OffroadNet

Abhay Chhagan Karade [1] **Denny Boby** [1] **Sumukh Sreenivasarao Balakrishna** [1] **Shreedhar Kodate** [1]

Abstract

OffroadNet is a collection of model architectures that focus on the task of improving performance of traversable path detection in unpaved roads. We present the results of our experiments on the Robot Unstructured Ground Driving (RUGD) Dataset and train models to specifically focus on path detection performance. We use mIoU and mAcc as our metrics to compare the performance of different models. OffroadNet (based on PSPNet) and ResNet101 is one of the best combination for the offroad path detection because of it's PPM module and fine tuning based on the Cityscape dataset.

(https://github.com/AbhayKarade/offroad_net)

1. Introduction

The latest advances in Deep Learning, Computer Vision, Sensor Technologies like LiDAR have facilitated the great advances in building the Autonomous Vehicles. Companies like Tesla have achieved level 3 autonomy and are working on incorporating autonomy levels 4 and 5. To head towards level 5 autonomy which is defined by SAE (Society of Automotive Engineers) as "same as level 4, but features can drive everywhere in all conditions" (International, 2021) to reach this, cars need to understand how to traverse on unpaved and unmarked roads.

The applications of autonomous vehicles in urban roads is a miniature area for application, the other challenging areas to deploy the autonomous vehicles are agricultural applications, transport of goods through hilly areas, village roads, wildland firefighting, construction sites, mining operations, detection of land mines in conflicted zones and other military applications. On urban road the autonomous vehicles are matter of comfort and ease where as on the above mentions applications it could have direct impact on human lives and big impact on logistics and agricultural development.

In this work, our aim is to parse images of RUGD dataset which is collected with outdoor robot and interpret them at the pixel level in order to inform navigation decisions and to be able to identify safe and suitable terrain to drive on. We are building on MMSegmentation Github repository (Contributors, 2020) by OpenMMLabs here we are experimenting with different architectural combinations of backbone, decodehead and auxiliary head. Previously researchers addressed this problem statement with the segmentation of 20 different classes and defining a 3-level affordance to characterize the drivability as, 'Preferable', 'Possible but not preferable' and 'Impossible or undesirable' respectively (Humblot-Renaux et al., 2022). However in our research we are only mapping traversable terrain which is preferable. We aimed to make the architecture as light as possible so that it can be deployed and used in real-time applications. We are using pre-trained weights by OpenMMlabs and feeding pre-processed dataset with binary labels for transfer learning this eluded the training of model from scratch and saved computation cost and time.

Figure 1. Complex and dangerous offroad paths. Mining, Agriculture, Armed Forces, Natural calamities.

1.1. Research contributions

Our work tries to bring out the main ideas of different state-of-the-art architectures for semantic segmentation and distill the focus of these architectures to improve the offroad path detection. We use MMSegmentation (mmseg) a well-structured framework to train models for our offroad path

[1]Worcester Polytechnic Institute. Correspondence to: Abhay Chhagan Karade, Denny Boby, Sumukh Sreenivasarao Balakrishna, Shreedhar Kodate <akarade, dboby, ssreenivasaraoba, sskodate@wpi.edu>.

CS/DS 541 Deep Learning Final Project, Worcester Polytechnic Institute.

segmentation task. We contribute the experiments, configurations, and pretrained weights for the RUGD dataset which can help further studies to improve offroad prediction and compare it with state-of-the-art models.

2. Related Work

2.1. GANav: Efficient Terrain Segmentation for Robot Navigation in Unstructured Outdoor Environments

(Guan et al., 2021) They proposed a novel group-wise attention mechanism to identify safe and navigable regions in off-road terrains and unstructured environments from RGB images using RUGD (Wigness et al., 2019a) and RELLIS-3D (Jiang et al., 2020) combined datasets . Their approach classifies terrains based on their navigability levels using coarse-grained semantic segmentation. Their novel group-wise attention loss enables any backbone network to explicitly focus on the different group's features with low spatial resolution. The input consists of an RGB image $I \in R^{3*H*W}$ and the corresponding ground-truth semantic segmentation labels $Y \in Z^{H*W}$ denoting the category to which each pixel belongs among G different groups. They used new coarse-grain labels $Y_G \in Z^{H*W}$ based on semantic labels provided by the dataset. For each group, they also computed the binary mask $Y_{Bg} \in \{0,1\}^(H*W)$.

