2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

TerrainSense: Vision-Driven Mapless Navigation for Unstructured Off-Road Environments

Bilal Hassan[1,2], Arjun Sharma[1,2], Nadya Abdel Madjid[1,2], Majid Khonji[1,2], and Jorge Dias[1,2]

***Abstract*— Navigating autonomous vehicles efficiently across unstructured and off-road terrains remains a formidable challenge, often requiring intricate mapping or multi-step pipelines. However, these conventional approaches struggle to adapt to dynamic environments. This paper presents TerrainSense, an end-to-end framework that overcomes these limitations. By utilizing a transformers, TerrainSense detects lane semantics and topology from camera images, enabling mapless path planning without the reliance on highly detailed maps. The efficacy of TerrainSense was rigorously assessed on six diverse datasets, evaluating its efficacy in detection, segmentation, and path prediction using various metrics. Notably, it outperforms the other state-of-the-art methods by 9.32% in precisely predicting the path with 18.28% faster inference time.**

I. Introduction

Navigating autonomous vehicles through unstructured and off-road terrains represents a crucial challenge in the expanding frontier of autonomous driving [1], [2], [3]. Off-road environments are characterized by their lack of structured pathways and can include a variety of natural landscapes such as forests, deserts, and mountainous areas. These terrains are unpredictable and dynamic, presenting obstacles like uneven ground, foliage, and absence of clear road markings. Traditional methods, which often rely on pre-existing maps and multi-stage pipelines, find themselves ill-equipped in addressing the dynamic and unpredictable nature of off-road landscapes [4]. These strategies, although effective in structured urban terrains, grapple with the rapidly changing off-road environments, especially when high-fidelity maps are absent or outdated [5], [6].

In the broader context of autonomous navigation, various strategies have been examined. These range from rule-based algorithms to intricate multi-step pipelines and the emerging deep learning-powered techniques [7], [8]. The inherent rigidity of rule-based algorithms limits their adaptability to evolving terrains, whereas multi-stage pipelines, with their sequential components of mapping, localization, and planning, might fall short in terms of real-time responsiveness, particularly in unstructured environments [9], [10], [11].

Deep learning has emerged as a transformative force across various fields, introducing avenues for innovative and efficient solutions [12], [13], [14]. Specifically, in the realm of autonomous driving, it serves as a game-changer for enabling efficient and adaptive navigation [15]. Predominantly, these learning-based techniques have shown tremendous promise in on-road navigation scenarios, harnessing vast data sets to evolve and adapt [16], [17], [18]. However, their efficiency dwindles when presented with off-road terrains where the luxury of detailed maps is often lacking [4], [19].

While urban terrains benefit from resources such as high-definition (HD) maps [20], OpenStreetMap (OSM) data [21], and an array of sensors like lidar, radar, and ultrasonic to understand their environment [22], [23], [11], off-road settings pose unique challenges. The volatile nature of these terrains, which can shift dramatically post natural events like rainfall, renders traditional urban-centric tools inadequate [24], [25]. This highlights the pressing need for innovative approaches that prioritize vision-centric navigation [26], a paradigm to which our solution, "TerrainSense", aligns itself. To address these challenges, our study tests and validates on a diverse set of off-road datasets, including those that represent natural parks, rural areas, deserts and simulated environments that encapsulate the vast spectrum of off-road domains. The key contributions of our work encompass:

1) We introduce an end-to-end framework designed for path planning in off-road environments, which uniquely relies on vision-based sensing, obviating the need for traditional maps. This framework employs CNNs as the backbone for robust feature extraction, augmented by transformer encoders to further enhance its capability.
2) Our approach ensures the joint detection and segmentation of paths, making it adaptable to a diverse spectrum of off-road terrains and conditions.
3) The framework showed notable proficiency in our experimental evaluations on six distinct off-road datasets. Interestingly, even without utilizing the training data, our model achieved promising results, emphasizing its efficiency and the potential for quicker training cycles in practical applications.

