Exploring Semantic Segmentation Problem in Urban Driving Datasets

1st Komel Merchant *Electrical Engineering*
Johns Hopkins University
Baltimore, Maryland
kmercha2@jh.edu

***Abstract*—This paper explores a binary semantic segmentation problem in the context of urban driving scenes – specifically the problem of car segmentation. In this paper, I explore 2 issue within the space. (1) Is the problem is of dealing with unbalanced datasets. Specifically, I look at the DICE loss and compare performance to the conventional binary cross-entropy loss. (2) Is the question of network scalability with respect to location. I train a U-Net on semantic data from the KITTI [3] dataset. I then evaluate performance qualitatively on a custom dataset generated on DC street and acquired from a standard iPhone camera.**

***Index Terms*—component, formatting, style, styling, insert**

I. Introduction

The area of urban driving is a really important one especially with the rise of autonomous mobility companies[citation here]. Smart cities are also becoming more of a reality in a lot of places. Being able to track vehicle positions over time is crucial for task within robotic perception, traffic estimation and many other technical innovations in urban situations. Vehicle segmentation sits high up in priority within this problem space. In this paper, I explore two main issue within the car semantic segmentation problem.

A. Semantic Segmentation for Urban Driving

u

B. Resolving Class Imbalance

The first is working with unbalanced datasets. In the context of urban scenes, the semantic label space is pretty large. Segmentation algorithms have to interact with roads, poles, signs, cars, sometimes horse-drawn carriage [4]. While cars are a common occurance in images, they take up a smaller proportion of samples in the context of binary segmentation. Furthermore, their frequency is largly governed by other external factors such as time of day, weather conditions, etc. Continuous obstacles such as road ways and such are far less challenging to detect and segment due to their consitency in most urban environments. In the Kitti dataset, the dataset I used for training and evaluation, car's take up approximately 5 percent of the data. If we don't account for this in training our model, our model will learn to produce segmentation masks which produce good general accuracy metrics, but suboptimal performace on the class of interest. An example of this would be the situation where the network might predict a segmentation mask of "not-car". In this case, the accuracy might be high, but the class-specific accuracy will be poor (ie. failure to predit the object of interest: the pixels assoiciated to car). This problem has been a topic of concern in the medical imaging space. In this context, the segmentation masks can potentially be small and localized to different portions of a sample. In the context of tumor segmentation for example, tumors can have varying shapes, sizes and locations within a 3D volume. Furthermore, tumors generally take up a smaller portion of the brain than unaffected brain matter. As such, many researchers in this space have attempted to address these issues, by designing loss functions which weight positive examples more than the negative ones.Milletari [1] propose using a DICE loss, which is similar to Insersection over Union (IoU) loss. For segmentation problems in the multi-label space domain, researchers can weigh certain labels more or less depending on their frequency. Unet **CITATION** does this by using a Wighted Cross Entropy for their training loss.

II. Methods

A. Architecture

For this project, I decided to use the U-Net model. This model was initially used in the context of medical imaging for the use of cell segmentation. The original model features an encoder-decoder pair. The encoder is comprised of a series of blocks which consist of 2 3x3 Convolutions, followed by a standard 2x2 max-pooling layer. There are 4 of these such blocks leading up to the bottleneck. For the decoder portion, there's a series of Up-Convolution blocks that correspond to each block in the encoder. These Up-Convolution blocks consist of 2 3x3 Convolutions, followed by an transposed convolution (also known as a fractional kernel) used to up scale the feature maps. The inputs to these "Up-Conv" blocks are a concatenation of 1) the preceding transposed convolution and 2) a center-crop from the corresponding block in the encoder pair. This cat-crop mirror is what forms the "U" shape in U-Net. Figure 1 shows the original architecture of the model.

I made a few modifications to help with training and testing. The input image for the standard U-Net was 572x572 and output image was 388x388. The authors then used morphological operations to resize the segmentation mask to it's original space. To speed up training time, I resized my input image

to 256x256. Furthermore, I padded my feature maps in the output of all the convolution steps to avoid having to rely on image processing methods to get back to the original size. This simplified validation and loss computations tremendously.

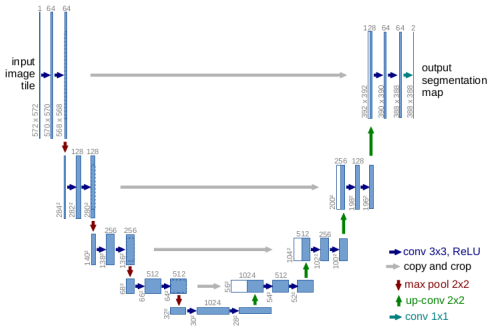

Fig. 1. Figure of the original U-Net.