Their novel architecture fuses a multi-scale feature extractor with a transformer architecture.Their group-wise segmentation head can fuse visual features from different scales and explicitly focus on different terrain types, which leads to better accuracy on different surfaces of varying areas. Their approach uses a transformer-based backbone architecture and leverages the MHSA(multi-head self attention) module to extract and fuse multi-scale features. In particular, they used a Mixed Transformer with some modifications as their backbone. Given an input feature vector,the output self attention is computed as follows

$$A_{out} = Softmax(K(A_{in})^T \bigodot q(A_{in})^T \bigodot V(A_{in}))$$

Where k(A_{in}), q(A_{in}), V(A_{in})represent key, query and value feature maps. k,q and v are linear projections as in the self-attention literature. The feature map is reshaped,transposed and flattened. The flattened output passed through the MHSA block with G attention heads.The MHSA component fuses flatten multi-scale feature to produce the new feature maps $F_{out} \in R^{(C_{out}*H_f*W_f)}$ and generates G attention maps (one for each group), $A_1, A_2, ...A_G \in [0,1]^{H_f W_f * H_f W_f}$ The final output $P \in R^{(G*H*W)}$ is obtained through a standard procedure of a segmentation network by a series of 1 × 1 convolutions and up-sampling from F_{out}

They used those attention maps as an additional branch in the detection head and trained the detection head to resemble the group distribution by explicitly guiding each attention map towards a corresponding category using a binary cross-entropy loss.

For each attention head h_i,they have its corresponding attention map $A_i \in [0,1]^{L*L}$ where $L = H_f * W_f$ using bi-linear image resizing. Each pixel in B_i represents represents the self-attention score with respect to h_i To guide each attention-map in the multi-head self-attention module, they applied a binary cross-entropy loss function:

$L^g_{GA} = -\sum_{h,w} y_G log(B_g)$ Where $y_G \in Y_G$.

2.2. Trseg

(Jin et al., 2021) Unlike existing methods that capture multi-scale contextual information through infusing every single-scale piece of information from parallel paths. They proposed a novel semantic segmentation network incorporating a transformer to adaptively capture multi-scale information with the dependencies on original contextual information. Given the original contextual information as keys and values, the multi-scale contextual information from the multi-scale pooling module as queries is transformed by the transformation decoder.

The overall transformer architecture is an encoder-decoder architecture. However, as the goal of the encoder is extracting and retaining contextual information from input sequence, the encoder is skipped in their model because the backbone CNN produces in-formative contexts. The single decoder layer is composed of a self-attention block, an encoder-decoder attention block and a Feed-Forward Network (FFN). The overall decoder part consists of a stack of several identical transformer decoder layers. The attention networks are computed as:

$$Attention(Q,K,V) = Softmax(QK^T/\sqrt(d_k))$$

Where Q,K,V and dk are queries, values, keys and dimension of keys.Correlations of all pairs between queries Q and keys K are computed and divided by $\sqrt(d_k)$ to have stable gradients. Then a softmax function is adapted to obtain the weights on the values V to produce the output of an attention network. All attention blocks in their model perform multi-head attention with 8 heads. Following the attention networks in the decoder is a point-wise FFN. It consists of two linear transformation layers, LayerNorm and ReLU activation. They further process the elements of the attended outputs individually, using different parameters from layer to layer.