By seamlessly integrating a new path planning technique and visual perception, TerrainSense offers a transformative approach that enables autonomous vehicles to navigate with agility and adaptability in the face of dynamic terrains. The following sections elaborate on the development, methodology, evaluation, and implications of the proposed framework, thereby contributing to the advancement of autonomous navigation technology.

*This publication is based upon work supported by the Khalifa University under Award No. RC1-2018-KUCARS-8474000136.

[1]These authors are affiliated with the Khalifa University Center for Autonomous Robotic Systems (KUCARS), Abu Dhabi, 127788, UAE.

[2]These authors are affiliated with the Department of Electrical Engineering and Computer Science, Khalifa University, Abu Dhabi, 127788, UAE. {bilal.hassan, arjun.sharma, 100049370, majid.khonji, jorge.dias}@ku.ac.ae

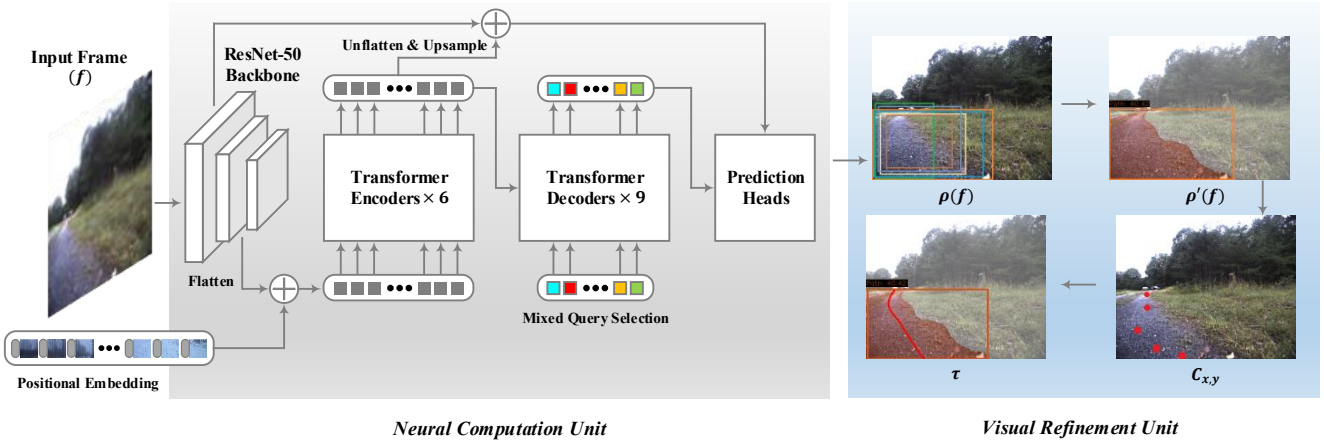

Fig. 1. High-level overview of TerrainSense working mechanism.

II. Proposed Methodology

Navigating off-road environments requires an efficient and robust path planning technique capable of discerning and delineating the main navigational path, which may not always be explicit due to similar surrounding landscapes. The primary objective of this research is to develop an end-to-end framework to extract pertinent features from raw images and utilize them to delineate a navigable path. To achieve this, we propose the TerrainSense framework, engineered to undertake frame-by-frame processing to extract the potential path that an autonomous vehicle can follow. The TerrainSense framework is composed of two distinct units: the neural computation unit and the visual refinement unit, as depicted in Fig. 1. The former unit, rooted in deep learning methodologies, is both learnable and adaptable, adjusting its behavior based on the data it processes. The latter, on the other hand operates as a fixed, deterministic post-processing unit specifically designed to generate paths based on the extracted visual clues.

