)}{(1+\beta^2) \cdot TP(t) + \beta^2 \cdot FN(t) + FP(t)}. \tag{2}$$

4 Dataset

One RGB-D dataset that is composed of video sequences from a variety of indoor scenes, NYU Depth V2 [1], was used in our project. These videos were recoded by RGB and Depth cameras from the Microsoft Kinect [6]. This dataset is consists of 1449 RGB-D images of size 480 (height) $\times$ 640 (width) and all images are densely labeled at pixel level, spanning totally 894 different classes. In addition, the depth information in this dataset has been preprocessed to fill in any missing depth data using colorization scheme of Levin et al [7]. RGB images in this dataset (same images without depth information) were used as baseline metrics. A sample figure is shown in Figure 2.

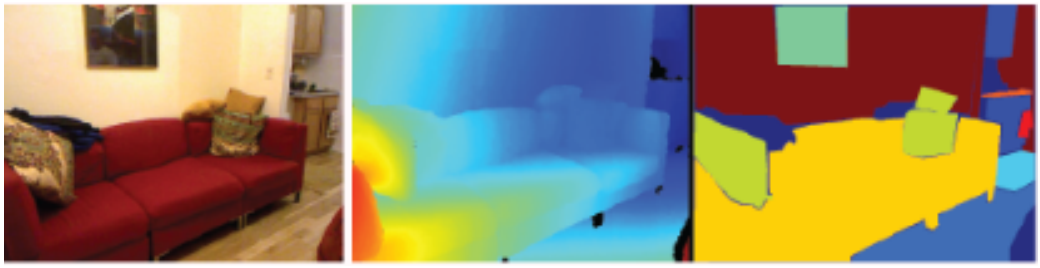

Figure 2: Sample RGB image, depth image, labeled image from the dataset

5 Preliminary Results

As for our current results, we have implemented a baseline model on RGB images as well as a preliminary trial on training with RGBD images by simply concatenate the input channels.

5.1 Experiment settings

For both RGB and RGBD cases, since the image size and the number of classes are comparatively large, 480 (height) $\times$ 640 (width) with nearly 900 classes, the training process is extremely GPU memory consuming (for AWS P2 xlarge instance). Thus, we set the batch size of 1 with a SGD optimizer start with learning rate 0.001. The loss function we adopted here is pixel wise Cross Entropy loss and no fancy image preprocessing methods are incorporated. The baseline model takes the average value of RGB 3 channels for each pixel and becomes 1 channel image as input while for RGBD U-Net, we simply concatenate the depth channel as the 2 channel image input. For validation, we use a train test split with a ratio of $4:1$, which is equivalent to say that 20% of the images are in the holdout set for validation.

5.2 Baseline model: Unet on RGB

After trained for 50 epoches on all training images, we obtained a model which has around 40% pixel wise accuracy on the test set. The learning curve for the first 30 epoches is shown in the figure 3

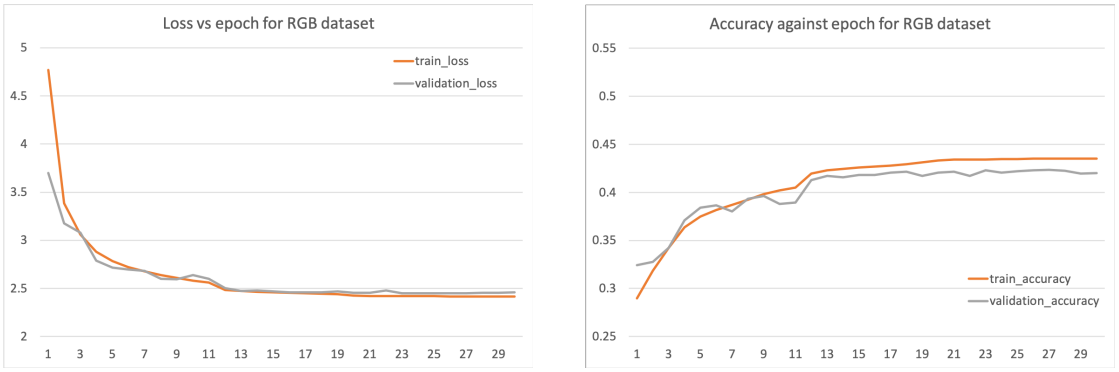

Figure 3: Training loss and accuracy on Unet baseline model on RGB images

5.3 Test model: Unet on RGB-D

Using the same setting as baseline, after trained 50 epoches on RGBD dataset, we obtained a model which has around 45% pixel wise accuracy on the test set, which is apparently higher than the baseline model (40%). Similarly, the

[1] `https://cs.nyu.edu/~silberman/datasets/`

learning curve for the first 30 epoches is shown in the figure 4. Since in the original dataset, the proportion of pixels for object classes (not the background, which is labeled as 0) is small (less than 50%) in most of images, we want to reject the trivial case that the model simply predicts 0 for all the pixels. Therefore we plot the predicted labels by comparing the ground truth labels in figure 5. It is clearly that the model indeed captures a clear boundary of all different objects.

