International Journal of Engineering Applied Sciences and Technology, 2025
Vol. 9, Issue 09, ISSN No. 2455-2143, Pages 24-34
Published Online January 2025 in IJEAST (http://www.ijeast.com)

SEMANTIC SEGMENTATION IN SELF-DRIVING CARS USING U-NET ARCHITECTURE ON CITYSCAPE DATASET

Jing Huang, Ebou A Sowe
Computer Science and Artificial Intelligence
Wuhan University of Technology, Wuhan, Hubei, China

Abstract—**Semantic segmentation has been one of the most researched topics in the field of computer vision in recent years. This study was conducted using U-Net architecture in the context of self-driving cars on a cityscape dataset. The dataset is an urban scene image that contains all scene scenarios in a typical city. It includes 5,000 high-quality finely annotated pixel-level images gathered from 50 cities over various seasons. The U-Net model uses a pre-trained RestNet101 for its encoder for feature extraction and has skip connections between the encoder and the decoder with RELU activation. The skip connection helps to retain spatial information after down sampling, this enables the model to combine deep layers and fine-grained features. The model achieved 88\% accuracy, 80\% pixel accuracy, 85\% precision, 84\% recall and 84.49\% F1 score metric. The model was trained for 75 epochs of 2 hours and 30 minutes of training time on the cityscape dataset. The model has shown good performance by achieving high accuracy and addressing class imbalance with augmentation techniques and weighted categorical loss in the context of autonomous driving. Theproposed U-Net model with RestNet101 encoders achieved high accuracy compared to VGG16 and ResNet50 in a typical complex scene environment in the cityscape dataset.**

Keywords—**Autonomous Vehicles; Cityscape dataset; Semantic Segmentation; ResNet101**

I. INTRODUCTION

Semantic segmentation plays an important role in enabling autonomous vehicles, such as self-driving cars, to understand and navigate their surroundings smoothly and effectively.

Autonomous vehicles, sometimes referred to as driverless vehicles [1], autonomous vehicles [3] – [4], or robotic cars [2], are among the most exciting new technologies at the moment and a hot topic of study. It is a widely used perception method for self-driving cars that associates each pixel of an image with a predefined class [16]. Compared to image recognition and target location and detection, semantic segmentation not only provides object classification information but also extracts location information, which lays the foundation forother computer vision tasks [17] – [18]. The achievement of high accuracy and computational efficiency in semantic segmentation is essential for the safe and efficient operation of autonomous vehicles.

A self-driving car is capable of understanding its environment and operating with less human intervention [12].

For autonomous driving to be successful, cars must be able to collect and process data from their surroundings in real-time using cameras and sensors to create a complete picture of driving conditions [5]. These cars mainly depend on the information collected by their sensors [6]. These cars use a variety of advanced sensors, cameras, and computer vision algorithms to perceive their environment and make decisions.

Convolutional neural networks (CNN) and other deep learning techniques have recently been used to achieve sophisticated results in image segmentation and classification [5]– [9]. These networks are made up of layers that can learn the information's understructure from multilevel data. Since the characteristics that make up these layers are learned from the data and do not require human design, deep learning techniques can efficiently extract features on their own, saving time and effort [14]. CNNs have shown good results in medical analysis, such as the segmentation of brain tumours [10], liver tumours [11], and pancreatic tumours [12], as well as incomputer-aided diagnostic applications [13] to improve body health.

The U-Net has a symmetric encoder-decoder structure with skip connections that combine the corresponding decoder layers with high-resolution encoder features. This design helps the model retain spatial information, making U-Net particularly effective for tasks requiring precise segmentation.

Initially developed for biomedical image segmentation, U-Net has shown good performance in many domains, including self-driving cars. This paper uses a U-Net structure neural network on a Cityscapes dataset with the use of ResNet101 as its encoder for feature extraction.

Achieving high accuracy performance and addressing class the imbalance that causes state-of-the-art models, suchas FCN [35] and SegNet [36] to ignore important small details has long been a challenge. As a result, semantic segmentation approaches are biased towards the dominant classes during inference [15]. This paper implemented a widely used architecture on medical imaging datasets known as U-Net on a

cityscape dataset for semantic segmentation in the context of autonomous driving. By adopting this architecture to the cityscape dataset, we demonstrated that the model has great accuracy in semantic segmentation tasks, which are critical for autonomous vehicles to understand their surroundings.
Using a pre-trained ResNet101, which features 101 layers for efficient feature extraction within the U-Net model, the study delivered good segmentation results, making it particularly well-suited for complex scenes.

The key contributions of the paper include the following:

- The use of a U-Net model in a cityscape dataset to improve accuracy and address the class imbalance in autonomous driving scenes.
- The use of ResNet101 as the encoder backbone for better feature extraction compared to ResNet50 and VGG16 of the state-of-the-art models.
- The use of a proprietary scaling layer to enable seam less up sampling and concatenation improves the model's ability to capture various elements such as trees, vehicles, and road signs.
- The use of weighted categorical loss and data augmentation techniques such as flipping, rotation, random cropping and Gaussian blur increases the variety of representation of the minority classes.

II.RELATED TOPICS

A. Deep Learningfor SemanticSegmentation

The main goal of the U-Net architecture was to address the issues of limited data in the medical field. It was designed to effectively analyse a smaller amount of data while maintaining computational effectiveness. Due to its versatility, it can also be used in CamVid and cityscape datasets to perform well. Other models such as PSPNet proposed by Hengshuang Zhao et al. [25] in the paper titled "Pyramid Scene Parsing Network" presented at CVPR 2017 had a remarkable performance. The model uses a pre-trained ResNet50 for feature extraction. This pre-trained is trained on the ImageNet dataset for classification tasks. This model classified a segmented object in relation to the contextual information available within the surroundings. Other deep learning models use encoder-decoder, these types include fully convolutional networks (FCN) [5], encoder-decoder-based techniques such as Segnet [26], ERFNet [27], and U-Net [28], as well as ESPnetv2 [29].

