Proceedings of the 5th International Conference on Smart Systems and Inventive Technology (ICSSIT 2023)
DVD Part Number: CFP23P17-DVD; ISBN: 978-1-6654-7466-5

Identification of Road and Surrounding Obstacles using U-Net Architecture for Better Perception

Raghava SaiNikhil[1]
M.Tech (CSE) Student
Gokaraju Rangaraju Institutte Of Engineering and Technology (GRIET)
Hyderabad,Telangana
nikoonikhil22@gmail.com

Dr S Govinda Rao[2]
Professor in CSE
GRIET
Hyderabad,India
govind.griet@gmail.com

Dr P Vara Prasada Rao[3]
Professor in CSE
GRIET
Hyderabad,India
prasadp.griet@gmail.com

Abstract— The ability to understand the road and traffic layout is necessary to implement vision-based autonomous driving. The process involves detecting and classifying images of roads, pedestrians, vehicles, etc. Additionally, driving videos can be used to track different patterns of motion based on their spatial location. Temporal connections between lines show the arrangement of roads and their surroundings. For an accurate understanding of the road profile, it is highly required to identify its various areas; roads, roadside, lane marks, vehicles, etc., make up the road profile in real-time. This problem can be solved by segmenting the different objects into different classes. By categorizing each pixel in an image, image segmentation is accomplished. Also, Self-driving vehicles is a highly concentrated research field where most of the researchers, industries and startups are focusing these days. In order to provide perception to a vehicle there is a lot of other things to be taken into consideration. For an instance a human can classify the surrounding objects by its features the same way a computer or robot needs to understand its surroundings for better perception, the proposed experimentation works on semantic segmentation-based road and surrounding detection. In order to carry out the experimentation, a CNN based U-Net architecture is selected to achieve the semantic segmentation and detection of surrounding objects. For training and validating the model, various images from the simulated urban environment were employed. The U-Net model was found to be 95.7% accurate in the experiment.

Keywords: Convolutional Neural Networks, Semantic Segmentation, U-Net

I. INTRODUCTION

There is a rapid advance in research today towards developing intelligent vehicles that are safe and enjoyable to drive. Several active safety features are being developed as part of advanced driver assistance systems (ADAS), such as pedestrian and vehicle detection, lane recognition, etc., based on external environments. It is a method of identifying individual pixels in an image using predefined labels or classes, also known as pixel-level classification. The technology has a number of applications in medical imaging, robotics, and autonomous vehicles, among others. Semantic segmentation is a critical element of intelligent vehicles since they should be able to understand and contextualize the surrounding environment in order to be integrated safely on current roads. The development of autonomous vehicles started in 1989, but due to limitations of conventional artificial neural networks and hardware resources, there were limitations with their implementation(1). The development of intelligent vehicles has been accelerated with advances in convolutional neural networks and GPU technology. Research on this topic is occurring in this era, however, the wide variation in geography continues to pose a challenge.

Deep learning has revolutionized the research field of semantic scene segmentation, which has seen great strides in recent years. Figure 1 illustrates the general design of a semantic segmentation framework using CNN-based encoder-decoders. However, although scene understanding in complex real-world scenarios has shown significant progress in recent years, this task remains challenging compared to the human-level understanding of a scene. Semantic image segmentation algorithms were traditionally employed prior to the advent of CNN-based approaches and based on handcrafted features together with classical classifiers. Nevertheless, since CNNs were able to successfully classify images at a high level of accuracy, they are also being used in a semantic segmentation framework for their capability to extract features. To obtain the high-level feature map of an original image, CNN reduces the original image's resolution by a factor of 32. CNN's have outperformed humans at classification tasks requiring a small feature map where only one dominant object is present in the image(2). CNN fails to segment the image as the tiny feature maps lose most of the spatial information needed to analyze complex scenes.

Developing a good road monitoring system is crucial for acquiring accurate information about road data and ensuring an effective and efficient outcome. This system can minimize the problems that arise from poorly maintained roads. For the most practical way of evaluating road conditions, a human expert can perform a visual inspection(3). These methods are not without their own disadvantages, including subjective results, making it difficult to compare and understand the results as well as the time commitment.

