Monocular Vision-Based Traversability Estimation for Off-road Navigation

Thabsheer Machingal
thabsheerjm1@gmail.com

***Abstract*—The ability to perceive and understand terrain traversability is paramount for the safe and efficient navigation of unstructured, off-road environments. In this work, we present a real-time system designed to generate a traversability map of the off-road environment using only monocular RGB images. We leverage the GOOSE dataset, which provides aligned RGB images and semantic masks, to train and fine-tune a DeepLabv3+ segmentation model with a MobileNet backbone. Our focus is on binary segmentation of traversable versus non-traversable terrain, derived from a curated set of semantic classes. We construct a PyTorch training pipeline with custom dataset handling, on-the-fly data filtering, and validation-based checkpointing. The final model is exported to ONNX and integrated into a real-time C++ semantic segmentation system. This pipeline enables scalable, sensor-efficient terrain understanding for autonomous ground vehicles (AGVs) operating in complex natural environments, offering a step toward low-cost, vision-based off-road navigation system.**

***Index Terms*—Navigation, Semantic, AGV, ONNX**

I. Introduction

Navigating unstructured and off-road environments remains a fundamental challenge in autonomous ground vehicle (AGV) systems. Traditional path planning and perception pipelines developed for structured urban roads often rely on high-definition maps, LiDARs, and GPS data[1], which are either unavailable or unreliable in remote or natural terrains. Furthermore, sensor fusion steps increase cost, complexity, and power demands, making them less suitable for scalable, lightweight AGVs operating in the field.

Recent research has shown that monocular vision-based systems, particularly semantic segmentation models, can be leveraged for real-time terrain understanding [2], [3]. Such models classify each pixel into semantic classes that can be abstracted into traversability categories. However, many existing classes are either too large for real-time inference on embedded systems or lack sufficient generalization when trained only on urban dataset like Cityscapes[4].

In this work, we propose a production-ready, real-time off-road traversability mapping system that utilizes only monocular RGB input. Our pipeline is built around that DeepLabv3+ architecture[5] with a MobileNetv3 backbone[6], offering a balance between segmentation accuracy and computational efficiency. Unlike Prior works that focus on theoretical or simulation results, we emphasize a deployable software stack using ONNX Runtime for C++ inference, optimized for integration into AGV platforms.

To ensure high-qualify segmentation in unstructured terrains, we fine tune our model using the GOOSE dataset[7], a recent contribution that provides RGB, NIR and semantic masks collected from diverse natural environments. While prior datasets like Mapillary vistas[8] or ADE20K[9] offers broad semantic classes, they fall short in representing the real-world off-road challenges such as vegetation, uneven terrain, and mud.

We frame the segmentation problem as a binary classification task: traversable vs. non-traversable. This abstraction enables efficient decision-making in downstream planning modules. Furthermore, our pipeline includes mechanisms for custom dataset parsing, real-time image processing, dynamic training-validation splitting, validation-based checkpointing; features essential for robust model development and reproducibility.

In contrast to works like [10], which explore multi-model or different sensor approaches involving LiDAR and RGB fusion (for example), our system proves that monocular vision alone, when paired with optimized deep learning models, can deliver actionable terrain understanding in real-time.

By integrating the trained model into a C++-based real-time inference pipeline using ONNX Runtime, we bridge the gap between research and deployment, delivering a scalable software stack for AGV navigation in complex natural environments. This paper presents a complete system from training to deployment and evaluates its performance on real-world off-road scenes.

II. Methods

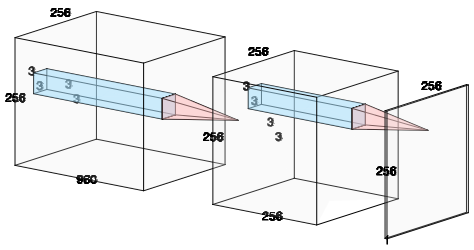

Fig. 1. Architecture of the DeepLabV3+ custom head used for binary segmentation.

A. Dataset

We utilize the GOOSE (**G**erman **O**utdoor and **O**ffroad Data**se**t) dataset[7], as recent off-road benchmark featuring aligned RGB, NIR and semantic segmentation masks collected in real world unstructured environments. For our use, we strictly used Monocular RGB images and semantic labels. The original dataset includes over 20 semantic classes, many of which are not relevant to the context of terrain traversability. To simplify the problem, we trained a binary segmentation model. Classes such as asphalt, gravel, grass, bikeway, and sidewalk are mapped as traversable, while others like vegetation, rock, water, and building are treated as non-traversable.

We created a custom PyTorch Dataset class that:

- Filters and loads image-mask pairs.
- Resizes input images to a uniform shape(520,520).
- Dynamically converts the multiclass semantic mask into a binary traversability mask using a label mapping CSV.
- Applies image augmentation and preprocessing transforms.

