2022 2nd International Conference on Intelligent Technologies (CONIT)
Karnataka, India. June 24-26, 2022

Image Segmentation for Self-Driving Car

Sanchit Gautam
Department of Software Engineering
Delhi Technological University
Delhi, India
sanchitgautam_2k18se112@dtu.ac. in

Tarosh Mathuria
Department of Software Engineering
Delhi Technological University
Delhi, India
taroshmathuria_2k18se128@dtu.ac.in

Shweta Meena
Department of Software Engineering
Delhi Technological University
Delhi, India
shwetameena@dtu.ac.in

Abstract—**Machine learning plays a vital role in making a car autonomous. The machine learning algorithms interpret the surrounding data collected from the car's sensors and camera and perform desired actions. Semantic segmentation is one of the various techniques which can be used in self-driving cars. High computational costs will be needed to obtain higher accuracy in semantic segmentation, which is not suited for embedded systems. So, there is a need to get high accuracy models with low computational requirements. This paper used a U-Net model for the semantic segmentation task, which shows good accuracy. We will be using the cityscape image pair dataset to train and test our model. We will evaluate our model based on the Receiver Operating Characteristic (ROC) curve and Intersection over Union (IOU).**

Keywords—U-Net, image segmentation, K-Means

I. INTRODUCTION

In this era where every family owns a vehicle, there's a great demand for vehicles. The need for cars is constantly increasing, which also leads to heavy traffic and car accidents. Car companies like Tesla, Morris Garages (Mg), Audi, Hyundai, etc., are launching cars with different autonomous levels. This autonomous level starts from 0, with no assistance system, and goes up to level 5, which describes fully autonomous driving. The first step in most autonomous vehicle systems is object recognition or semantic segmentation.

Image segmentation is the partitioning of an image (car dashboard image, surveillance camera image.) into segments or sets of pixels to identify and locate different objects or entities in the image. Each pixel is assigned a label so that pixels with the same label share some common characteristics like color or texture. Image segmentation can be divided into two groups:

Semantic Segmentation: It is an approach in which a class of the object is detected for every pixel. Every pixel is grouped class-wise. For example, in a street view, it will label streets, sky, vehicles, humans, trees as different classes.

Instance Segmentation: It is an approach in which an instance of the object is detected for every pixel. It detects all distinct objects. For example, in a street view, it will label all humans as different objects.

Image segmentation assigns a label to every pixel of the image, whereas in image classification, a discrete value is assigned to the entire image. Image segmentation finds applications in various fields like self-driving cars, video surveillance, medical imaging, and many robotics-related tasks. A major problem in this task is the tradeoff between accuracy and computational cost. Since it is a real-world deployment task, it needs to be both accurate and fast.

This study is divided into different sections: Section II summarizes related work in the area of semantic segmentation. Section III discusses research methodologies used for the study. Section IV includes our proposed system of semantic segmentation for self-driving cars. Section V discusses the results shown by the model. This study finally concludes in Section VI.

II. RELATED WORK

This section will discuss the related work done in the image segmentation field. We will summarize various studies done recently in this field.

Semantic segmentation has seen some progress in recent years with the advance in deep learning algorithms such as convolutional neural networks (CNN). The first work proposed in this field was Fully Convolutional Networks (FCNs) [1], a method to classify each pixel using transposed convolution for up sampling.

Conditional Random Field (CRF) [2-3] came, which refined the output and increased the accuracy, but with increased computational cost.

M. Treml, J. Medina et al. [4] introduced an architecture consisting of ELU activation functions in 2016, an encoder, followed by convolutions, and a decoder with refinement modules. They have used the Cityscapes dataset. The architecture achieved 59.8 per-class mean IoU and 84.3 per-category mean IoU. Hence, the network architecture outperforms both E-Net as well as Seg-Net.

A. Sagar and R. Soundrapandiyan [5] used Cityscapes and Camvid dataset and used a ResNet-based feature extractor in their architecture. A new attention module was recommended in their report to encode more contextual information and improve the network's receptive field.

M. Siam and M. Gamal [6] examined various segmentation methods and introduced a real-time segmentation benchmarking platform in 2018. They have used the Cityscapes dataset. They proposed a study on generic meta-architecture based on a decoupled design that allows different types of encoders and decoders to be plugged in separately.