2.3. A RUGD Dataset for Autonomous Navigation and Visual Perception in Unstructured Outdoor Environments

(Wigness et al., 2019b) They selected a number of state-of-the-art semantic segmentation approaches and trained/evaluated them on the RUGD dataset. For each semantic segmentation approach they used a fixed encoder, ResNet50 (He et al., 2015), with and without the dilated convolution of size 8. Thus, the baseline approaches differ in their decoder portion of the network. They selected Pyramid Scene Parsing Network (PSPNet)) and Unified Perceptual Parsing Network (UPerNet) (Xiao et al., 2018).s. PPM pools features in several different pyramid scales, upsamples them using bilinear interpolation to match the original feature map. These maps are then concatenated to be fed into a convolutional layer which makes the final prediction map. UPerNet approach is based on the Feature Pyramid Network (FPN) (Lin et al., 2016) which is devised to efficiently learn and make use of semantically strong, multi-scale features. This top-down architecture with lateral connections creates high-level feature maps across all scales. In UperNet, FPN and PPM function together as PPM is applied on the last layer of the backbone network, before the feature is fed into the top-down hierarchy in FPN. Note that the dilated convolution which has become the de facto model for semantic segmentation was omitted in UperNet as it presents several drawbacks including computational complexity. For PPM in their UPerNet, they used down-sampling rate of 4.

3. Proposed Method

Inspired by the GANnav research work, we have used MMSegmentation framework to train our segmentation models. Our main focus is to evaluate different architectures for training on the RUGD dataset. We use pre-trained weights of the various sub-architectures trained on baseline datasets like CityScapes, and ADE20K. We modify the dataset for binary class prediction and train models for multi-class as well as binary class segmentation with different hyper-parameters. We collected some test examples from the internet, and manually check the inference performance.

3.1. MMSegmentation

The framework has enabled researchers to implement their custom models and datasets specifically for the semantic segmentation task. They have also included the pre-trained weights for the state-of-the-art model architectures which have been optimized for various datasets. The framework have designed a flexible yet structured way to build new architectures for the segmentation tasks.

The whole model architecture is broken down into sub-architectures, explained as following: **Backbone** is the feature engineering for the input image. e.g. ResNet architecture. **Neck** acts as an additional feature representation that might be useful for extracting information. e.g. MultilevelNeck. **DecodeHead**'s task is to compute the pixel-wise classification based on the feature matrices. e.g. PSPHead. **AuxiliaryHead**, the final sub-architecture design before segmentation. e.g. FCN.

The detailed steps of implementing new models, architectures, configurations, datasets, are explained in the MMSeg documentation.

3.2. PSPNet

In the ImageNet 2016, Scene Segmentation Challenge (Zhao et al., 2016). proposed the winning model, Pyramid Scene Parsing Network (PSPNet). Although CNN-based methods are widely used for image datasets, for the segmentation task, they have a few drawbacks, like the size of the block around a pixel of interest often affects the size of the sensing area, and the segmentation time is slow and inaccurate. Fully Convolutional Network (FCN) improves the segmentation accuracy by recovering the pixel locations responsible for the features at different layers. But, FCN ignores the relationship between pixels, and hence, lacks the spatial consistency. PSPNet overcomes these limitations by implementing a Pyramid Pooling Module (PPM) which captures the global and local features contextually in the image. Due to these reasons, we select PSPNet for the task of Offroad path detection. It is important to understand the contextual features in and around the traversable path so that it can be used in further motion planning pipelines (Zhu et al., 2021).