B. Attention And Gating Mechanism

CNNs have recently improved in various vision tasks, including classification [16], detection [17], segmentation [18], image captioning [19], and visual recognition [20] using attention mechanisms. Attention processes guide the model, helping it focus on the most important features and ignoring those not relevant to a particular task. To capture long-range dependencies, Wang et al. [16] presented a residual attention network that uses non-local self-attention processes. Hu et al. [21] introduced the squeeze-and-excitation method for ILSVRC 2017 image classification with channel-wise attention computed to emphasise the valuable channels via global average pooling and surpassed the existing methods. An interesting work on self-attention was presented by Woo et al. [22], wherein they proposed a convolutional block attention module (CBAM) that leverages both spatial and channel information, allowing for effective feature refinement.

C. Other Variants And Modifications

The Dilated-UNet model improves medical image segmentation by utilizing the advantages of the U-Net architecture and the Dilated Transformer blocks [23]. The Dilated Trans- former blocks help to portray a bigger background without compromising detail.
Attention U-Net [23] is a version that integrates an attention gate to improve feature selection during segmentation, increasing sensitivity and prediction accuracy, especially in complex image contexts.

D. Challenges And Existing Problems

Achieving high-accuracy performance has long been a challenge. Specifically, ENet [31]and Fast-SCNN [32] aim to address this gap in speed and accuracy with their lightweight architectures; a crucial area for robotics, self-driving cars, and aerial vehicles, but this issue still exists.
Another challenge is a class imbalance in semantic segmentation training datasets. Certain kinds of items (such as roads and automobiles) may dominate the dataset, resulting in a model that excels at segmenting common objects but struggles with less common categories such as cyclists or pedestrians (particularly in rural regions). This lack of robustness can undermine the reliability and safety of autonomous vehicles [30]. This bias might cause significant performance disparities when facing unusual circumstances on the road, jeopardizing autonomous vehicle safety and operational dependability.
To address these challenges, we proposed a U-Net model with ResNet101 encoder, which has 101 layers suitable for featureextraction, thus achieving high accuracy and addressing the issue of class imbalance on the Cityscape dataset. We also used augmentation techniques including flipping, rotation, scaling, and random cropping to increase the variety and representation of minority classes so that the model becomes robust to such class variations.
We further used a categorical cross-entropy loss function for multi-class problems. The loss function treated each class as mutually exclusive, allowing the model to optimize predictions over majority and minority classes alike without any explicit weighting.

International Journal of Engineering Applied Sciences and Technology, 2025
Vol. 9, Issue 09, ISSN No. 2455-2143, Pages 24-34
Published Online January 2025 in IJEAST (http://www.ijeast.com)

III. METHODOLOGY

A. Dataset

The Cityscape dataset is an urban scene dataset that contains all scene scenarios in a typical city. It includes 5,000 high-quality finely annotated pixel-level images gathered from 50 cities taken over various seasons. For training, validation and testing, the images are separated into sets with the numbers 2,975, 500, and 1,525. Identifies 19 categories or classes that include both things and junk [14]. In addition, two comparison settings are given, training with only fine data or training with both fine and coarse data, 20,000 coarsely annotated images [25]. The dataset supported our experimental goal.

The dataset supported the experiment goal by dealing with various issues in semantic segmentation, like the accuracy of understanding an entire scene and scene context, which are important for advancing research into self-driving cars as well as urban scene understanding. The dataset offers high-quality images that are meticulously annotated with pixel-precise labels. Detailed annotations allow for precise evaluation of models' performance. There are various scenarios, including lighting conditions, weather, times of day, and scenes containing multiple objects. This allows models to generalize well and understand more in varied conditions, thus, driving cars can work well. It includes a variety of urban scenes with a range of object kinds and occlusions.

B. Data Preprocessing

Download and Resize: First, the dataset is downloaded and uploaded on the Kaggle cloud environment for storage and processing. The images are then resized to 200x256 pixels and stored in their respective directories of test, validation and training.

Normalization and Dataset creation: The images are then normalized between the range of [0,1] by dividing by 255.0 into three channels known as the RGB color standardization. respective directories of test, validation and training.

$$\text{image}_{\text{norm}} = \text{clip}\left(\frac{\text{image}}{255.0}, 0.0, 1.0\right) \quad (1)$$

C. Data augmentation

1) Random Horizontal and Vertical Flip

The images are then flip horizontally and vertically with a probability of 50%.

$$I'(x,y) = I(W - x - 1, y) \quad (2)$$

$$I'(x,y) = I(x, H - y - 1) \quad (3)$$

III. EXPERIMENT AND RESULT

The test set for this evaluation experiment watermark ima

2) Random Rotation

The images are rotated by a random angle, within a specified range rotation by -10 and 10 degrees for each pixel location (x,y)(x,y) is transformed.

$$x' = x\cos(\theta) - y\sin(\theta) \quad (4)$$

$$y' = x\sin(\theta) + y\cos(\theta) \quad (5)$$

3) Gaussian Blur

Blurs the image using a Gaussian kernel, it is in smoothing out noise and small details. The Gaussian kernel of size k×k and standard deviation σ

$$G(x,y) = \frac{1}{2\pi\sigma^2}\exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) \quad (6)$$

D. Model Architecture

U-Net is a popular image segmentation model developed primarily for biomedical imaging. It has been popular for its power encoder and decoder with skip connection which helps to retain useful spatial information during downsampling. The term U-Net described its U-shaped architecture of the network architecture [24].

In our model architecture, the encoder uses ResNet101 of five convolutional layers. ResNet101 is used for the feature extraction as the encoder, it contains 101 layers of trained imageNet dataset for classification tasks. The encoder in our U-Net model as shown in Fig1, extracts increasingly abstract characteristics at various spatial scales by gradually downsampling the input image; each of these convolutional layers is followed by a Rectified Linear Unit (ReLU) activation function. To restore spatial resolution, the decoder subsequently up samples these features with the help of skip connections. The decoder then used UpSampling2D layers to up sample the feature maps back to the original of the input size. The feature maps from the encoder, which is the ResNet101 are resized and concatenated with the upsampled maps in a manner similar to U-Net.