M. Yang, K. Yu, and C. Zhang [7] introduced DenseASPP in 2018. DenseASPP was introduced in the paper to address the difficult problem of street scene segmentation, where objects differ mostly in scale. The effectiveness of DenseASPP is demonstrated through theoretical analysis, visualisation, and quantitative experimental findings on the Cityscapes dataset.

D. Feng, C. Haase-Schuetz et al. [8] have presented a survey for deep multi-modal object detection and

2022 International Conference on Intelligent Technologies (CONIT) | 978-1-6654-8407-7/22/$31.00 ©2022 IEEE | DOI: 10.1109/CONIT55038.2022.9847974

978-1-6654-8407-7/22/$31.00 ©2022 IEEE

segmentation applied to autonomous driving in 2019. They have discussed both multi-modal datasets and fusion methodologies, considering “when to fuse,” “how to fuse,” and “what to fuse.”.

B. Chen, C. Gong et al. [9] claimed that different classes have different levels of value for safe driving. They used the datasets: Cityscapes and Camvid. and developed Importance-Aware Loss (IAL), forward and backward propagation rules and applied them to deep neural networks.

III. Research Methodology

In this section, we will discuss methodologies involved in image segmentation. Image Segmentation is the partitioning of an image into multiple regions and then extracting the specified region/area, also known as the Region of Interest (ROI). If we want to segregate a flower's region or the car's region from that image, then that region is our ROI.

It is not feasible to process the entire image altogether; thus, we divide the image into different segments. A picture is the combination of pixels, or we can say pixels are the building blocks for a photo. Each pixel has some attributes. Pixels in an image can either be the same or can differ from other pixels depending on the features. Pixels having similar attributes will be masked with the same color, and pixels with different characteristics will have a different color. For example, if we consider the pixels of the road from the image, then all the pixels in the road will have the same attribute, and we will mask it with a single color. Similarly, we will mask the pixels of the sky with the same color. After color clustering, the road will look pink, and the sky will look blue in our images.

A. *Semantic Segmentation*

In Semantic Segmentation, every pixel belongs to some class. Similar pixels belong to the same class with the same color. In other words, we can say that semantic segmentation is giving a class to every pixel in an image. To assign the class to every pixel, we can use different segmentation algorithms like K-means clustering, Mask-RCNN, edge detection segmentation, region-based segmentation, etc.

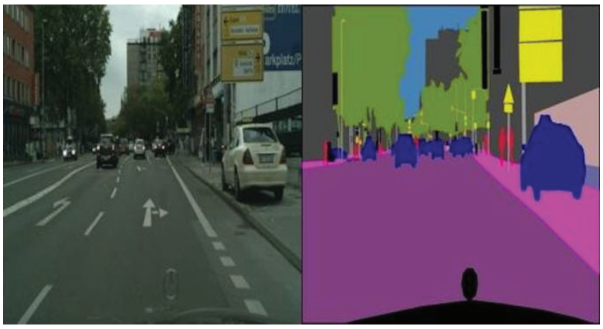

Fig. 1. Original image on the left and segmented image on the right

B. *K-Means Clustering*

Clustering algorithms [10] are unsupervised learning algorithms. Clustering divides the data points into some groups or clusters. Data points in the same groups are similar to each other and data points in different groups are different from each other and these groups are known as clusters.

K-Means clustering is a commonly used technique for color clustering. The character K in K-means represents the total number of clusters. Steps followed in clustering are as follows:

In step 1, depending on the number of clusters required, we decide the value of K. In step 2, we assign each data point randomly to any of the K clusters. In step 3, we calculate the center of the cluster. Then we calculate the distance of all the data points from the center of the clusters. In step 4, depending on the distance, we assign the data points to the nearest clusters. In step 5, we repeat steps 3 and 4.

We repeat the same process until data points don’t change the clusters (the center of all the clusters are stabilized, clustering is successful) or reach the assigned number of iterations. We get the optimized clusters after these steps and clustering is successfully performed.