This paper investigates the automatic detection and segmentation of roads and objects in RGB image data generated within a simulation environment. This research work intends to develop a deep neural network system that can perform more than just road segmentation from urban images. As this system makes widespread applications in navigation more accessible, it will be useful for the development of object segmentation and road detection planning for autonomous vehicles.

The following is the outline of this paper: Section-II presents a comprehensive review of recent papers on brain tumor detection techniques. Section-III describes an effective architecture and pre-processing methodology for detecting

brain cancer. Section-IV provides the experimental comparison. This is followed by a description of the results and analysis in section-V. In section-VI, the conclusions are presented.

II. RELATED WORKS

In 2021 Zhixiong Nan et al., Developed a cross-attentional and inner-attentional object detection and semantic segmentation model(4). In order to fully exploit the correlation between the two subdividing branches of detection and segmentation, the cross-attention mechanism enables the development of the essential interaction between them. Further, inner attention contributes to a better representation of a feature map's characteristics in the model by helping to strengthen its representation. Using a series of encoder-decoder networks, initial feature maps are extracted from images by first using an encoder-decoder network. It is then augmented with the feature maps to obtain segmentation feature maps, which are subsequently used to produce segmentation feature maps. Finally, the authors performed semantic segmentation on the feature maps of segmentation followed by object detection on the detection feature maps. To evaluate this model, two popular public traffic datasets were analyzed. The proposed model achieved detected and segmented the road, sidewalk, building, wall, etc., with 91.4%, 67.8%, 82.5%, 38.9% respectively (4).

In 2021, Ozan Unal et al., proposed an auxiliary 3D object detection task is explicitly leveraged as localization features in a novel Detection Aware 3D Semantic Segmentation framework (DASS) (5). Through multitask training, the network's shared features are guided to be aware of per class detection features by which geometrically similar classes can be differentiated.

TABLE 1: RESULTS OF OZAN UNAL ET AL., PROPOSED AN AUXILIARY 3D OBJECT DETECTION.

	BEV [%]			Orientation [%]		
Method	Easy	Moderate	Hard	Easy	Moderate	Hard
PointRCNN[30]	**92.13**	**87.39**	**82.72**	95.90	91.77	86.92
DASS+RCNN	91.74	85.85	80.97	**96.20**	**92.25**	**87.26**

The authors also demonstrate how the added supervisory signal improves 3D orientation estimation capabilities by using DASS to generate high recall proposals for existing 2-stage detectors. There are two partial datasets used in this pipeline: (1) semantic labels for pointwise labels and (2) 3D object annotations for pointwise labels. Segmentation datasets are cropped to avoid introducing additional domain shifts since the detection datasets only provide annotations for the image FOV. Training for the 3D semantic segmentation task and the auxiliary 3D proposal generation task is performed using PointNet++ feature extractors trained on supervisory signals. An overview of network extensions. For PointRCNN, the RPN is DASS. Before generating a proposal, semantic feature fusion (SFF) is used to improve the results.Using the PointNet++ encoder-decoder, four set abstraction layers with multi-scale grouping are used in this architecture [6]. In each of the two scales, three linear layers follow the set abstraction layers' grouping and sampling operations [6].To obtain per point feature vectors rich in semantic and class-specific information, 4 feature propagation layers with skip connections are fed into the set abstraction layers. We can introduce scale invariance to our network by using 2-scale grouping, however, the hierarchical structure of the PointNet++ feature extractor captures more local properties

that are advantageous to both tasks. Object proposal generation and 3D semantic segmentation heads use the same 1D convolution layer of size 128. Every layer is activated by batch norms and ReLUs. There is a learning rate of 0.002 (6). A one-cycle learning rate scheme is used with Adam optimizer. Default values for weight decay and momentum are 0.001 and 0.9, respectively. These experimentations done using Semantic-KITTI and KITTI object datasets and achieved good results which are shown below.