The encoder starts with a 7 x 7 convolutional layer on the input image with stride 2,capturing low-level features from the input image and reducing its spatial dimension. Then a 3x3 max pooling layer with stride 2 is applied, further reducing the size of the feature map. The ResNet blocks in this architecture are divided into four stages with multiple residualblocks within each stage. In these stages, the feature maps progressively downsample, spatial resolution reducing and channel depth increasing by; Stage 1 —64 channels, Stage2 — 256 channels, Stage 3 —512 channels, Stage 4 —1024 channels and Stage 5 - 2048channels as shown in Figure 4-1.

International Journal of Engineering Applied Sciences and Technology, 2025
Vol. 9, Issue 09, ISSN No. 2455-2143, Pages 24-34
Published Online January 2025 in IJEAST (http://www.ijeast.com)

The decoder mirrors the encoder, and progressively upsampling feature maps back to the spatial resolution of the input image

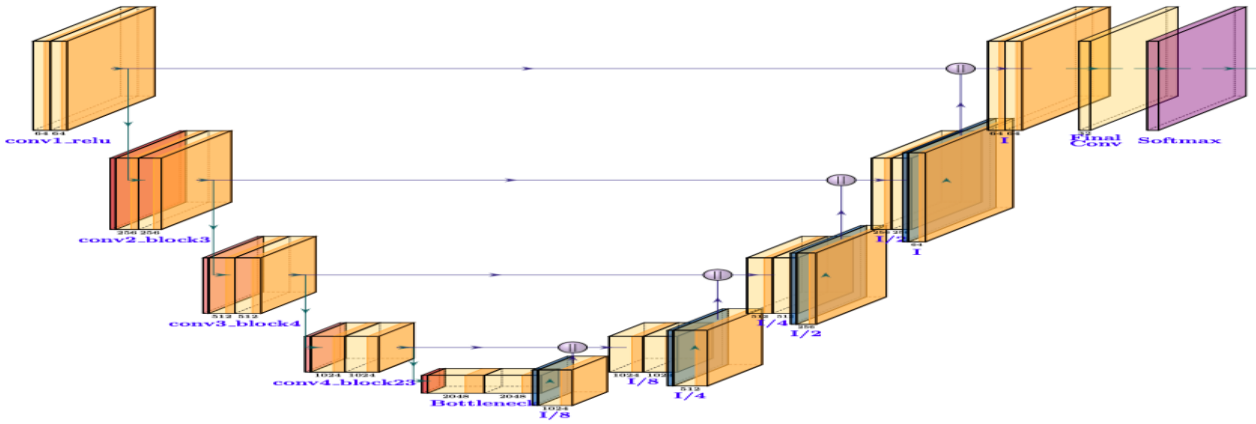

Fig1: U-Net With 101 Encoder

E. Loss Function

The weighted categorical cross-entropy loss is a commonly used loss function in multi-class classificationtasks, particularly in scenarios where class imbalance exists. It quantifies the differencebetween the true class labels and the predicted probabilities produced by the models,with an emphasis on penalizing incorrect classifications more heavily for underrepresented classes. The main objective duringtraining is to minimize this loss, encouragingthe model to produce predicted probabilities that are as close as possible to the true classdistribution while giving higher importance to misclassifications of minority classes.

The weighted categorical cross-entropy loss can be defined as:

$$L = -\sum_{i=1}^{C} w_i \cdot y_i \log(p_i) \qquad (7)$$

where:

- **C** is the total number of classes in the classification problem.
- $\mathbf{y}_i$ is the binary indicator (0 or 1) representing whether the true class label for the sample is class i For a single sample, this will be 1 for the true class and 0 for all others.
- $\mathbf{p}_i$ is the predicted probability that the sample belongs to class i, output by the model. These probabilities are usually obtained from the softmax function in the multi-class classification.
- $\mathbf{w}_i$ is the weight assigned to class i. The weights are typically chosen to be inversely proportional to the frequency of the classes in the dataset, which helps address class imbalance by giving higher penalties to misclassifications of minority classes. For instance, the weight for class. $\mathbf{w}_i$ can be defined as: $w_i = \frac{N}{N_i}$, where N is the total number of samples in the dataset, and N_i is the number of samples in class i. This ensures that classes with fewer samples will contribute more to the loss, encouraging the model to better classify them.

F. Experimental Parameter Setting

Table:1 Experimental Parameter Setting

Hyper-Parameter	Value
Dataset	Cityscape
Computational Resources	Kaggle Notebook
	GPU: NVIDIA P100 (16GB Memory)
	CPU: 57.6GB Disk, 29GB RAM
	Session Limit: 30 Hours Per week
Image Dimensions	200 × 256
Batch Size	16
Optimizer	Adam
Learning Rate	1×10^{-4}
Loss Function	Weighted Categorical class
Number of Epochs	75
Evaluation Metric	Pixel Accuracy
	F1 score
	Precision
	Recall
Augmentation Techniques	Random Flip (Horizontal/Vertical),
	Random Resizing (0.5x to 2x),
	Random Rotation (±10°),
	Gaussian Blur
	Random Brightness
	Color jittering

International Journal of Engineering Applied Sciences and Technology, 2025
Vol. 9, Issue 09, ISSN No. 2455-2143, Pages 24-34
Published Online January 2025 in IJEAST (http://www.ijeast.com)

IV. RESULTS AND DISCUSSION

The model has shown some notable results of 88% model accuracy, 80% of pixel accuracy, 84% of recall accuracy, precision of 85% and F1 score of 84.49%. This has shown that ResNet101 can be used for feature extraction as an encoder. Fig 9 shows the original, masked, and output of the predicted image on the cityscape dataset. Thus, we address the class imbalance and achieve high accuracy. Table2 shows very good and consistent performance across 19 classes in the cityscape dataset. Our model excelled in precisely identifying numerous common as well as uncommon classes. The "Road" class, in particular, got a remarkable Class Accuracy of 98.2%. This level of accuracy signifies that the model is able to correctly identify a dominant class while still solving for class imbalance.