3.3. ENCNet (Context Encoding for Semantic Segmentation)

ENCNet-101 is yet another model which achieved a score of 0.5567 surpassing the winning entry of COCO-Place Challenge in 2017 (Zhang et al.). Fully Convolutional Neural Network (FCN) pioneered the era of endto-end segmentation. However, recovering detailed information from downsampled featuremaps is difficult due to the use of pre-trained networks that are originally designed for image classification. To address this difficulty, one way is to learn the upsampling filters, i.e. fractionally-strided convolution or decoders. The other path is to employ Atrous/Dilated convolution strategy to the network which preserves the large receptive field and produces dense predictions. Prior work adopts dense CRF taking FCN outputs to refine the segmentation boundaries, and CRFRNN achieves end-to-end learning of CRF with FCN. Recent FCN-based work dramatically boosts performance by in-

creasing the receptive field with larger rate atrous convolution or global/pyramid pooling. However, these strategies have to sacrifice the efficiency of the model, for example PSPNet applies convolutions on flat featuremaps after Pyramid Pooling and upsampling and DeepLab employs large rate atrous convolution that will degenerate to 1 × 1 convolution in extreme cases. We propose the Context Encoding Module to efficiently leverage global context for semantic segmentation, which only requires marginal extra computation costs. In addition, the proposed Context Encoding Module as a simple CNN unit is compatible with all existing FCN-based approaches. The proposed Context Encoding Network achieves state-of-the-art results 85.9% mIoU on PASCAL VOC 2012 and 51.7% on PASCAL in Context.

3.4. ViT Transformer with UperNet DecoderHead

Although convolutional models are widely used as backbone because of faster learning Vision Transformer (ViT) achieves remarkable results compared to convolutional neural networks (CNN) while obtaining fewer computational resources for pre-training. In comparison to convolutional neural networks (CNN), Vision Transformer (ViT) show a generally weaker inductive bias resulting in increased reliance on model regularization or data augmentation (AugReg) when training on smaller datasets. We used UPerNet as decoder because it demonstrates best performance on other large-scale semantic segmentation benchmark datasets such as ADE20K and Cityscapes.

4. Experiment

4.1. Datasets

4.1.1. RUGD

The RUGD paper (Wigness et al., 2019a) explains details about the dataset. It has 20 classes for semantic segmentation. The dataset is representative of the dataset required for the training a robust offroad path detection model. We have worked on two versions of the RUGD dataset: **Multi class semantic segmentation**: Include all the classes and train a semantic segmentation model. However, focus only on improving the performance of classes of interest viz. gravel, asphalt, concrete, and rockbed. These are selected after observing the dataset for traversable paths. **Single class or Binary segmentation:** Include only relevant class (e.g. gravel) as the Path class and rest all as background. We theorized that if the model performs better on this modified binary segmentation dataset, then we might prune the model to get the final model as efficient and as accurate as possible for the offroad path detection.

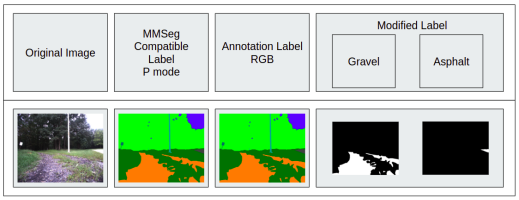

Figure 2. RUGD Dataset: The annotation labels provided by authors had to be modified to palletized labels (mode p). Formed the new binary annotations for individual classes of interest.

4.1.2. YamahaCMUDataset

The YamahaCMUDataset (Maturana et al., 2018) paper explains details about the dataset. It consists of 1076 images collected in four different locations in Western Pennsylvania and Ohio, spanning three different seasons.The dataset was labelled using a polygon-based interface with eight classes: sky, rough trail, smooth trail, traversable grass, high vegetation, non-traversable, low vegetation and obstacle.

4.2. Training

In this project, we carried out our training using the MMSegmentation library. The models were trained on WPI Turing clusters (WPI, 2018) with 1 CPU, 1 Tesla P100 GPU, and 80 GB memory. Because, we used transfer learning to train all our models, and the dataset size was small (about 5GB), the models achieved high mIoU and mAcc within 3 hours. We trained each model architecture on both datasets (Binary class dataset and Multi-class dataset) for about 3 hours.

Following models were trained: PSPNet with ResNet 50 as backbone (Model 1), PSPNet with ResNet 101 as backbone (Model 2), ENCNet with ResNet 101 as backbone (Model 3).