Figure 4: Training loss and accuracy on Unet model on RGB-D images

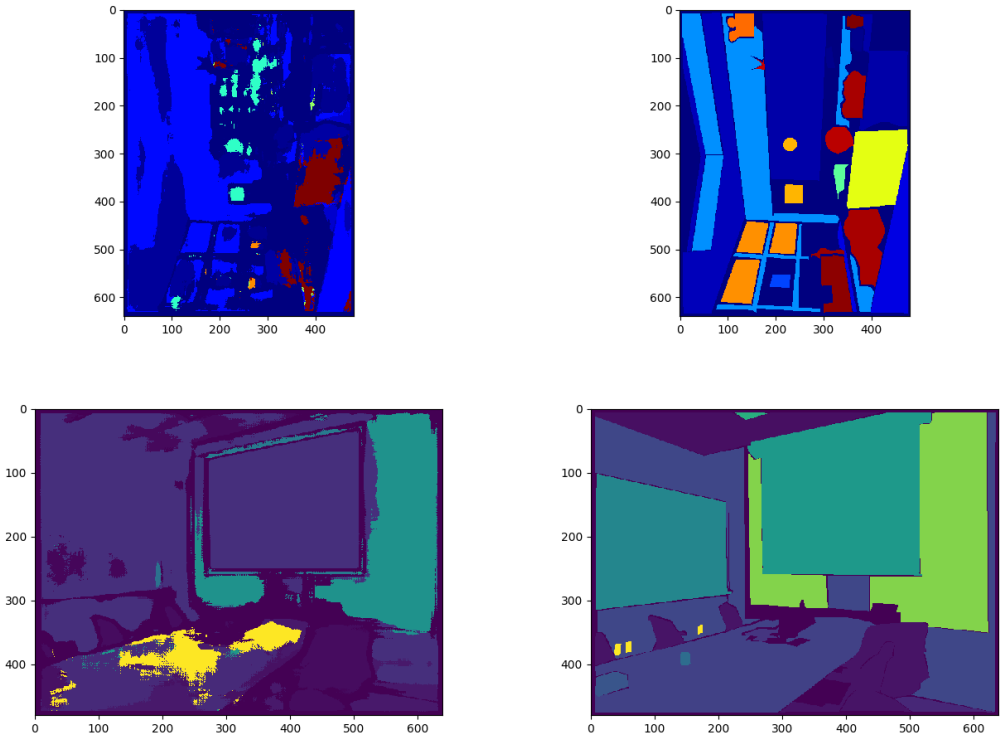

Figure 5: Model prediction (left) and ground truth (right) on Unet on RGB-D images, the actual colors may not correspond exactly since the color map will automatically assign colors to the normalized image

6 Conclusion

The baseline model has reached 40% of pixel wise accuracy, while by adding depth channel, the performance can be slightly improved to 50%. What is more, the prediction in RGB-D can accurately capture the edges of objects, which has compensated the intrinsic shortcoming for RGB image segmentation. However, the overall performance of the model is far from enough and some potential reasons can be interpreted as following:

1. Too many classes with too few training examples: The entire dataset contains around 900 classes with an extremely imbalanced distribution: some of the classes have much more examples than the rest of others. As a result, images for some classes are insufficient for training such a deep convolutional neural network.

2. Illnesses in experiment settings: We used SGD with a batch size 1, which is insufficient for some manipulations such as batch normalization.
3. No image preprocessing: Since some images are much brighter than the rest of others, the scale of input may not be in the same scale.

Base on the analysis above, we are going to do several experiments to solve each of the problem. The lack of training examples can be addressed by erasing some images containing rarely appeared classes or try some data augmentation approaches to boost the dataset. The illness in experiment can be addressed by transplanting the training process to a more powerful GPU instance. Also, we will test different image preprocessing operations such as enhancing the contrast and other color based manipulation instead of simply taking the average of RGB.

7 Timeline

By the mid-term milestone, we have accomplished the following tasks:

1. Implemented the baseline U-Net method and analyzed the performance.
2. Use the same model on RGB-D dataset to verify the hypothesis that RGB-D image works better in this task.

By the end of the semester, we plan to:

1. Implement different models with the RGB-D dataset such as Mask-RCNN.
2. Try different image preprocessing methods in order to better incorporate depth information.
3. Postprocess the original image into depth-of-field effect(clear foreground with blurring background) using the segmentation result and depth information.

References

[1] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. "u-net: Convolutional networks for biomedical image segmentation". In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[2] Jonathan Long, Evan Shelhamer, and Trevor Darrell. "fully convolutional networks for semantic segmentation". *CoRR*, abs/1411.4038, 2014.

[3] Alex Kendall Badrinarayanan, Vijay and Roberto Cipolla. "segnet: A deep convolutional encoder-decoder architecture for image segmentation". 2015.

[4] Liang-Chieh Chen. "semantic image segmentation with deep convolutional nets and fully connected crfs". 2014.

[5] S. Hong H. Noh and B. Han. "learning deconvolution network for semantic segmentation". In *ICCV*, pages 1520–1528, 2015.

[6] Pushmeet Kohli Nathan Silberman, Derek Hoiem and Rob Fergus. "indoor segmentation and support inference from rgbd images". In *ECCV*, 2012.

[7] Anat Levin, Dani Lischinski, and Yair Weiss. "colorization using optimization". In *ACM transactions on graphics (tog)*, volume 23, pages 689–694. ACM, 2004.