In 2020 Jianbo Liu et al., proposed a holistically-guided decoder model to obtain high-resolution semantic-rich feature maps via the multi-scale features from the encoder for semantic segmentation(6). By combining high-level and low-level features from the encoder, the authors created a novel, holistic codeword generation and assembly operation that allows for decoding. Accordingly, the researchers have implemented the Efficient FCN architecture as a way of semantically segmenting data and the HGD-FPN architecture as a method of object detection and segmentation of instances for the proposed holistically-guided decoder. Using the proposed holistic-guided decoder, authors feed the encoder feature maps into the decoding algorithm, which then uses the output up-sampled feature map, *f8, to perform the classification process. For the initialization of the encoder network, the pre-trained weights from ImageNet are utilized. Using the proposed Efficient-FCN model on the PASCAL VOC 2012 test set, 85.4 % of the test sets were obtained without using the MS COCO dataset pre-training and 87.6% with the MS COCO dataset pre-training (7).

In 2020, Young wan Lee et.al., proposed a simple and efficient anchor-free instance segmentation, called Center Mask, which adds a novel spatial attention-guided mask (SAG-Mask) branch to anchor free one stage object detector (FCOS) in the same vein as Mask R-CNN [7]. FCOS object detector and the SAG-MASK branch are used together with the spatial attention map in this proposed system. The spatial attention map helps to focus on informative pixels and suppress noise by detecting and predicting segmentation masks on each detected box. Also, the authors propose an improved backbone network, VoVNetV2, that introduces two new useful strategies: (1) residual connections to alleviate some of the problems associated with optimization in larger VoV-Nets, and (2) effective squeeze excitation (eSE) to deal with the issue of channel information loss associated with original squeeze excitation. They evaluated their model using AP_{mask} and AP_{box} and achieved 38.3% and 43.5% for VoVNetV2-99 model, 36.6% and 41.5% for VoVNetV2-57 model, 35.6% and 40% for VoVNetV2-39 and VoVNetV2-19 got 32.2% and 35.9% (7).

Using pixel-wise person segmentation with other DCNNs, Lin et al., in 2017, presented a quandary(8). The problem with semantic segmentation networks that existed before Refine-Net is the same problem that they all face. In addition to too many convolutions and pooling, they use too many other techniques. Layers cause the image to shrink and the texture map to be blurred. This problem has been addressed by the Refine-Net: Multi-Path Refinement Networks for High-Resolution Semantic Segmentation of 2016 paper(9). Multipath refinement is introduced at each stage of

the proposed method. The proposed network Refine-Net came up with an IoU of 83.4%.

III. PROPOSED METHODOLOGY

A. Dataset

CARLA self-driving car simulator images and segmentations are provided in this dataset. Lyft Udacity Challenge participants generated the data. The dataset consists of 5 sets of 1000 images and corresponding labels. It allows you to train neural network algorithms for semantic segmentation of cars, roads, etc. The dataset originates from CARLA and was produced in 2018(9).

Fig1: Carla Simulated dataset

B. *Data Preprocessing*

Convolutional Neural Networks use pre-processed image data for training by converting the input image data into meaningful floating-point tensors. In the case of the 64 X 64 image, the tensor will have the following dimensions: (64, 64, 3)(10).

Our dataset has images with a resolution of 600 * 800 * 3 pixels for which we trained our model. A possible solution is to resize images to smaller dimensions in order to simplify the model training process. 96 * 128 pixels have been changed to the input images (11).

C. *Data Annotation*

In order to show the data features your model is supposed to understand on its own, deep learning uses image annotations to label or categorize an image either with text, annotation tools, or both, in order to essentially create your mode (12)l. A dataset can be annotated by adding metadata to it as a result of an annotation. Data annotation is sometimes referred to as tagging, transcription, or processing, and is a type of labeling for images. As well as annotating videos continuously, you can also do it frame by frame. To train the model using the annotated images, and can use supervised learning to train it. When deploying the model, you want it to recognize features in images that aren't annotated and take appropriate action on that basis. Object and boundary detection, meaning, and whole image understanding are among the most commonly used annotations. Machine learning models need to be trained, validated, and tested to achieve each of these outcomes. In this experimentation we got both RGB and annotated images to train our model.