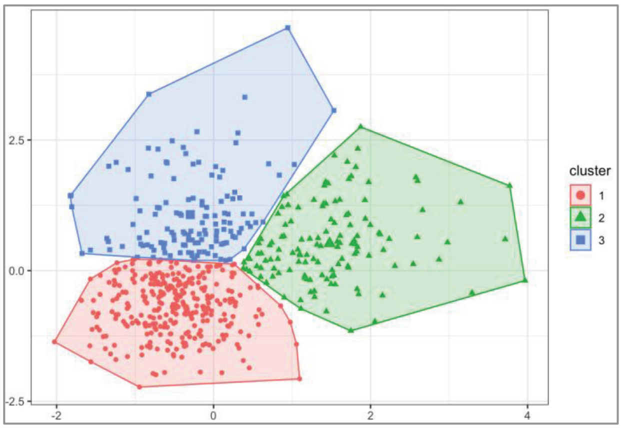

Fig. 2. K-Means clustering

C. *U-Net*

U-Net is one of the most popular network architectures in semantic segmentation, developed by Olaf Ronneberger [11].

U-Net learns from fewer training samples since it is a Fully Convolutional Network (FCN). U-Net consists of four blocks of encoders and decoders that are connected via a bridge to give it a U-shaped structure. The encoder doubles the number of filters and reduces the spatial dimensions of the feature maps to half, and doubles the feature at each encoder block. Similarly, Decoders in U-Net double the number of spatial dimensions of the feature maps and reduce the number of filters to half at each encoder block.

Skip connections in U-Net architecture help generate better semantic features by providing additional information to the Decoder. Skip connection is a shortcut that allows the indirect flow of gradients to the initial layers without any degeneration.

U-NET architecture is like a bridge that completes the flow of information by connecting encoders and decoders.

