Semantic Segmentation for Autonomous Driving Applications

Gouthamaan Manimaran
M.Tech., School of Engineering,
Amrita Vishwa Vidyapeetham,
Amritapuri Campus, Kollam, Kerala, India
goutham123manimaran@gmail.com

S.Swathi
M.Tech., School of Engineering,
Amrita Vishwa Vidyapeetham,
Amritapuri Campus, Kollam, Kerala, India
swathisampath18@gmail.com

***Abstract*— Image Segmentation is a critical task in many autonomous robot workflows including self driving cars. The agent must be able to not only percieve but also understand its environment. A computer cannot understand an image like humans do and cannot differentiate between a pedestrain or a traffic signal. With Image Classification, It can differentiate between the two but would not know where it is located. Object detection solves this problem by localising based on a bounding box coordinates. Still, for a task like autnomous driving, rough numbers would prove to be dangerous. This is where semantic segmentation comes in. With this method, A computer can know where exactly an object or multiple objects are in a frame and the accuracy comes down to a tiny pixel. This paper focuses on the segmentation application of driverless cars with four different architectures to provide a comparative study using the IoU and Dice score. The architectures discussed here are Fully Connected Networks (FCN), SegNet, UNet and Pix2Pix using the famous cityscapes dataset. This consists of about 30 different classes so that a robot can make informed decisions based upon its environment.**

Keywords*— *Image Segmentation, cityscapes, FCN, UNet, Pix2Pix, SegNet, IoU, Dice

I. INTRODUCTION

Autonomous driving is one of most convoluted tasks where following certain rules is not enough, there are huge number of factors like weather conditions, unexpected situations, split second decisions to avoid endangering human life. In the age of artificial intelligence, many companies are investing a lot to of money for the research and development of hardware and software to support this function. There are many components to self-driving cars like LIDAR, cameras, sensors supporting the algorithms that are running behind to provide a unified output.

One of the most important sensors is the visual sensor and the self-driving cars depend largely on the well-functioning image recognition task. The complex system is based on many components of which a major one is the semantic segmentation [1]. Semantic segmentation is a process where the objects of the image are identified, and boundaries of the object are set according to the labels. It mainly clusters the more similar pixels and assigns the pixels to a predefined class. The major difficulty is the data gathering where huge amount of data is required to train the model.

In our study, four different architectures of semantic segmentations are applied for comparative study of the performance and accuracy in the application of autonomous cars. The two performance measures considered are the IoU and Dice score.

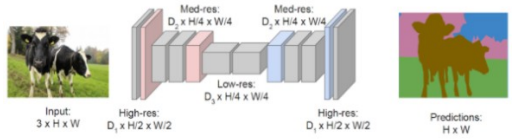
II. ARCHITECTURES

Implementing Semantic Segmentation:
Naive Semantic Segmentation can be done with a slight modification to the Classification example.
Instead of the final layers where we apply softmax activation over the number of classes, we can predict the value of every pixel of an image and assign it to a specific category.
Although this type of architecture seems like we can approach segmentation tasks, there is a huge constraint on the number of parameters as the final Dense layer must contain N number of pixels and the advantage of feature sharing in Convolutions is not used.

A. Fully Convolutional Networks (FCN) [2]

One Approach to counter the problem of too many parameters as the model above is to have convolutional layers with same padding (to preserve dimensions) and output a final segmentation mask. To further improve this model and reduce parameters, We can introduce Down sampling and Up sampling. In the first half of the model, we down sample the spatial resolution of the image developing complex feature mappings. We obtain finer information of the image but the location is lost. To recover from this, Up sampling is done which takes multiple low-resolution images and outputs a high-resolution segmentation map as output. Skip Connections

are added across the down sampling and up sampling layers to preserve spatial information.



There are different variants of FCN [3] depending on the level of up sampling required.
FCN-32 upsamples wit stride set to 32 and this does not have any skip connections.
FCN-16 are bit more precise with skip connections added and has stride 16.
FCN-8 have stride 8, with even more precision compared to the above two.

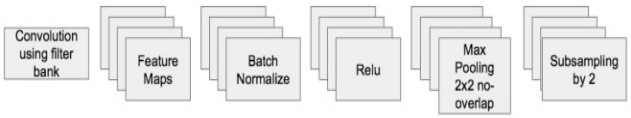
B. SegNet

SegNet [4] is designed to be a pixel wise segmentation model which is used to study the spatial relationship of the various objects (classes) in the image, where the various classes are clustered together to provide a smooth segmentation.
SegNet has an encoder-decoder type architecture where the encoder has the pre-trained VGG-16 model for down sampling the image in study. The 13 convolution layers in the VGG-16 are used and for each of encoder layer there is a decoder to up sample the pixels in the image to its original size, thus the SegNet type architecture preserves the original dimensions of the image. This architecture consists of a final pixel wise classification layer to provide the class wise probabilities of the independent pixels from the decoder. The encoder retains the higher resolution feature maps from the encoder while decoder which up sample these maps. The final output from the decoder is fed into the multi-class softmax classifier.