B. Network Architecture

We utilized the DeepLabv3+ architecture with a MobileNetV3-Large backbone, selcted for its trade-off between segmentation accuracy and computational efficiency, an essential consideration for real-time deployment in resource-constracriterionined environments. To tailor the model to our binary classification task, we replace the default classifier head with a lightweight custom decoder (Fig. 1). This cusotm head reduces the model's parameter footprint while maintaining sufficient representational capacity to differentiate traversable and non-traversable terrain. The network produces a single-channel output representing the traversability likelihood per pixel. We apply a binary cross-entropy loss with logits during training, and use a sigmoid activation during inference to generate threshold binary masks.

C. Training & Evaluation

Training pipeline is fully configurable and modular. All parameters, including data paths, hyperparameters, and checkpoint locations, are specified via a structured JSON configuration file to ensure reproducibility and flexibilty.

The dataser is partitioned into training and validation subsets with a fixed seed to maintain consistency across experiments. To monitor generalization performance, we compute two primary evaluation metrics:

- **Pixel-wise Accuracy**, indicate the percentage of correctly classified pixels.
- **Intersection over Union(IoU)**, measures the overlap betweem predicted and ground truth traversable regions.

We used a **ReduceLROnPlateau** scheduler to adaptively adjust the learning rate based on validation loss, and apply early stopping if no improvement in IoU is observed for a fixed number of epochs. The model achieving the highest validation IoU is persisted for deployment.

D. Deployment & Real-Time Inference

To enable deployment in production and embedded settings, we export the trained PyTorch to ONNX format. This facilitates fast and portable inference in C++ using ONNX Runtime in conjunction with OPENCV for processing and visualization.

The inference system consists of the following components:

- **Preprocessing**: Input images(RGB) are resized, normalized, and converted to a contiguous CHW tensor suitable for ONNX inference.
- **Model Execution**: The ONNX model is loaded into an *Ort::Session*, and inference is performed via *Ort::Value* objects for I/O tensors.
- **PostProcessing**: The model output is passed through a sigmoid activation, followed by thresholding to produce a binary segmentation mask.
- **Visualization**: The resulting mask is overlaid on the original image to aid interpretability and debugging.

The full pipeline is compiled into a single standalone executable and distributed as **.deb** package for seamless installation on Debian-based systems.

1) Jetson Deployment: To ensure compatibility with NVIDIA Jetson platforms(e.g., Xavier, Nano), the model is exported in FP16 precision to leverage TensorRT acceleration. The C++ inference code is cross-compiled for ARM64 architecture, and JetPack SDK provides all required runtime libraries(OpenCV, ONNX Runtime, CUDA/cuDNN). This makes the system fully deployable on edge devices, enabling real-time semantic segmentation in the field without reliance on external compute.

CONCLUSION

We presented a complete, lightweight semantic segmentation system for off-road traversability estimation using monocular RGB camera. By leveraging the DeepLabv3+ architecture with a custom head and training on the GOOSE dataset, we achieved accurate binary segmentation suitable for real-time use. The trained model was exported to ONNX, integrated into C++ inference pipeline, and successfully packaged as a .deb installer. The final release has been successfully tested on a debian based system, demonstrating practical readiness for deployment. The project is now fully functional, and the release version is available for integration into autonomous navigation stacks.

REFERENCES

[1] Thrun, S., Montemerlo, M., et al. "Stanley: The robot that won the DARPA Grand Challenge." Journal of Field Robotics, 2006.

[2] A. Valada, R. Mohan, and W. Burgard, "Self-Supervised Model Adaptation for Multimodal Semantic Segmentation," 2018, doi: 10.48550/ARXIV.1808.03833.

[3] G. Mattyus, W. Luo, and R. Urtasun, "DeepRoadMapper: Extracting Road Topology from Aerial Images," in 2017 IEEE International Conference on Computer Vision (ICCV), Venice: IEEE, Oct. 2017, pp. 3458–3466. doi: 10.1109/ICCV.2017.372.

[4] M. Cordts et al., "The Cityscapes Dataset for Semantic Urban Scene Understanding," 2016, arXiv. doi: 10.48550/ARXIV.1604.01685.

[5] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation," 2018, arXiv. doi: 10.48550/ARXIV.1802.02611.

[6] A. Howard et al., “Searching for MobileNetV3,” 2019, arXiv. doi: 10.48550/ARXIV.1905.02244.
[7] P. Mortimer, R. Hagmanns, M. Granero, T. Luettel, J. Petereit, and H.-J. Wuensche, “The GOOSE Dataset for Perception in Unstructured Environments,” 2023, arXiv. doi: 10.48550/ARXIV.2310.16788.
[8] G. Neuhold, T. Ollmann, S. R. Bulo, and P. Kontschieder, “The Mapillary Vistas Dataset for Semantic Understanding of Street Scenes,” in 2017 IEEE International Conference on Computer Vision (ICCV), Venice: IEEE, Oct. 2017, pp. 5000–5009. doi: 10.1109/ICCV.2017.534.
[9] B. Zhou et al., “Semantic Understanding of Scenes through the ADE20K Dataset,” 2016, arXiv. doi: 10.48550/ARXIV.1608.05442.
[10] P. Jiang et al., “GO: The Great Outdoors Multimodal Dataset,” 2025, arXiv. doi: 10.48550/ARXIV.2501.19274.