D. Architecture

We turned to U-net as it has been widely used in medical image analysis, mainly in cardiology and neurology, in the field of computer vision research image processing. Deep neural networks are an important component of high-resolution image segmentation, so we used deep neural networks in their analysis. Medical images of various types have been segmented very well by U-Net. The architecture is composed of 23 convolutional layers, each of which is composed of an encoding and decoding step which is shown in fig-2. For instance, Seg-Net employs repeated convolution blocks for its encoder. Activation is followed by a maximum pool operation (2*2), and each layer includes size filters (3*3). Each sub-sampling doubles the number of feature cards. An inverted version of the contraction part of U-Net is the expansion part. A characteristic card number is halved every time a block is completed. Concatenating the characteristic cards of the encoder and the decoder parts allows the encoder to be connected to the decoder. To provide classification maps with the same number of classes as the desired classes, the last layer uses convolution with size filters (1 * 1). This will result in a smaller output than the input. Using convolution, rather than adding pixels around the image, will result in smaller output. An overlap mosaic strategy is used to predict the entire image part by part to get the same entry size.

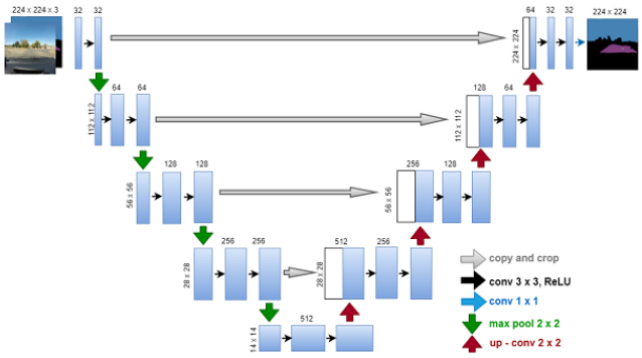

Fig 2: U-Net Architecture

E. *Optimizer:*

Gradient descent is an optimization technique based on adaptive moment estimation. When dealing with large problems with many variables or data, the method is very efficient. Memory is consumed less and the method is more efficient. It is an algorithm that has been developed by merging the 'gradient descent with momentum' algorithm with the 'Root Mean Square Propagation' algorithm and ADAGrad (13). In Adam, gradient descent is combined with one of two methods: By taking into account the 'exponentially weighted average' of the gradients, this algorithm is used to accelerate the gradient descent algorithm. The algorithm converges towards minima faster by using averages. The formulation of Adam Optimizer is,

$$m_t = \beta_1 m_{t-1} + (1-\beta_1)\left[\frac{\delta L}{\delta \omega_t}\right] \mathcal{V}_t$$

$$= \beta_2 \mathcal{V}_{t-1} + (1-\beta_2)\left[\frac{\delta L}{\delta \omega_t}\right]^2$$

IV. EXPERIMENTATION

The proposed model has preprocessing steps which were implemented using keras and tensorflow framework. In training our U-Net model, fig-3, we used the segmentation masks corresponding to the cropped 96 * 128 carla simulated environment images. To represent pixels with roads we used three channels of input (RGB) and one channel of output

(UNet). To avoid interpolation errors, we used padded convolutions. Using pixel-wise soft-max activation, we calculated the negative log likelihood of the cross-entropy loss. Authors and Affiliations.

$$loss(x, class) = -log\left(\frac{exp(x[class])}{\sum_j \exp(x[j])}\right)$$

In this case, x[j] denotes the output score of class j.

1) Training: The batch size for our model was set to 16 and we trained it for 20 epochs. No transfer learning was applied to the model. To train the model, resized images were used along with their corresponding masks.