The model has also performed much better in under-represented classes. For example: The class Accuracy of "Fence" was 89.0%, and for "Bicycle" it reached 93.7%. This is a marked improvement over the state-of-the-art models and our model fully resolves complicated issues with different object types.

Apart from Class Accuracy, other important metrics like Pixel Accuracy and F1 Score showed regular performance too. All previously underperforming classes exhibited higher

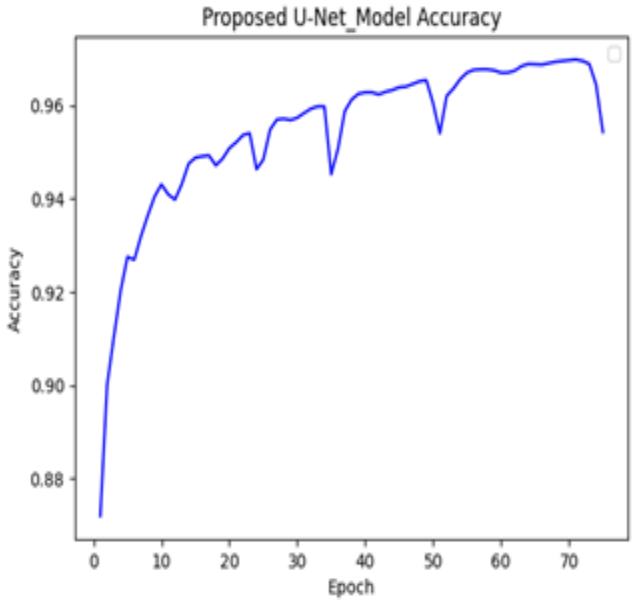

Fig2. Model Accuracy Curve

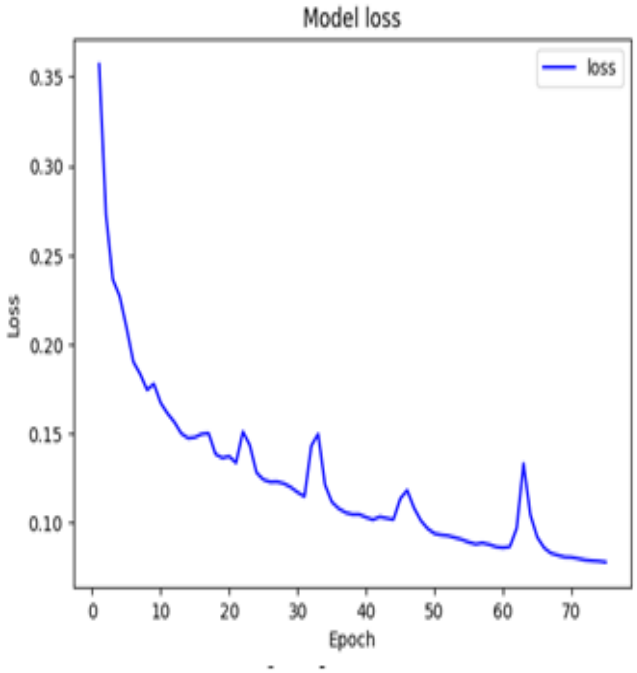

Fig3: Model LossCurve

Table 2shows very good and consistent performance across 19 classes in the cityscape dataset. Our model excelled in precisely identifying numerous common as well as uncommon classes. The "Road" class, in particular, got a remarkable Class Accuracy of 98.2%. This level of accuracy signifies that the model is able to correctly identify a dominant class while still solving for class imbalance.

The model has also performed much better in under-represented classes. For example: The class Accuracy of "Fence" was 89.0%, and for "Bicycle" it reached 93.7%. This is a marked improvement over the state-of-the-art models and our model fully resolves complicated issues with different object types.

Apart from Class Accuracy, other important metrics like Pixel Accuracy and F1 Score showed regular performance too. All previously underperforming classes exhibited higher values for these metrics. The model effectively handles common and rare classes by balancing precision with recall; this helps to reduce segmentation errors steadily and correctly by identifying all object types.

International Journal of Engineering Applied Sciences and Technology, 2025
Vol. 9, Issue 09, ISSN No. 2455-2143, Pages 24-34
Published Online January 2025 in IJEAST (http://www.ijeast.com)

Table 2: Per Class Performance on Cityscape Dataset

Class	Class Accuracy(%)	Pixel Accuracy(%)	Precision(%)	Recall(%)	F1 Score(%)
Road	98.2	89.0	93.0	91.0	92.0
Sidewalk	95.1	85.0	89.0	92.0	90.5
Building	97.3	91.0	94.0	92.0	93.0
Wall	94.5	88.0	91.0	87.0	89.0
Fence	89.0	84.0	83.0	82	82.5
Pole	96.2	88.0	90.0	89.0	89.5
Traffic Light	93.4	91.0	88.0	92.0	90.0
Traffic Sign	95.3	86.0	91.0	92.0	91.0
Vegetation	98.0	94.0	95.0	93.0	94.0
Terrain	94.3	86.0	88.0	87.0	87.0
Sky	98.0	95.0	98.0	97.0	97.0
Person	96.5	92.0	96.0	93.0	94.0
Rider	95.8	90.0	94.0	93.0	94.0
Car	97.0	95.0	96.0	94.0	95.0
Truck	93.3	87.0	91.0	89.0	90.0
Bus	96.0	91.0	93.0	91.0	92.0
Train	97.4	93.0	96.0	92.0	94.0
Motorcycle	94.3	86.0	87.0	85.0	86.0
Bicycle	93.7	84.0	88.0	86.0	87.0

A. EVALUATION METRIC

1) Pixel Accuracy

A common Pixel evaluation metric is used to further assess the model. Its calculates the percentage of the correctly identified pixels across all given classes, providing a simple but effective method for evaluating the overall performance of the model. Our model obtained 80% Pixel Accuracy. The equation for Pixel Accuracy is given by:

$$\text{Pixel Accuracy} = \frac{\sum_{i=1}^{N} 1(\hat{y}_i = y_i)}{N} \qquad (8)$$

- N is defined as the total number of pixels in the image or across all images in the cityscape dataset.
- $\hat{y}_i$ is the predicted label for the pixel i.
- y_i stands for the true label for the pixel i.
- $1(\hat{y}_i = y_i)$ is the indicator function that equals 1 if the predicted label $\hat{y}_i$ matches the true label y_i, and 0 otherwise.