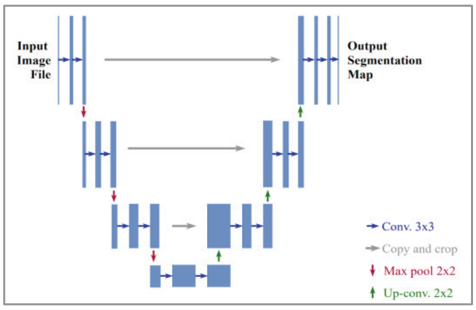

Fig. 3. U-Net architecture

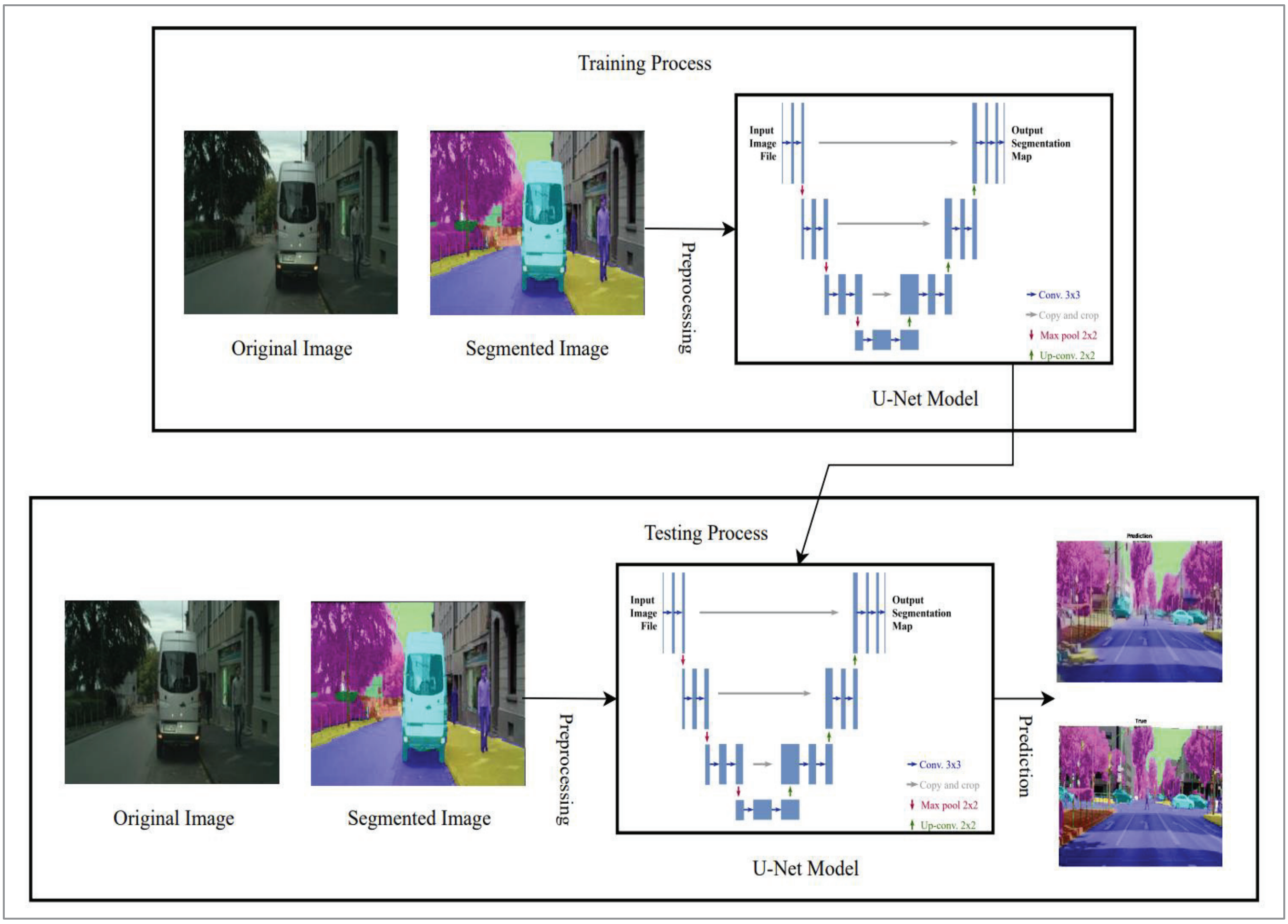

Fig. 4. Workflow of the model

IV. PROPOSED SYSTEM

We are using the Cityscapes dataset. There are 3475 images in total, where 2975 are training images files, and 500 are validation images. The size of each image is 256x512 pixels. Each image file contains an original image on the left and a labeled (segmented) image on the right. A function is used to separate the image.

To find the essential colors in the image, we used color clustering using K-means. This enables us to represent classes. After K-means clustering, the image contains data about 14 different layers. By using RGB color space, the layer representation is converted into color representation.

Image augmentation is used to rotate and flip the images to increase the adequate size of the dataset.

We proposed a model similar to the U-Net type network. Then we train the model and use checkpoints to save the model with the smallest validation loss. Then we visualized the validation accuracy, IOU score, and validation loss. Our model is evaluated on evaluation metrics such as accuracy, Balanced Accuracy (BAC) Score and IOU.

The proposed work consists of 4 Modules:

1. Data Collection
2. Data Preprocessing
3. Development of Model
4. Model Evaluation

A. Data Collection: -

The dataset which we used is the CityScapes dataset for our model. It can be found on their website: https://www.cityscapes-dataset.com/.

This dataset contains dashboard images and their segmented halves of a driving car in Germany. Dataset consists of 2975 training and 500 validation image pairs.

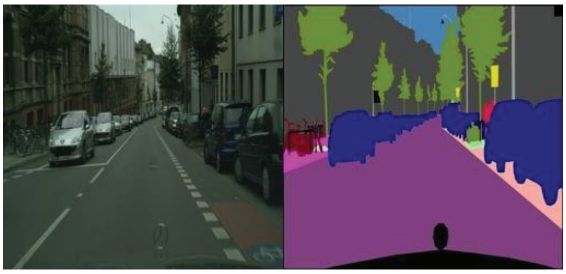

Fig. 5. Sample data from cityscape dataset

The above image shows a sample from the dataset which contains a dashboard and segmented image pairs. Different colors are clearly visible which represent different classes. For example, streets in magenta and cars in blue.

B. Data Preprocessing: -

First, we load the original images of size 512x256. Then we cut the image in half to separate raw and segmented images and remove the bottom portion of the image to remove the engine hood part. Now, we have images of size

256x200 pixels each. We store raw and segmented images in separate arrays to make the work easy.