2) Optimization: HeUniform was used to start the model, and Adam was used as an optimizer to train the gradient descent. An operating rate of 0.0001 was used to train the model. The learning rate was calibrated to 0.001. As a cost function, the crossentropy was used. We choose our patiance level to 5 if the validation accuracy dosent improve in patiance level the model cosidered its best weights stops traing which helps to avoid the overfitting problem. The learning accuracy is shown below in fig.3.

```
Epoch 1/20
60/60 [==============================] - ETA: 0s - loss: 1.8912 - accuracy: 0.4039
Epoch 00001: val_accuracy improved from -inf to 0.56928, saving model to .\best_weights.hdf5
60/60 [==============================] - 73s 1s/step - loss: 1.8912 - accuracy: 0.4039 - val_loss: 1.4200 - val_accuracy:
0.5693 - lr: 0.0010
Epoch 2/20
60/60 [==============================] - ETA: 0s - loss: 0.9290 - accuracy: 0.7025
Epoch 00002: val_accuracy improved from 0.56928 to 0.76022, saving model to .\best_weights.hdf5
60/60 [==============================] - 66s 1s/step - loss: 0.9290 - accuracy: 0.7025 - val_loss: 0.7180 - val_accuracy:
0.7602 - lr: 0.0010
Epoch 3/20
60/60 [==============================] - ETA: 0s - loss: 0.5138 - accuracy: 0.8313
Epoch 00003: val_accuracy improved from 0.76022 to 0.88180, saving model to .\best_weights.hdf5
60/60 [==============================] - 65s 1s/step - loss: 0.5138 - accuracy: 0.8313 - val_loss: 0.5355 - val_accuracy:
0.8818 - lr: 0.0010
Epoch 7/20
...
60/60 [==============================] - ETA: 0s - loss: 0.1459 - accuracy: 0.9527
Epoch 00016: val_accuracy did not improve from 0.88719
60/60 [==============================] - 64s 1s/step - loss: 0.1459 - accuracy: 0.9527 - val_loss: 0.3425 - val_accuracy:
0.8865 - lr: 1.0000e-06
Epoch 00016: early stopping
```

Fig3: Training model and accuracy

3) Regularization: The weight decay and dropout rates we used for regularization were 0.0001 and 0.1, respectively. Dropout was applied along the expansive path's last layer.

4) Hardware and Framework: Our network implementation was built using keras and tensorflow frameworks, which ran on an Intel-i7 at 3.40GHz, 3.41GHz, Nvidia RTX3060 GPU and 16GB of RAM.

V. RESULTS

In this proposal, a semantic segmentation architecture is based on Tensor-flow-2.0. An image with dimensions 600*800*3 is resized to 96*128*3, and the network is trained for 16 batch sizes and 20 epochs. The model training accuracy increased every epoch upto 13 epoch after that the model

reached its saturation state where the validation accuracy did not improve after the 12th epoch. The training accuracy of the proposed model on the carla simulated dataset achieved 95.27 accuracy where as validation accuracy of the model achieved 88.65% of accuracies. The accuracies and loss of the model graphs are shown below in fig-4 and fig-5.

In the model accuracy graphs the blue shows the training accuracy where the initial accuracy started at 40% at the first epoch and gradually increased directly proportional with epochs and reached a saturation state 13 epoch.

An analysis of a Carla validation dataset using a proposed semantic segmentation algorithm. Images from an unstructured environment are displayed in the first column. The second and third columns show the predicted segmentation map as well as the ground truth map also called as true mask where different color's represent different classes shown in figure.6