In the proposed framework, a frame f of dimensions $h \times w$ is passed through a unified learnable model to produce an ensemble of detection boxes paired with their respective instance segmentation masks. Specifically, for the class of interest, the *Path*, we obtain the $\rho(f) = \{\mathbb{C}(b_1, s_1), \mathbb{C}(b_2, s_2), \ldots, \mathbb{C}(b_n, s_n)\}$, where n denotes the number of detected path instances, b_i represents the bounding box of the i^{th} path, and s_i is its binary mask of the exact dimensions as f. Moreover, the model associates a classification confidence score $\mathbb{C}$, for each detection box b_i and its mask s_i.

At the core of our study lies the objective to generate a navigable path denoted as τ. This relationship is formally expressed as $\tau = \rho(s*) = \{(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)\}$, where m is the number of path points, $s*$ refers to the mask associated with the bounding box that achieves the highest confidence score, provided this score surpasses a predetermined threshold ϑ. Three fundamental constraints guide our approach:

1) For a mask s_i to be eligible for path generation, the associated classification score, $\mathbb{C}(b_i, s_i)$, must surpass the threshold ϑ, captured by the inequality $\text{argmax}\{\mathbb{C}(b_i, s_i) \geq \vartheta\}$.
2) Each point in the resultant path τ must be tethered within the spatial confines stipulated by the segmentation mask (m), ensuring that $0 \leq x_j \leq m_w$ and $0 \leq y_j \leq m_h$ for every (x_j, y_j) in τ.
3) Post processing of τ for smoothing to purge outliers and yield a refined, final, and true navigable path.

With this foundation set, the following sections will detail our proposed TerrainSense framework.

A. Neural Computation Unit

The neural computation unit in the proposed TerrainSense framework is inspired by MaskDINO [27], a transformers-based model with refined capability to perform both segmentation and detection tasks simultaneously. In our approach, we integrate detection and segmentation functionalities to significantly improve path planning in environments defined solely by 'free space' and 'background' classes. The detection feature, although not directly utilized for object recognition due to label constraints, enhances the segmentation precision by implicitly aiding in the differentiation of freespace from complex backgrounds. This approach ensures more reliable path generation from segmented free space, crucial for effective navigation in environments where clear distinction between navigable paths and potential obstacles is essential.

To extract the navigable path from the input frames, the model begins with the ResNet-50 [28] backbone to extract multi-scale features from the off-road datasets. As these features are drawn from the raw images, they serve as the primary information input into subsequent processing stages. They are fed into Transformer encoders, equipped with positional embeddings. Through a series of self-attention mechanisms, the encoders enhance the extracted image features, preparing them for the subsequent decoding stages.

In the next stage, Transformer decoders use deformable attention to combine the outputs from the encoder layers, updating the queries with the encoded information in the

process. The model outcomes are relayed through prediction heads, ensuring refined bounding boxes and classification outcomes. Furthermore, it incorporates the segmentation branch, designed primarily for mask predictions. It generates a pixel embedding map by integrating the feature map from the backbone with the output from the Transformer encoders. This process facilitates detailed mask predictions, which are pivotal for our path detection task. For any given frame, the model would produce an output set defined as $\rho(f) = \{\mathbb{C}(b_1, s_1), \mathbb{C}(b_2, s_2), \ldots, \mathbb{C}(b_n, s_n)\}$, as shown in Fig. 1. Furthermore, Table I represents the architectural details of the neural computing unit in TerrainSense.

TABLE I
ARCHITECTURAL DETAILS OF THE NEURAL COMPUTING UNIT IN TERRAINSENSE

Block	Structure	Key Operations
Backbone (ResNet-50)	Stem	3x{3x[Conv+FBN]}
	Res_1 – 3 BottleNeck	3x{3x[Conv+FBN]}
	Res_2 – 4 BottleNeck	4x{3x[Conv+FBN]}
	Res_3 – 6 BottleNeck	6x{3x[Conv+FBN]}
	Res_4 – 3 BottleNeck	3x{3x[Conv+FBN]}
Transformer	PE Layer	PE+Conv+2x[Conv+GN]
	6 Encoder Layers	6x{MDA+DO+LN+MLP}
	9 Decoder Layers	9x{CA+LN+DO+MHA+MLP}
Predictor	Seg Head	3x{Conv+GN}
	Det Head	9x{MLP+Linear}