B. Loss Functions

To address the binary segmentation problem, I experimented with 3 different loss functions: Binary Cross Entropy (BCE), DICE, and a modified version of DICE where I dropped the negative example term. Sudre, C.H., et al. [2] refer to these losses. Let r_i define a ground truth pixel in the semantic map. Let p_i represent a pixel in the predicted semantic map. I define the BCE loss as:

$$-\frac{1}{N}\sum_{N} r_i log(p_n) + (1 - r_n) log(1 - p_n) \tag{1}$$

The DICE loss as:

$$1 - \frac{\sum_{i=0}^{N} p_n r_n + \epsilon}{\sum_{i=0}^{N} p_n + r_n + \epsilon} - \frac{\sum_{i=0}^{N} (1 - p_n)(1 - r_n)}{\sum_{i=0}^{N} 2 - p_n - r_n + \epsilon} \tag{2}$$

And the ”modified” DICE loss

$$1 - \frac{\sum_{i=0}^{N} p_n r_n + \epsilon}{\sum_{i=0}^{N} p_n + r_n + \epsilon} \tag{3}$$

The last function was one that came up with. Due to the severe imbalance in the dataset (which will be covered in the Datasets section), I wanted to see would happen if we got rid of the term associated with the background class. In any case, BCE in theory is less suited for imbalanced class problems compared to the overlap-based losses. This comes from the denominator of the loss. While BCE focuses on raw numbers of positive and negative samples, the DICE losses normalize their errors based on the frequency of the classes.

C. Datasets

For this paper, I used two dataset. The first was the Kitti [3] dataset. This comprised of 200 images along with semantic maps. The semantic label space consist of 33 different classes. These classes consist of items such as cars, poles, road, etc. For this assignment, I decided to focus strictly on cars detections. One difficult of working with this dataset was that each image had a slightly different size. The dimensions were roughly 1200x375. To normalize the image size, I resized each image and corresponding segmentation mask to a 265x265 patch. Learning and inferencing on images of this size was useful. I was able to achieve good training speed all while maintaining decent performance. I applied a standard 80-10-10 split on the 200 images to generate samples. After converting the multi-class dataset into a binary segmentation one, I computed the proportion of ”car” pixels to ”non-car” pixels as 0.067. This low value clearly points to a high degree of binary class imbalance.

D. Training

For training, I used a standard SGD optimizer with a momentum of 0.9 with an initial learning rate of 0.001. I also added a learning scheduler that’s set to decrease the learning rate by a factor of 10 every time the validation loss remains unchanged for 10 epochs. I set the model to train for 300 epochs. This was more to bound the computation to a tractable amount, but – as we’ll see in the experiments section – the models’ converged fairly reasonable minima within this time.

III. Experiments

A. Training on KITTI Semanitic Segmentation Dataset

In general, all the models seems to converge reasonably well as seen in Figues 2, 3 and 4. I noticed that there are some ”jagged portions” in the validation loss around the 150-200 epoch range that leads into a plateu. This plateu seemed to occur directly after the loss was lowered by the scheduler. This leads me to believe that it potentially had to do with our optimizer continually missing the local minima (ie. underfitting due too large of a learning rate).

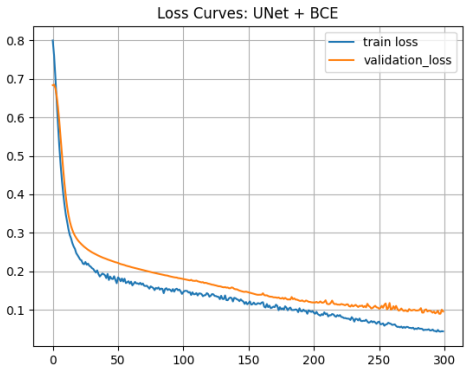

Fig. 2. Training and Validation Loss for BCE loss on KITTI dataset.

B. Evaluation on KITTI Dataset

Overall the model performed well on the KITTI test dataset. Using the AUC measure (See in Figure 5) of the PR curve, the ranking seems to favor DICE original, followed by DICE ”Modified” and then BCE. This makes sense as DICE has proven to be a better loss to deal with class imbalance. The

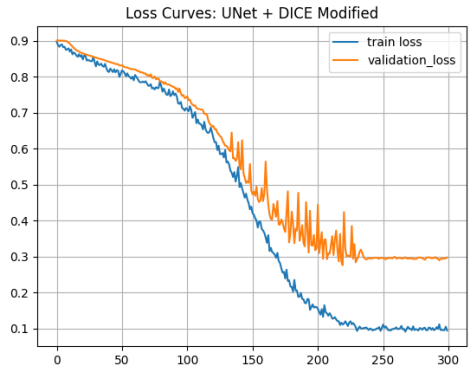

Fig. 3. Training and Validation Loss for DICE ”modified” loss on KITTI dataset.