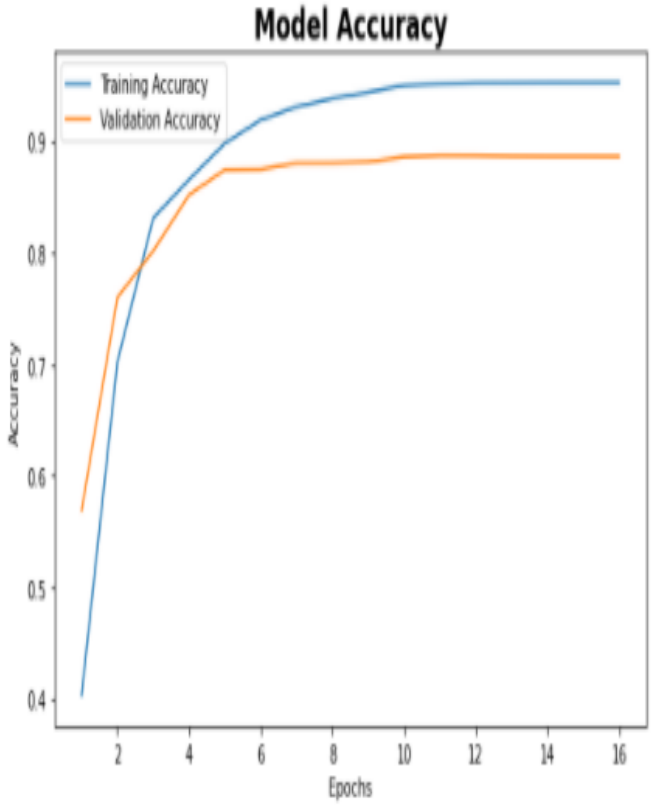

Fig 4: Training and validation model accuracies

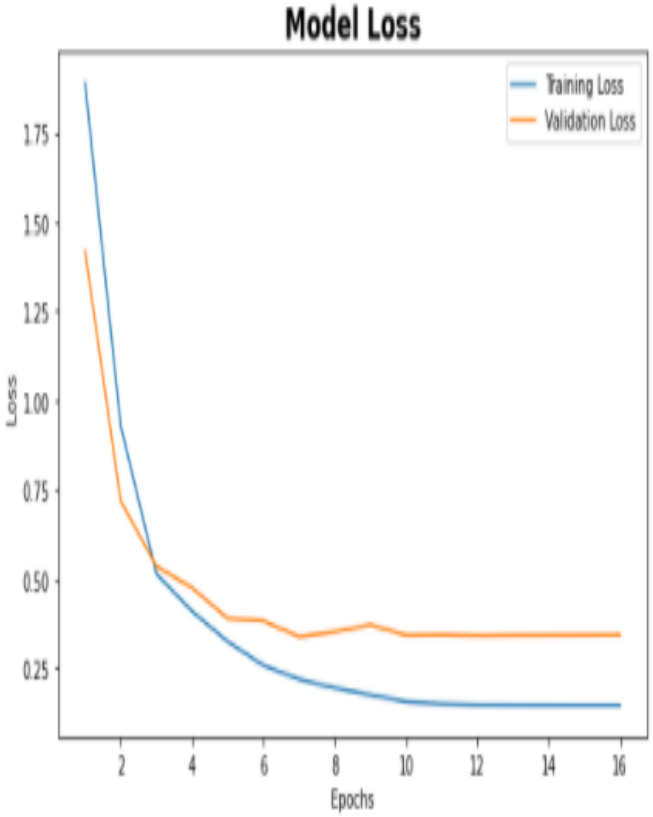

Fig5: Training and validation model accuracies

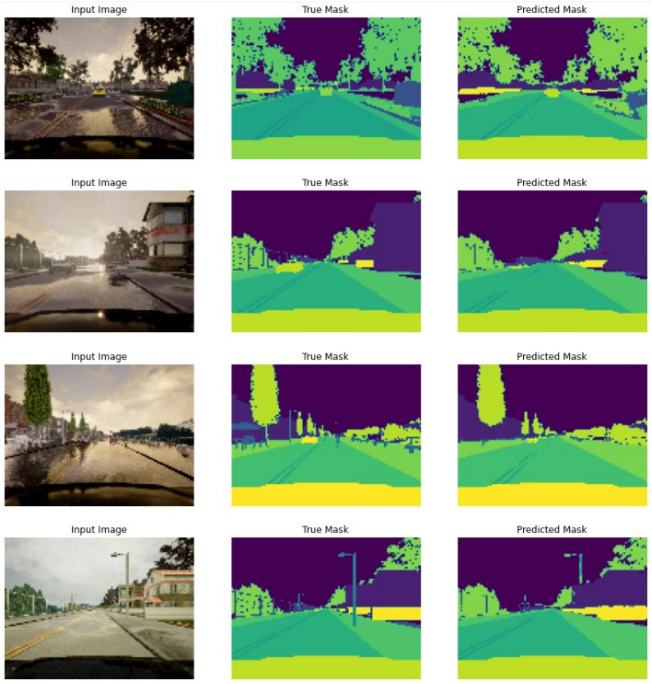

Fig6: Training model and accuracy

A. Novelty

There are two major novelties in the U-net model. To begin with, there are three bands used in the segmentation process: the segments themselves, their inner segments (which have been reduced by two pixels), and their borders. There is an overlap of one or two pixels between all of these mask layers. As a second step, each U-net's results are concatenated with those of the other two networks, followed by several convolution layers, and the result is an activation map of the mask layer of each network. As a result of these two novelties, the mask and the model are now significantly more redundant, which appears to improve the convergence of the model. Specifically, it learns the spatial relationship between the mask layers, as well as the mask layers individually.