Where;

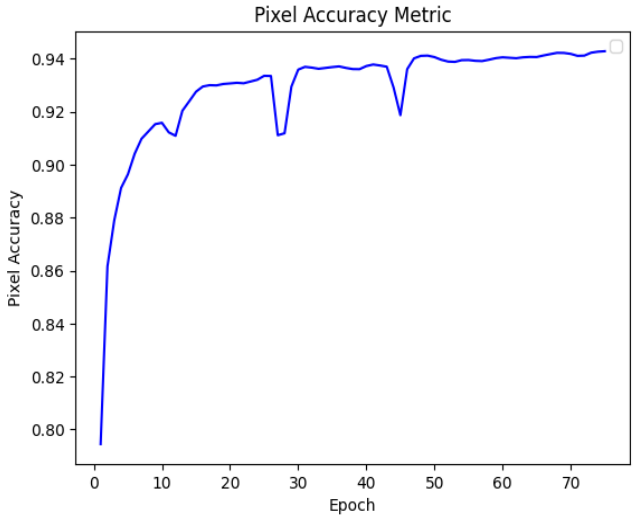

Fig4: Pixel Evaluation Metric Curve

2) Precision

Precision is the percentage of real positive pixels among all pixels labelled as positive by the model. It is an essentialmetric, particularly in circumstances where false positives might have serious consequences, such as in medical diagnostic applications and autonomous systems [33]. A high precision score indicates that a model reliably identifies the relevant pixels, minimizing instances of falsely labelling background pixels as part of the target class [34]. Our model obtained 85% precision.

International Journal of Engineering Applied Sciences and Technology, 2025
Vol. 9, Issue 09, ISSN No. 2455-2143, Pages 24-34
Published Online January 2025 in IJEAST (http://www.ijeast.com)

3) Recall

We further used the Recall evaluation metric to further assess our model by measuring the percentage of real positive samples that the model accurately detected. Thus, our model achieved 84% recall accuracy.

$$\text{Recall} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Negatives (FN)}} \quad (9)$$

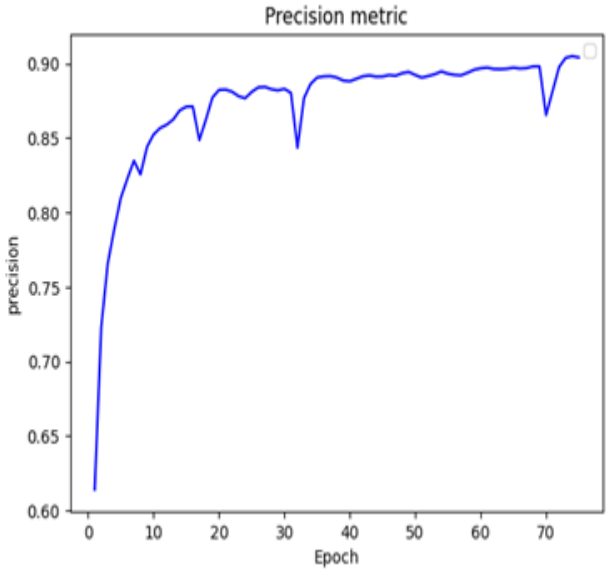

Fig5 : Precision Evaluation Metric

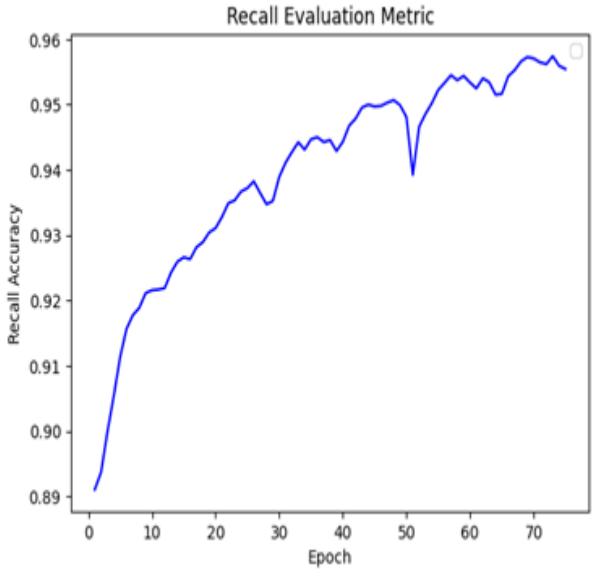

Fig6: Recall Evaluation Metric

3) F1 Score

We further analysed with F_1 score which is a harmonic mean of precision and recall, providing a balanced evaluation metric for our proposed model, where there are imbalance class distributions. We obtained an 84.49% F1 Score

$$F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \quad (10)$$

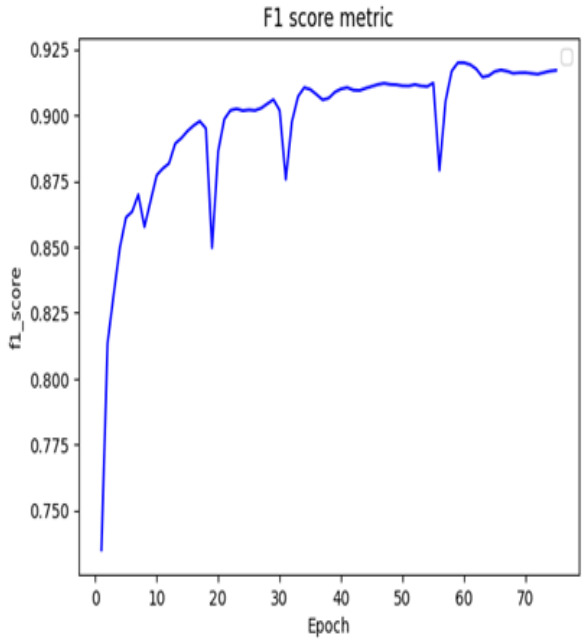

Fig7 : F1 Score Evaluation Metric

B. COMPARISON OF EVALUATION METRICS FOR U-NET WITH DIFFERENT BACKBONES

Table 3and Fig shows how U-Net performs using different backbones —VGG16, ResNet50 and ResNet101—on important benchmark metrics (Pixel Accuracy, Precision,Recall and F1 Score).U-Net + VGG16: Pixel accuracy reached 71%, precision stood at 75.2%, recall at 78.9% and F1 score was 77.0%. This setup works, but deeper networks capture more complex features.U-Net + ResNet50: An increase in performance was observed, with pixel accuracy of 75%; precision, of 80.4%; recall, of 82.6%; and F1 score, of 81.5.%