The dataset's images were split into 5947 images for training and 1487 images for validation. These images were loaded and trained for 1200 epochs and 10000 epochs to evaluate the accuracy change and we saved the best model. The evaluation period were 200 and 2000 epochs for the above mentioned training. The type of optimizer used is SGD with learning rate of 0.01, momentum of 0.09 and weight decay of 0.0005.

4.3. Evaluation

After training the models the inference was made on the validation set. We also got some offroad diving videos from youtube and split them into images and added it to validation set for seeing the models inference on images that it has never seen before. The evaluation metrics used

Backbone	Method	mIoU
ResNet 50	Upsampling	32.24
ResNet 50	PSPNet	31.78
ResNet 50	UperNet	31.95
ResNet 50	Deeplabv3	32.81
ResNet 50	TrSeg	33.91
ResNet 50	OffRoadNet(PSPNet)	**42.03**
ResNet 101	PSPNet	**42.76**
ResNet 101	ENCNet	37.3

Table 1. The table shows the comparison of our model to various state-of-the-art model in terms of overall class mIoU.

is mIoU (mean Intersection-Over-Union) and mAcc (mean Accuracy). mIoU is the mean 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. The global accuracy measure (gAcc) counts the number of pixels correctly classified. The gAcc metric counts each pixel equally, so the results of the long-tailed categories have little impact on the metric number. The mean accuracy (mAcc) metric mitigates this by normalizing within each category.

5. Results

We observe that semantic segmentation helps offroad path detection and is better than doing a binary class segmentation. We also observe that the proposed model architectures are highly complex and easily over-fit the small RUGD dataset. So, after training by transfer learning the ResNet backbone with OffroadNet (based on PSPNet) we get a higher mIoU on the validation set.

6. Discussion

Availability of dataset for varied terrains, lighting conditions, seasonal weather, natural calamities, etc. is hard to collect. Detecting a traversable path in wilderness is a challenging task and needs much more training dataset which can be collected and labelled by crowd sourcing.

7. Conclusions and Future Work

Training models based on Vision Transformer (ViT) as a backbone and ENCNet as decode head might . Incorporating more data based on different traversable terrains. There is a lack of dataset like mining fields, forest fire, agricultural terrains, which can be collected and train models on those. The dataset can varied and more unpredictable for diverse offroad conditions based on seasonal weather, light conditions, natural calamities, etc. We plan to refine our work, configurations, and models and submit to MMSegmentation GitHub repository to help further studies on offroad path detection.

8. Acknowledgement

We would like to express our gratitude to the CS/DS 541 instructor, Prof. Jacob Whitehill for conducting enthusiastic lectures on Deep Learning. We would like thank WPI for their HPC Turing clusters without which training our models would have been much more tedious.

References

Contributors, MMSegmentation. MMSegmentation: Openmmlab semantic segmentation toolbox and benchmark. https://github.com/open-mmlab/mmsegmentation, 2020.

Guan, Tianrui, Kothandaraman, Divya, Chandra, Rohan, Sathyamoorthy, Adarsh Jagan, Weerakoon, Kasun, and Manocha, Dinesh. Ganav: Efficient terrain segmentation for robot navigation in unstructured outdoor environments, 2021. URL https://arxiv.org/abs/2103.04233.

He, Kaiming, Zhang, Xiangyu, Ren, Shaoqing, and Sun, Jian. Deep residual learning for image recognition, 2015. URL https://arxiv.org/abs/1512.03385.

Humblot-Renaux, Galadrielle, Marchegiani, Letizia, Moeslund, Thomas B., and Gade, Rikke. Navigation-oriented scene understanding for robotic autonomy: Learning to segment driveability in egocentric images. *IEEE Robotics and Automation Letters*, 7(2):2913–2920, apr 2022. doi: 10.1109/lra.2022.3144491. URL https://doi.org/10.1109%2Flra.2022.3144491.