CONCLUSION

In the compelled and unstructured environment of developing nations, improving the semantic segmentation system is an arduous endeavor. As part of IDD Lite's semantic division challenge, we proposed a clever method for pixel-level division. To achieve our goal of exact segmentation, we propose to utilize U-Net architecture, which combines both undeniable level elements and low-level spatial data. This results in improved performance due to the U-Net keeping up with compound scaling of the network. This proves the effectiveness of the proposed approach since it has achieved the highest ranking in our experimentation and achieved 88.65% validation accuracy and 95.27% training accuracy. As this is not a highly accurate and précised to deploy in robotic

or autonomous systems but still there is a huge scope developing and research on this architecture in future. For now this model can be implemented and test on simulation environments.

REFERENCES

[1] "Hu, Xiao, F. Sergio A. Rodriguez, and Alexander Gepperth. "A multi-modal system for road detection and segmentation." 2014 IEEE Intelligent Vehicles Symposium Proceedings. IEEE, 2014.".

[2] "Ronneberger, Olaf, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation." International Conference on Medical image computing and computer-assisted intervention. Springer, Cham, 2015.

[3] "Fritsch, Jannik, Tobias Kuehnl, and Andreas Geiger. "A new performance measure and evaluation benchmark for road detection algorithms." 16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013). IEEE, 2013.".

[4] "Nan, Zhixiong & Peng, Jizhi & Jiang, Jingjing & Chen, Hui & Yang, Ben & Xin, Jingmin & Zheng, Nanning. (2021). A Joint Object Detection and?Semantic Segmentation?Model with Cross-Attention and Inner-Attention Mechanisms. Neurocomputing. 463. 10.1016/j.neu".

[5] "Unal, O., Van Gool, L., & Dai, D. (2021). Improving point cloud semantic segmentation by learning 3d object detection. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (pp. 2950-2959).".

[6] "J. Liu, J. He, Y. Zheng, S. Yi, X. Wang and H. Li, "A Holistically-Guided Decoder for Deep Representation Learning with Applications to Semantic Segmentation and Object Detection," in IEEE Transactions on Pattern Analysis and Machine Intelligence, doi: 10".

[7] "Lee, Y., & Park, J. (2020). Centermask: Real-time anchor-free instance segmentation. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 13906-13915).".

[8] "Lin, G., Milan, A., Shen, C., & Reid, I. (2017). Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1925-1934).".

[9] N. BUHAGIAR, "Kaggle," Carla, May 2020. [Online]. Available: https://www.kaggle.com/code/nbuhagiar/carla-semantic-segmentation/data.

[10] A. Fredrick, "Section.IO," 31 August 2021. [Online]. Available: https://www.section.io/engineering-education/image-preprocessing-in-python/#:~:text=In%20this%20tutorial%2C%20we%20shall,used%20to%20preprocess%20image%20data..

[11] "Pereira, Vosco, et al. "Semantic segmentation of paved road and pothole image using U-net architecture." 2019 International Conference of Advanced Informatics: Concepts, Theory and Applications (ICAICTA). IEEE, 2019.".

[12] G. Boesch, "Viso.ai," [Online]. Available: https://viso.ai/computer-vision/image-annotation/#:~:text=Image%20segmentation%20is%20a%20type,a%20specific%20object%20or%20class..

[13] prakharr0y, "Geeks For Geeks," [Online]. Available: https://www.geeksforgeeks.org/intuition-of-adam-optimizer/#:~:text=Adam%20optimizer%20involves%20a%20combination,minima%20in%20a%20faster%20pace..