ResNet50 backbone's enhanced feature extraction and segmentation.U-Net + ResNet101: Scored highest amongst all models with a pixel accuracy of 80%, precision of 85%, recall of 84% and F1 score is 84.49%. The ResNet101 model has a deeper backbone that allows for more accurate segmentation, which is optimal in high-performance tasks.

International Journal of Engineering Applied Sciences and Technology, 2025
Vol. 9, Issue 09, ISSN No. 2455-2143, Pages 24-34
Published Online January 2025 in IJEAST (http://www.ijeast.com)

In terms of performance, it's evident that making the backbone deeper enhances U-Net's feature capture ability — essential for precise semantic segmentation. The use of ResNet50 as a backbone significantly improves upon VGG16, indicating that residual connections help to extract features better and capture more details in the data. Yet the significant performance boost from ResNet50 to ResNet101 is worth mentioning particularly since the deeper ResNet101 backbone surpassed both in all major metrics. This likely shows that its depth enables it to extract more complex features and improve generalization of the model in a complex environment. For applications such as urban scene segmentation where details are important, the accuracy, precision and recall gains of ResNet101 become highly important. So, ResNet50 does bring significant enhancement over VGG16 but ResNet101 can capture much more complexity with depth and has better trade-offs between —accuracy versus computational— segmentation making it the preferable choice for high-performance.

Table 3: Comparison of Evaluation Metrics for U-Net with Different Backbones (%)

Model	Pixel Accuracy	Precision	Recall	F1 Score
U-Net+VGG16	71	75.2	78.9	77.0
U-Net+VGG16	75	80.4	82.6	81.5
U-Net+ResNet101	80	85	84	84.49

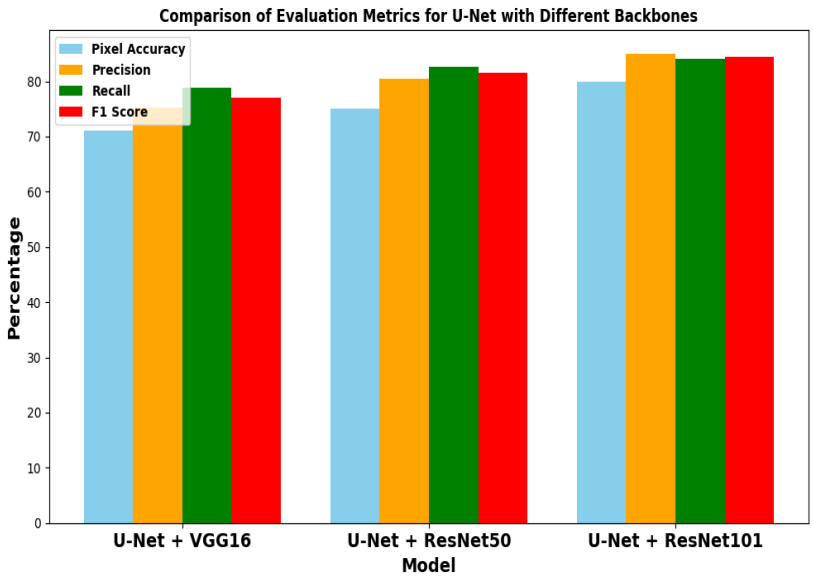

Fig8: Comparison Of Evaluation Metrics For U-Net With Different Backbones

International Journal of Engineering Applied Sciences and Technology, 2025
Vol. 9, Issue 09, ISSN No. 2455-2143, Pages 24-34
Published Online January 2025 in IJEAST (http://www.ijeast.com)