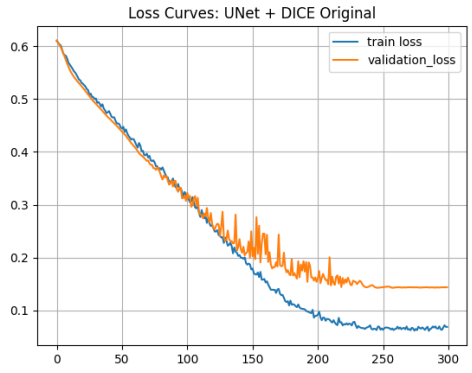

Fig. 4. Training and Validation Loss for DICE loss on KITTI dataset.

delta between BCE and DICE ”Mod” is much larger than the distance between DICE ”Mod” and DICE ”Original”, indiciating that the overlap based approach work better than the raw-frequency-based loss approach.

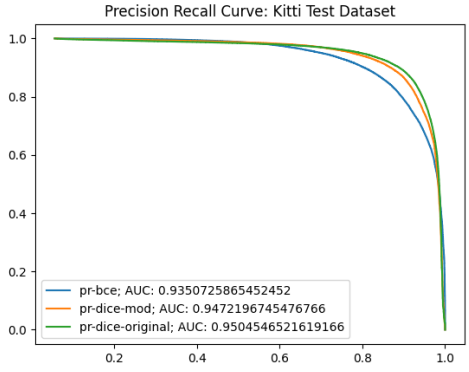

Fig. 5. PR Curve Comparison KITTI Test Dataset.

C. Evaluation on Custom ”DC Commute” Dataset

IV. Discussion

These experiments have shown a few important concepts about training segmentation models. 1) Designing a loss function that works with your data is extremely important. If your data is severly imbalanced, choosing an overlap-based loss function is a viable solution to working with this deficinecy. 2) It is important to training your model with a sufficient amount of data. Based on the results of a KITTI-training U-Net on a custom dataset, your training data must capture various different aspects inorder to be properly scalable. This includes lighting conditions, type of camera, resolution of camera, etc.

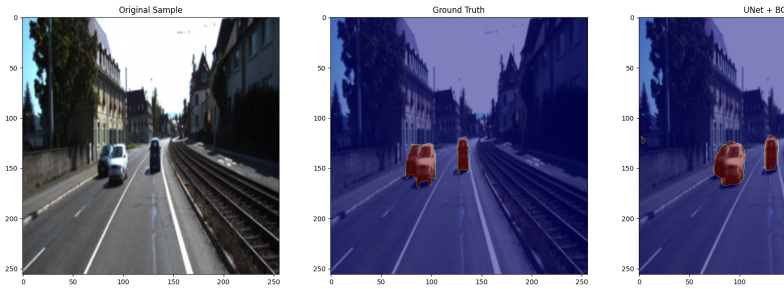

Fig. 6. Qualitative Sample 1 KITTI Test Dataset.

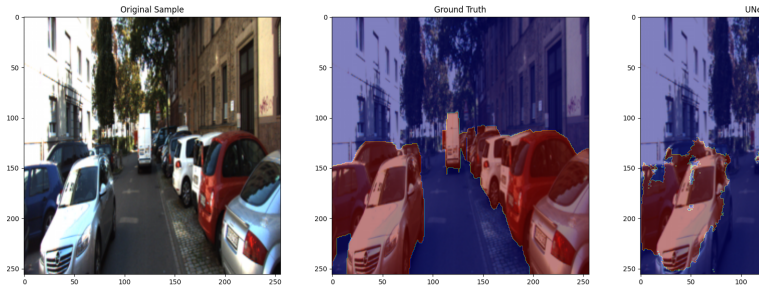

Fig. 7. Qualitative Sample 2 KITTI Test Dataset.

Acknowledgment

N/A

References

[1] Brosch, T., Yoo, Y., Tang, L.Y., Li, D.K., Traboulsee, A., Tam, R.: Deep convo- lutional encoder networks for multiple sclerosis lesion segmentation. In: MICCAI 2015. pp. 3–11. Springer (2015) Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are weready for autonomous driving? the kitti vision benchmarksuite. In2012 IEEE conference on computer vision and pat-tern recognition, pages 3354–3361. IEEE, 2012.

[2] Sudre, C.H., Li, W., Vercauteren, T., Ourselin, S., Jorge Cardoso, M.: GeneralisedDice overlap as a deep learning loss function for highly unbalanced segmentations.Lecture Notes in Computer Science (including subseries Lecture Notes in ArtificialIntelligence and Lecture Notes in Bioinformatics)10553 LNCS, 240–248 (2017).

[3] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are weready for autonomous driving? the kitti vision benchmarksuite. In2012 IEEE conference on computer vision and pat-tern recognition, pages 3354–3361. IEEE, 2012.

[4] https://www.reddit.com/r/technology/comments/wt89lv/video$_a ppears_t o_s how_a t eslas_a utopilot_s ystem/$