Encoder:
The encoder is used to produce feature maps by applying convolutions with a filter bank. The following is performed to produce the feature maps:

1. Convolutions are performed with a filter block.
2. The outputs are batch normalized.
3. An element wise application of rectified nonlinear activation (ReLU) activation is applied.
4. A max-pooling with a 2x2 window and a stride of 2 is performed.
5. A sub sampling by a factor of 2.

The sub sampling results in a large input image context for each image in the feature maps.



The subsequent layers of max pooling and sub sampling leads to a more invariant translations but a loss in spatial resolution in the feature maps. The loss(boundary detail) is mitigated by storing the boundary information in the encoder feature maps before the sub-sampling is performed, in a more practical sense the maxpool indices, i.e. the location of the maximum feature values in the pooling window are stored in the feature maps to reduce the storage space.

Decoder:
The up sampling of the input feature map from the encoder using the stored feature map with the max pooling indices which produce sparse feature maps. The decoding follows the following steps:

1. The feature maps are convoluted with a filter bank to produce dense feature maps.
2. Batch normalization is introduced next to the feature maps. It produces feature maps with same number and size of inputs as the encoder inputs.

The high dimensional feature representation is the output of the decoder.

Classifier:
The high dimensional feature representation is applied to a multiclass softmax classifier which classifies each of the pixels independently. The output has k-channel image probabilities where the k is the classes.
The main advantage of using SegNet is the low memory requirement during both the training and validation and the model is much smaller compared to FCN or DeconvNet, but it is slower due to its decoder architecture.

On the application of autonomous vehicle by using the SegNet application, the self-driving cars requires decision making in the order of milliseconds, the present case SegNet provides less memory usage, but the response time [5] is slow for autonomous vehicles.
The SegNet has much fewer trainable parameters since the decoder layers use max-pooling indices from corresponding encoder layers to perform sparse up-sampling. This reduces inference time at the decoder stage since, unlike FCNs, the encoder maps are not involved in the up-sampling. Such a technique also eliminates the need to learn parameters for up-sampling, unlike in FCNs.
It produces a dice and IoU score of and thus it not much suitable for self-driving car application.

C. UNet

The UNet was developed for Bio Medical Image Segmentation. It focuses on image classification where the presence is not alone identified but the location of the object presence must also be identified by doing classification on all the pixels.

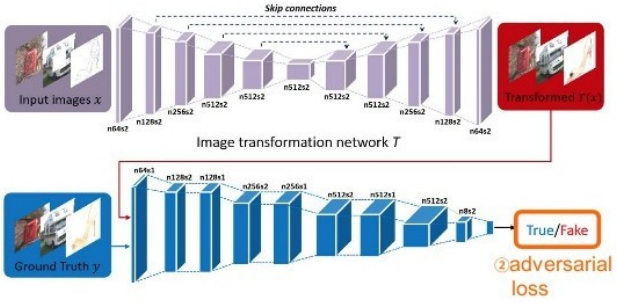
The UNet architecture has a “U” shape which is symmetric and consists of 2 paths. First, the contracting path used to capture the context of the image. Second, expansion path used to enable precise localization using transposed convolutions. The size of the image is not affected since it only has convolution layers and not any dense layers.

Contracting path:
The contracting path follows the formula:
conv_layer1 > conv_layer2 > max_pooling > dropout (optional)
Step 1: Each process has two convolutional layers; the number of channels changes from 1/3 to 64 and the depth of the image increases. By applying max pooling, it halves the size of the image.
Step 2: The above process is repeated thrice.
Step 3: And finally, 2 convolution layers are built with no max pooling layer
After the final step it has a feature maps with the context of the image.

Expansive path:
In the expansive path the image is upsized to its original size and follows the formula:
conv_2d_transpose > concatenate > conv_layer1 > conv_layer2
Step 1: Transpose convolutions is the up-sampling technique that expands the size of the image by providing some padding followed by concatenating layers from the down sampling layer. This is done to combine the information from the previous layers to provide a precise prediction. After this, convolution operations are performed.
Step 2: The above process is repeated three times.
Step 3: This step is to reshape the image to satisfy the label and requirements to be predicted.
The last layer has a convolutional layer with a filter of 1x1.
The contraction-expansion network with skip connections between them to provide knowledge from the earlier phase to the later
UNet is able to do image localization by using pixel by pixel prediction and many applications is carried out using UNet architecture which was designed mainly for medical image classification/segmentation.