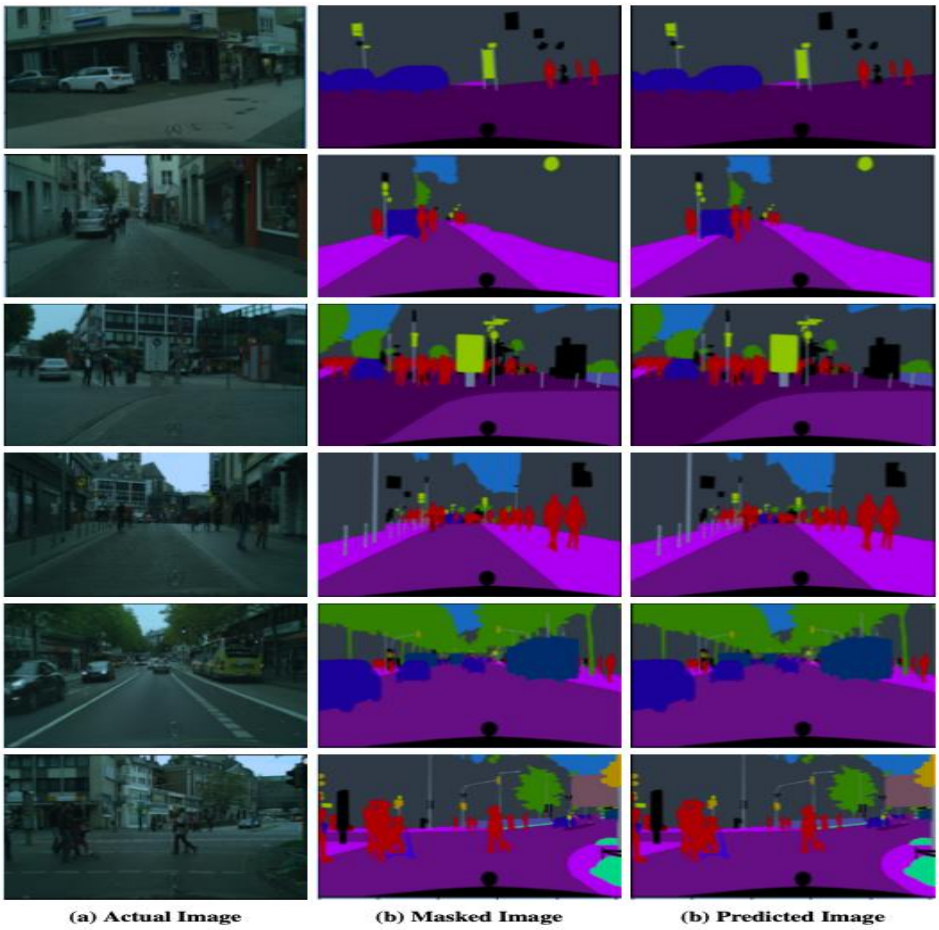

Fig 9:Proposed U-Net Model Results on the Cityscape Dataset

V.CONCLUSION

The proposed model successfully extracts detailed and contextually important information from images in a complex scene of the Cityscape dataset by utilizing the pre-trained ResNet101 architecture as an encoder. We obtained 88% accuracy, 85% precision, 80% pixel accuracy, 84% recall and 84.49% F1 score. The model trained over 75 epochs for 2 hours 30 minutes on the cityscape dataset.

Metrics indicate that our method reduced class imbalance and delivered an accurate, consistent performance for all classes. Classes with more frequency, i.e., "Road" and "Sky" obtained F1 scores of 92% and 97% respectively; whereas difficult, minority classes— "Fence" and "Truck" also performed comparatively well with F1 scores of 82% and 90% respectively.

The study findings provide validation that data augmentation and categorical loss functions help to handle class im-balance, allowing the model for reliable performance over a Using a pre-trained ResNet101 encoder within the U-Net framework also notably improved performance on the Cityscape dataset. The data when U-Net with VGG16 backbone, ResNet50 backbone and ResNet101 backbone was compared showed utilise of pre-trained encoders. For example, The pre-trained ResNet101 encoder enabled higher accuracy and F1 score than both U-Net + VGG16, and U-Net + ResNet50. Since ResNet101 has deeper layers with residual connections it can capture more fine-grained features of complex urban scenes in the Cityscape dataset. The U-Net + VGG16 was less effective because of its relative shallowness; whereas the U-Net + ResNet50, though deeper than VGG16 still performed worse than ResNet101. Pre-trained weights in ResNet101 helped the model to generalize well; boosting performance on both frequent classes ("Road" and "Sky") as well as difficult, minority classes ("Fence" and "Truck"). It proves that pre-trained encoders like ResNet101 not only help in improving segmentation accuracy but also save time and resources during training, making it a suitable candidate for high-performance segmentation on complex datasets such as Cityscape.

ACKNOWLEDGMENT

First and foremost, we would like to express my deepest gratitude to Professor **Jing Huang** for his invaluable guidance, unwavering support, and continuous encouragementthroughout this research. His profound

expertise and insightful feedback not only shapedthe direction of this thesis but also played a crucial role in helping me publish other related articles,further strengthening my research contributions. His mentorship has been an incrediblesource of inspiration, and I am truly grateful for the opportunity to work under hissupervision.

We would also like to extend my appreciation to Kaggle for providing free access to limited GPU resources, which were essential for conducting the computational experimentsrequired for this research. The availability of these resources significantly facilitated thetraining and evaluation of deep learning models in an environment with constrained computational power.The codes are available upon request on the following emails; **huangjing@whut.edu.cn** and **eboua.sowe@whut.edu.cn**

REFERENCE

[1] Arena, F., Pau, G., and Collotta, M. (2018). A Survey on Driverless Vehicles: From Their Diffusion to Security, Journal of Internet Services and Information Security, vol. 8, (pp. 1–19).

[2] Thrun, S. (2010). Toward Robotic Cars, Communications of the ACM, vol. 53, (pp. 99–106), doi: 10.1145/1735223.1735242.

[3] Litman, T. (2017). Autonomous Vehicle Implementation Predictions, Victoria Transport Policy Institute, Victoria, BC, Canada.

[4] Van Brummelen, J., O'Brien, M., Gruyer, D., and Najjaran, H. (2018). Autonomous Vehicle Perception: The Technology of Today and Tomorrow, Transportation Research Part C: Emerging Technologies, vol. 89, (pp. 384–406), doi: 10.1016/j.trc.2018.02.016.

[5] Long, J., Shelhamer, E., and Darrell, T. (2015). Fully Convolutional Networks for Semantic Segmentation, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, (pp. 3431–3440).

[6] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep Residual Learning for Image Recognition, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, (pp. 770–778).

[7] Simonyan, K., and Zisserman, A. (2015). Very Deep Convolutional Networks for Large-Scale Image Recognition, in Proc. International Conference on Learning Representations, San Diego, CA, USA.

[8] Tanzi, L., Vezzetti, E., Moreno, R., and Moos, S. (2020). X-ray Bone Fracture Classification Using Deep Learning: A Baseline for Designing a Reliable Approach, Applied Sciences, vol. 10, no. 4, (p. 1507).

[9] Gribaudo, M., Moos, S., Piazzolla, P., Porpiglia, F., Vezzetti, E., and Violante, M. G. (2019). Enhancing Spatial Navigation in Robot-Assisted Surgery: An Application, in Proc. International Conference on Design, Simulation, Manufacturing: The Innovation Exchange, Cham, Switzerland: Springer, (pp. 95–105).

[10] Havaei, M. et al. (2017). Brain Tumor Segmentation with Deep Neural Networks, Medical Image Analysis, vol. 35, (pp. 18–31).