Color Clustering: Since the image is not a collection of single-color classes, but is a mixture of them. So, we need to cluster similar data to divide them into classes, and for this purpose, we will use K-Means clustering. We tested that there are 14 frequently occurring classes out of a total of 30 classes in our dataset. So, we will use 14 classes. Now, the clustering is used to determine the 14 most frequently occurring and most different colors. These colors are replaced by colors assigned by us.

Color to class: Now, the color representation of the image is converted to the 14-layer class representation to be understandable by the model.

Now, the more data, the better our model will perform. So, we perform image augmentation to increase the data for the model. For that, we are rotating and flipping the images. Now, the data is ready to be fed into our U-NET model.

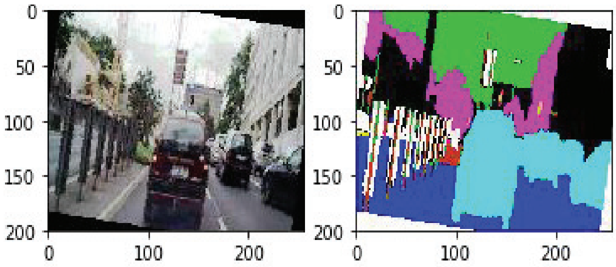

Fig. 6. Processed image after image augmentation

C. *Development of Model: -*

We have built a deep convolutional neural network using U-Net architecture. This architecture gets this name because of its U-shaped architecture or we can say it has a double funnel shape.

Our Model has the following characteristics:

- Initially, the encoder doubles the number of filters, reduces the spatial dimensions to half, and doubles the feature at each encoder block.
- The max-pooling layer reduces the spatial dimension of the feature maps by half, thus reducing the trainable parameters and computational cost.
- Skip connection acts as a shortcut connection that allows the indirect flow of gradients to the initial layers without any degeneration.
- Decoder doubles the number of spatial dimensions of the feature maps and reduces the number of filters to half at each encoder block.
- Then, there's the bridge that completes the flow of information by connecting encoders and decoders.
- We have used Adam optimizer.
- The loss function we used is categorical cross-entropy.
- The ReLU (Rectified Linear Unit) activation function is used with convolutional layers to avoid vanishing gradient problems while training. Also, it includes non-linearity which increases model capability and helps in the better generalization of the training data.
- The softmax activation function is used at output layers to limit output values from 0 to 1.