International, SAE. Sae levels of driving automation, 2021. URL https://www.sae.org/blog/sae-j3016-update.

Jiang, Peng, Osteen, Philip, Wigness, Maggie, and Saripalli, Srikanth. Rellis-3d dataset: Data, benchmarks and analysis, 2020.

Jin, Youngsaeng, Han, David, and Ko, Hanseok. Trseg: Transformer for semantic segmentation. *Pattern Recognition Letters*, 148:29–35, 2021.

Lin, Tsung-Yi, Dollár, Piotr, Girshick, Ross, He, Kaiming, Hariharan, Bharath, and Belongie, Serge. Feature pyramid networks for object detection, 2016. URL https://arxiv.org/abs/1612.03144.

Maturana, Daniel, Chou, Po-Wei, Uenoyama, Masashi, and Scherer, Sebastian. Real-time semantic mapping for autonomous off-road navigation. In *Field and Service Robotics*, pp. 335–350. Springer, 2018.

Wigness, Maggie, Eum, Sungmin, Rogers, John G, Han, David, and Kwon, Heesung. A rugd dataset for autonomous navigation and visual perception in unstructured outdoor environments. In *International Conference on Intelligent Robots and Systems (IROS)*, 2019a.

Wigness, Maggie, Eum, Sungmin, Rogers, John G., Han, David, and Kwon, Heesung. A rugd dataset for autonomous navigation and visual perception in unstructured outdoor environments. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5000–5007, 2019b. doi: 10.1109/IROS40897.2019.8968283.

WPI. Wpi turing cluster, 2018. URL https://arc.wpi.edu/computing/hpc-clusters/.

Xiao, Tete, Liu, Yingcheng, Zhou, Bolei, Jiang, Yuning, and Sun, Jian. Unified perceptual parsing for scene understanding, 2018. URL https://arxiv.org/abs/1807.10221.

Zhang, Hang, Dana, Kristin, Shi, Jianping, Zhang, Zhongyue, Wang, Xiaogang, Tyagi, Ambrish, and Agrawal, Amit. Context encoding for semantic segmentation. URL https://arxiv.org/abs/1803.08904.

Zhao, Hengshuang, Shi, Jianping, Qi, Xiaojuan, Wang, Xiaogang, and Jia, Jiaya. Pyramid scene parsing network, 2016. URL https://arxiv.org/abs/1612.01105.

Zhu, Xiliang, Cheng, Zhaoyun, Wang, Sheng, Chen, Xianjie, and Lu, Guoqing. Coronary angiography image segmentation based on pspnet. *Computer Methods and Programs in Biomedicine*, 200:105897, 2021. ISSN 0169-2607. doi: https://doi.org/10.1016/j.cmpb.2020.105897. URL https://www.sciencedirect.com/science/article/pii/S0169260720317302.