D. Pix2Pix

This architecture is like that of a Generative Adversarial Network (GAN) and can be used for many image-to-image translation tasks like generating pencil outlines of an image, segmentation,
Image Inpainting, etc. As all these outputs are conditional on the input and not random, Pix2Pix Falls under the category of Conditional GANs or CGANs [6].
As all GANs, we have a generator and a discriminator model for training. What makes this powerful is that in the Generator side, we have a full-fledged UNet model. The UNet architecture is explained in the above section, and we can see how powerful it already is. Training this alongside the discriminator does wonders to Image-to-Image translation tasks.



The generator is not only tasked to fool the discriminator but also to be near the ground truth output in an L2 sense. But L1 Distance is used rather than L2 as L1 encourages less blurring. Thus, this too is added to the objective loss function during training.

III. Experiment and Result

A. Dataset

Cityscapes dataset consists of urban scenes for the semantic understanding. It provides a dense pixel wise annotation of 30 classes which are grouped in 8 categories of flat surfaces, humans, vehicles, constructions, objects, nature, sky, and void. It has 5000 frames of high-quality pixel annotations and 20000 low level pixel annotations. The data was captured in 50 countries in various conditions of weather and situations (day and night times) across various months. It is firstly recorded as a video, thus having various features like large number of dynamic objects, varying scene layout, and varying backgrounds.

B. Comparison Metric

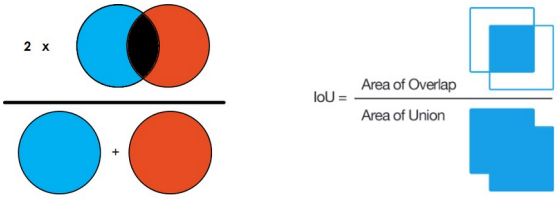
To compare the given four architectures on the cityscapes dataset, we use two different comparison metrics – IoU and Dice score. A brief on them is given below:

1. Intersection Over Union (IoU)

IoU [7] is the area of overlap between the predicted segmentation and the ground truth divided by the area of union between the predicted segmentation and the ground truth. For binary (two classes) or multi-class segmentation, the mean IoU of the image is calculated by taking the IoU of each class and averaging them.

2. Dice Score

Simply put, the Dice [8] Coefficient is 2 * the Area of Overlap divided by the total number of pixels in both images.



C. Comparison Results

We have used the same training and testing set on all four architectures and these are the results we found.

	IoU	Dice
FCN	0.353	0.480
SegNet	0.299	0.459
UNet	0.283	0.441
Pix2Pix	0.412	0.513

*All scores range from 0 to 1 with 1 being a perfect score.

IV. Conclusion

As we can see, Pix2Pix is clearly the superior model and also the current benchmark in Image to Image translation tasks outperforming all the other models.
There are more models built on top of Pix2Pix which get better scores on certain datasets once they are finetuned to it [9]. UNet outperforms Pix2Pix in medical Image Segmentation tasks [10], thus no model is best across different types of data.
But for Autonomous Driving application with a wide variety of classes, Pix2Pix has an edge compared to the rest of the architectures.
Working on this further, This model can be combined with a depth perception model or a LIDAR to have complete knowledge of its surroundings.

V. References

[1] Audebert, N., Le Saux, B., & Lefèvre, S. (2017). Segment-before-detect: Vehicle detection and classification through semantic segmentation of aerial images. *Remote Sensing, 9*, 368.

[2] Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. *Proceedings of the IEEE conference on computer vision and pattern recognition*, (pp. 3431–3440).

[3] Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., & Yuille, A. L. (2014). Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv preprint arXiv:1412.7062*.

[4] Badrinarayanan, V., Kendall, A., & Cipolla, R. (2017). Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence, 39*, 2481–2495.

[5] Siam, M., Gamal, M., Abdel-Razek, M., Yogamani, S., Jagersand, M., & Zhang, H. (2018). A comparative study of real-time semantic segmentation for autonomous driving. *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, (pp. 587–597).

[6] Kniaz, V. V. (2018). Conditional GANs for semantic segmentation of multispectral satellite images. *Image and Signal Processing for Remote Sensing XXIV, 10789*, p. 107890R.

[7] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid and S. Savarese, "Generalized Intersection Over Union: A Metric and a Loss for Bounding Box Regression,"*2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 658-666, doi: 10.1109/CVPR.2019.00075.

[8] T. Eelbode et al., "Optimization for Medical Image Segmentation: Theory and Practice When Evaluating With Dice Score or Jaccard Index," in IEEE Transactions on Medical Imaging, vol. 39, no. 11, pp. 3679-3690, Nov. 2020, doi: 10.1109/TMI.2020.3002417.

[9] Wang, X., Yan, H., Huo, C., Yu, J., & Pant, C. (2018). Enhancing Pix2Pix for remote sensing image classification. *2018 24th International Conference on Pattern Recognition (ICPR)*, (pp. 2332–2336).

[10] Weng, Y., Zhou, T., Li, Y., & Qiu, X. (2019). NAS-Unet: Neural architecture search for medical image segmentation. *IEEE Access, 7*, 44247–44257.