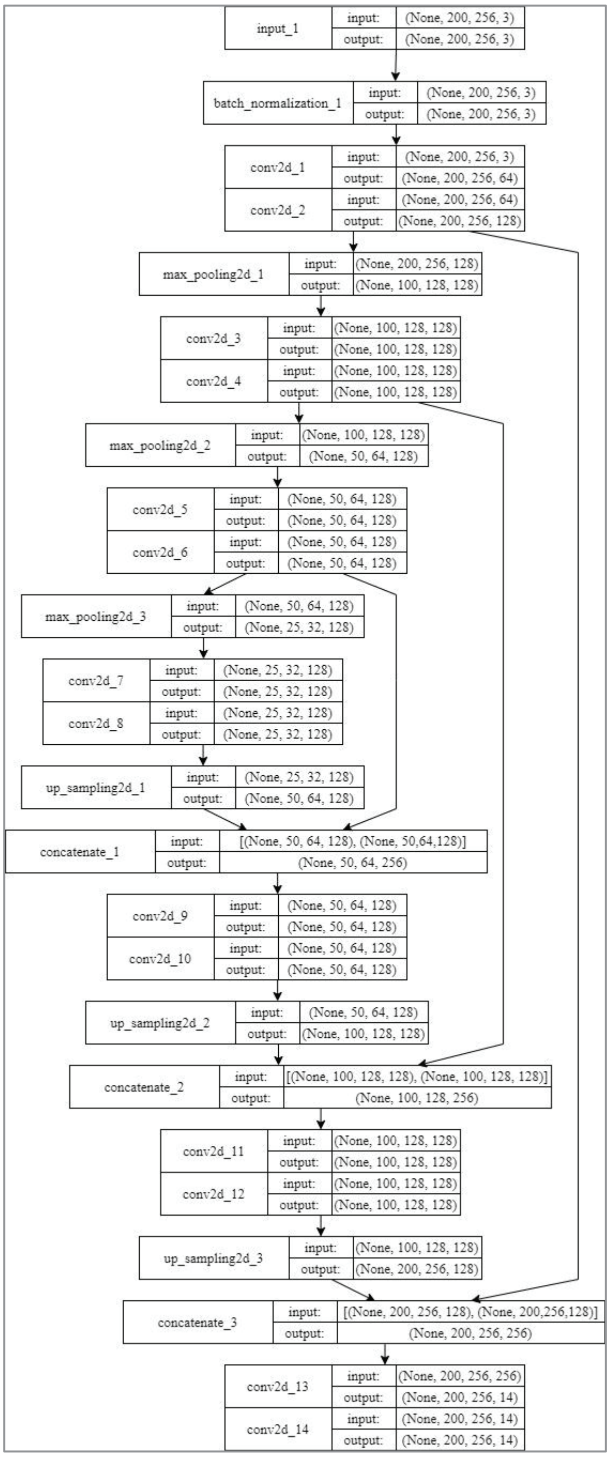

Fig. 7. Full schematic diagram of our network architecture

D. *Model Evaluation: -*

Now, we evaluate our models on certain evaluation metrics such as accuracy, ROC curve, BAC, IOU score.

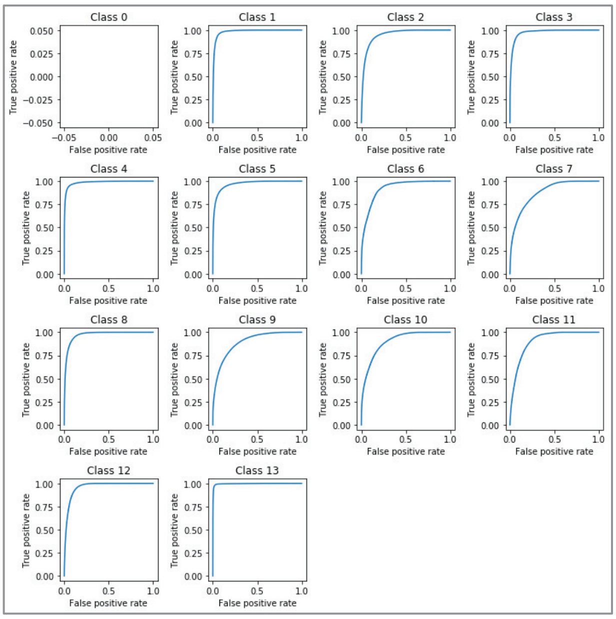

Fig. 8. ROC curves for all k classes

Fig. 8. given above shows the ROC curves for all 14 classes used in our model.

Fig. 9. given below shows the Validation Loss and Validation Accuracy of our model. The model learns quickly as can be seen in the figure and reaches the saturation point at around 300 epochs.

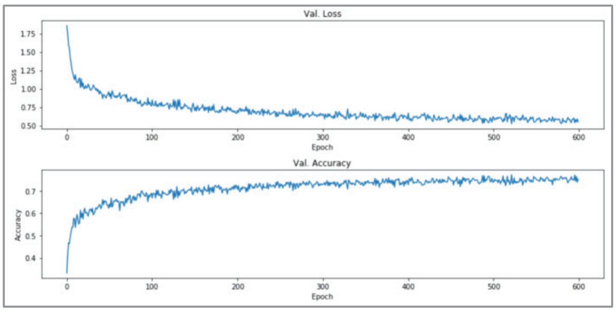

Fig. 9. Validation loss and Validation accuracy curve

TABLE I. BAC AND IOU SCORES FOR 14 CLASSES

Class	Balanced Accuracy	IOU Score
0	1	1
1	0.93	0.86
2	0.86	0.76
3	0.76	0.70
4	0.90	0.83
5	0.89	0.80
6	0.52	0.51
7	0.52	0.50
8	0.67	0.63
9	0.59	0.56
10	0.54	0.53
11	0.51	0.50
12	0.52	0.51
13	0.94	0.88