[11] Li, W., Jia, F., and Hu, Q. (2015). Automatic Segmentation of Liver Tumor in CT Images with Deep Convolutional Neural Networks, Journal of Computer Communication, vol. 3, (pp. 146–151).

[12] Roth, H. R. et al. (2015). DeepOrgan: Multi-level Deep Convolutional Networks for Automated Pancreas Segmentation, in Proc. IEEE International Conference on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, (pp. 556–564).

[13] Shin, H. C. et al. (2016). Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning, IEEE Transactions on Medical Imaging, vol. 35, no. 5, (pp. 1285–1298).

[14] Le, T. B. K., Dao, D.-P., Ho, N.-H., and Yang, H.-J. (2024). Enhancing U-Net with Spatial-Channel Attention Gate for Abnormal Tissue Segmentation in Medical Imaging, Applied Sciences, vol. 14, no. 1, (p. 1234).

[15] Bressan, P. O., Junior, J. M., Martins, J. A. C., Melo, M. J., Gonçalves, D. N., Freitas, D. M., Ramos, A. P. M., Osco, L. P., Silva, J. A., Luo, Z., et al. (2021). Semantic Segmentation with Labeling Uncertainty and Class Imbalance, in Proc. Int. Conf. Semantic Segmentation (ICSS).

[16] Wang, F. et al. (2017). Residual Attention Network for Image Classification, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, (pp. 6450–6458).

[17] Li, H., Liu, Y., Ouyang, W., and Wang, X. (2019). Zoom Out-and-In Network with Map Attention Decision for Region Proposal and Object Detection, International Journal of Computer Vision, vol. 127, no. 2, (pp. 225–238).

[18] Li, H., Xiong, P., An, J., and Wang, L. (2018). Pyramid Attention Network for Semantic Segmentation, in Proc. British Machine Vision Conference, Northumbria, UK.

[19] Pedersoli, M., Lucas, T., Schmid, C., and Verbeek, J. (2017). Areas of Attention for Image Captioning, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, (pp. 1251–1259).

[20] Yang, Z. et al. (2016). Stacked Attention Networks for Image Question Answering, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, (pp. 21–29).

[21] Hu, J., Shen, L., and Sun, G. (2018). Squeeze-and-Excitation Networks, in Proc. IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, (pp. 7132–7141).

[22] Woo, S., Park, J., Lee, J.-Y., and Kweon, I. S. (2018). CBAM: Convolutional Block Attention Module, in Proc. European Conference on Computer Vision, Munich, Germany, (pp. 3–19).

[23] Saadati, D., Manzari, O. N., and Mirzakuchaki, S. (2023). Dilated-UNet: A Fast and Accurate Medical Image Segmentation Approach Using a Dilated Transformer and U-Net Architecture, School of Electrical Engineering, Iran University of Science and Technology, Tehran, Iran.

[24] Arulananth, T. S. et al. (2024). Semantic Segmentation of Urban Environments: Leveraging U-Net Deep Learning Model for Cityscape Image Analysis, PLOS ONE, doi: 10.1371/journal.pone.0300767.

[25] Zhao, H., Shi, J., Qi, X., Wang, X., and Jia, J. (2017). Pyramid Scene Parsing Network, in Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, (pp. 2881–2890).

[26] Badrinarayanan, V., Kendall, A., and Cipolla, R. (2017). SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, (pp. 2481–2495).

[27] Romera, E., Alvarez, J. M., Bergasa, L. M., and Arroyo, R. (2017). ErfNet: Efficient Residual Factorized ConvNet for Real-Time Semantic Segmentation, IEEE Transactions on Intelligent Transportation Systems, vol. 19, (pp. 263–272).

[28] Siddique, N. et al. (2024). U-Net and Its Variants for Medical Image Segmentation: A Review of Theory and Applications, IEEE Access, vol. 9, (pp. 82031–82057).

[29] Mehta, S. et al. (2018). ESPNet: Efficient Spatial Pyramid of Dilated Convolutions for Semantic Segmentation, in Proc. European Conference on Computer Vision (ECCV), Munich, Germany, (pp. 552–568).

[30] Fantauzzo, L. et al. (2022). FedDrive: Generalizing Federated Learning to Semantic Segmentation in Autonomous Driving, in Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), (pp. 11504–11511), doi: 10.1109/IROS47612.2022.9981098.

[31] Paszke, A., Chaurasia, A., Kim, S., et al. (2016). ENet: A Deep Neural Network Architecture for Real-Time Semantic Segmentation, arXiv preprint, arXiv:1606.02147.

[32] Bhattacharyya, P., Mishra, N., Nair, P. A., et al. (2019). Fast-SCNN: Fast Semantic Segmentation Network, in Proc. IEEE International Conference on Computer Vision Workshops (ICCVW), (pp. 1972–1978).

[33] Jordan, J. (2018). Evaluating Image Segmentation Models. [Online]. Available: https://www.jeremyjordan.me/evaluating-image-segmentation-models/. [Accessed: Jan. 17, 2025].

[34] Daniella, D. (2024). Semantic Segmentation in AI, Principle and Applications. [Online]. Available: https://en.innovatiana.com/post/semantic-segmentation-in-ai. [Accessed: Jan. 17, 2025].

[35] Long, J., Shelhamer, E., and Darrell, T. (2015). Fully Convolutional Networks for Semantic Segmentation, Proc. IEEE Conf. Comput. Vis. Pattern Recognit., (pp. 3431–3440), doi: 10.1109/CVPR.2015.7298965.

[36] Badrinarayanan, V., Kendall, A., and Cipolla, R. (2017). SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation, IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 12, (pp. 2481–2495), doi: 10.1109/TPAMI.2016.2644615.

[37] Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., and Adam, H. (2017). Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation, in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), (pp. 833–841), doi: 10.1109/CVPR.2017.112.

[38] Zhou, Z., Siddiquee, M., Tajbakhsh, N., and Liang, J. (2018). UNet++: A Nested U-Net Architecture for Medical Image Segmentation, in Proc. Int. Conf. Med. Image Comput. Comput. Assist. Interv. (MICCAI), (pp. 3–11), doi: 10.1007/978-3-030-00889-5_1.