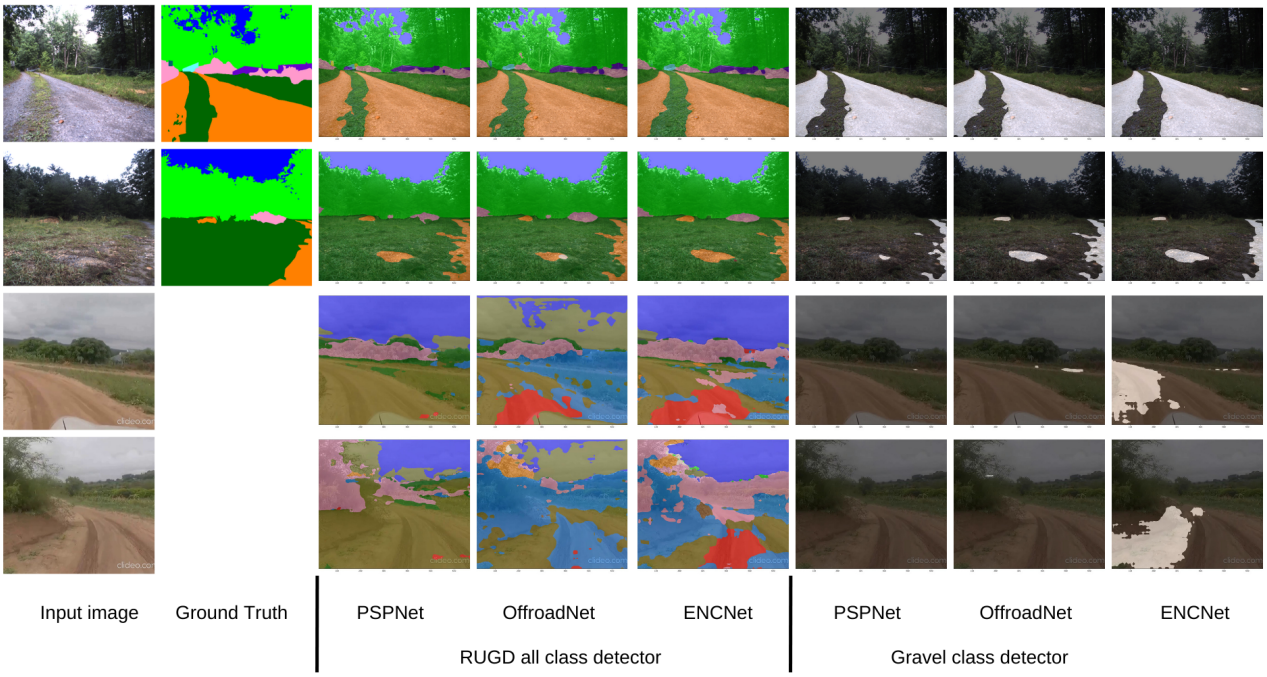

Figure 3. Inference on validation dataset (first two images from top). Inference on random images from internet (bottom two). PSPNet performs much better when used on multi-class semantic segmentation, while ENCNet seems to better for the binary segmentation. Although all the models in general perform worse for the binary segmentation.

Backbone	DecodeHead	AuxillaryHead	Metrics	A	G	R	C	O	OG
ResNetV1c 50	PSPHead	FCNHead	mIoU	**92.52**	**82.04**	–	0.96	42.04	80.73
			mAcc	**95.07**	**92.02**	–	0.97	52.21	87.04
ResNetV1c 101	OffRoadHead	FCNHead	mIoU	91.99	81.38	–	**1.6**	**42.76**	**82.18**
			mAcc	94.4	91.73	–	**1.78**	**52.61**	88.73
ResNetV1c 101	EncHead	FCNHead	mIoU	91.26	77.55	–	0.72	37.3	80.09
			mAcc	94.3	90.63	–	0.73	49.29	**90.18**

Table 2. The table shows the results for various models on RUGD dataset based on the mean IoU, and mean Accuracy. The OnlyGravel class is performance on the binary segmentation task. Since Rock-bed is not present in validation set its values are '–'. A: Asphalt, G: Gravel, R: Rock-bed, C: Concrete, O: Over all classes, OG: OnlyGravel

Backbone	DecodeHead	AuxillaryHead	Metrics	TG	RT	ST	O
ResNetV1c 50	PSPHead	FCNHead	mIoU	77.18	64.72	49.46	**47.79**
			mAcc	93.74	80.12	72.21	**58.39**
ResNetV1c 101	OffRoadHead	FCNHead	mIoU	78.06	65.38	**50.45**	47.29
			mAcc	**94.97**	84.87	**72.26**	57.55
ResNetV1c 101	EncHead	FCNHead	mIoU	**78.32**	**66.16**	44.19	46.52
			mAcc	94.59	**87.09**	63.62	57.0

Table 3. The table shows the results for various models on Yamaha-CMU Off-Road dataset based on the mean IoU, and mean Accuracy. TG: Traversable grass, RT: Rough trail, ST: Smooth trail, O: Over all classes