V. RESULTS

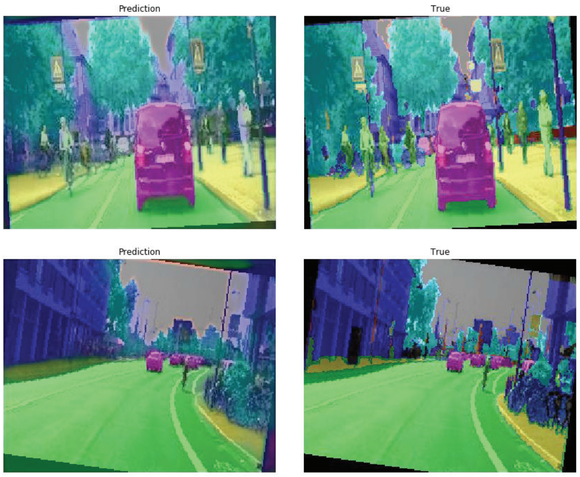

Fig. 10. Figures showing predicted image on the left and true image on the right

The Model performs pretty well in the segmentation task which can be seen from these images. We can clearly distinguish streets, sidewalks, cars, and trees.

Class 0 doesn’t have a ROC curve because there is not any classification for this class. The ROC curve for each class is shown in Fig. 9. Our model predicted most of the classes accurately. More detailed information can be drawn from BAC and IOU scores of each class which again showed good results.

The accuracy is 71.27% which is really good considering the fact that we are working with 14 classes and there are around 51200 pixels per image. There is still some room for improvement which can be achieved by training the model on a higher number of epochs.

VI. CONCLUSION

In this paper, we attempted to solve the problem of identifying objects on the road for self-driving cars using Semantic Segmentation. Among various network architectures, we used U-NET for semantic segmentation. For segmentation, we used color clustering using K-means clustering. Later we optimized our model for the training data using Adam optimizer. We also calculated the Balanced accuracy and Intersection over Union (IOU) score for evaluating our network architecture.

In the future study, we plan to design better methods for segmentation like mask R-CNN. Different network architectures can also be used like VGG, Res-Net, etc. to draw comparisons. There is also a scope of improvement by using multiple network architecture together.

REFERENCES

[1] J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp 3431-3440, 2015.

[2] C. Liang, G. Papandreou, I. Kokkinos, K. Murphy et al, “Semantic image segmentation with deep convolutional nets and fully connected crfs,” in International Conference on Learning Representations (ICLR), 2015.

[3] S. Zheng, S. Jayasumana, B. Romera-Paredes et al, “Conditional random fields as recurrent neural networks,” in International Conference on Computer Vision (ICCV), 2015.

[4] M. Treml, J. Medina, T. Unterthiner et al, “Speeding up semantic segmentation for autonomous driving,”, in MLITS, NIPS 10 Workshop, 2016. .

[5] A. Sagar, and R. Soundrapandiyan, “Semantic segmentation with multi-scale spatial attention for self-driving cars,” in Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 2650-2656, 2021.

[6] M. Siam, M. Gamal, M. Abdel-Razek et al, “A comparative study of real-time semantic segmentation for autonomous driving,” in Proceedings of the IEEE conference on computer vision and pattern recognition workshops, pp. 587-597, 2018

[7] M. Yang, K. Yu, C. Zhang et al, “Denseaspp for semantic segmentation in street scenes,” in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3684-3692, 2018.

[8] D. Feng, C. Haase-Schuetz, L. Rosenbaum, H. Hertlein et al, ‘‘Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges,” 2019, arXiv:1902.07830. [Online]. Available: http://arxiv.org/abs/1902.07830

[9] B. Chen, C. Gong, and J. Yang, “Importance-aware semantic segmentation for autonomous vehicles,” in IEEE Transactions on Intelligent Transportation Systems, vol. 20, pp. 137–148, 2019.

[10] S. Ray, R. H. Turi, “Determination of number of clusters in k-means clustering and application in colour image segmentation,” in Proceedings of the 4th international conference on advances in pattern recognition and digital techniques, pp. 137-143, 1999.

[11] P. Fischer, and T. Brox et al, “U-net: Convolutional networks for biomedical image segmentation,” in International Conference on Medical image computing and computer-assisted intervention, pp. 234-241, 2015.