Conv=convolution, FBN = Frozen Batch Normalization, GN = Group Normalization, MDA = Multiscale Deform Attention, DO = Dropout, LN= layer normalization, MLP= Multilayer Perceptron, CA = Cross Attention, MHA = Multihead Attention

B. Visual Refinement Unit

In this section, we explain the visual refinement unit for extracting the paths from the processed frames by the neural computation unit. It consists of various components, as explained in the following subsections.

1) Masks Refinement: During the processing of frame f, we obtain multiple segmented instances. These are represented in the set $\rho(f) = \{\mathbb{C}(b_1, s_1), \mathbb{C}(b_2, s_2), \ldots, \mathbb{C}(b_n, s_n)\}$. Here, each segmented instance s_i is paired with a specific classification score $\mathbb{C}$. We aim to select the set with the highest score. However, choosing solely based on the peak score can be misleading. There might be scenarios where the top score is relatively low, leading to unreliable results. To circumvent this, we employ a two-stage refinement. The primary goal of the first stage is to filter out segmentations with subpar confidence scores. The refining decision is formulated as expressed in Eq. (1):

$$\rho'(f) = \begin{cases} s_i & \text{if } \mathbb{C}(b_i, s_i) > \vartheta \\ \emptyset & \text{otherwise} \end{cases}, \tag{1}$$

where s'_i is the refined segmentation result for the i^{th} instance. ϑ is the minimum confidence threshold. If, for a particular frame f, no instance surpasses the threshold ϑ, the frame is deemed unreliable for extracting the path. For such frames, the system reverts to using data from the previous frame. It is important to note that the fallback mechanism is designed to only consider the immediately preceding frame as a direct reference for continuity, ensuring that the system does not recursively look back beyond one frame. This approach limits the backward reference to a single step, maintaining temporal proximity and relevance. From the pool of refined sets, the final selection aims at picking the segmentation with the highest classification score, represented as expressed in Eq. (2):

$$s_{max} = \max_{s'_i \in \rho'(f)} \mathbb{C}(b_i, s'_i), \tag{2}$$

where $\rho'(f)$ represents the set of refined segmentations from frame f. Next, we leverage mathematical modeling, interpolation methods, and filtering to ensure an optimal and smooth path from the extracted mask s'_i.

2) Weighted Average Center Extraction: Given the binary mask s'_i, each row is individually processed to find the center of the potential path. This is achieved through a weighted average mechanism as expressed in Eq. (3):

$$center_x(i) = \frac{\sum_{j=1}^{N_i} w_j x_j}{\sum_{j=1}^{N_i} w_j}, \tag{3}$$

where N_i is a number of white pixels in row i, x_j is the column position of the j^{th} white pixel in that row, and w_j is the associated weight of the j^{th} pixel (uniformly weighted in the current implementation, but can be adapted for more sophisticated weightings for advanced scenarios).

3) Adaptive Path Decimation: The extracted center path is usually noisy and contains some outliers. To circumvent this, an adaptive decimation strategy condenses the path by retaining every tenth waypoint, yielding a representation that encapsulates essential path characteristics.

4) Filtering for Path Smoothing: In the subsequent stage, we utilized two filtering processes to refine the path. Initially, we employed B-Spline interpolation on the extracted waypoints, ensuring a fit with minimal squared error. This approach not only preserved intricate terrain contours but also effectively eliminated minor inconsistencies. Following that, we applied the Savitzky-Golay filter. Unlike traditional moving-average techniques, this filter fits low-degree polynomials to specific subsets of the input data. This results in a smoother trajectory while maintaining essential high-frequency details, which is crucial for off-road terrains where retaining path characteristics is paramount.

5) Fidelity-Preserving Interpolation: The final trajectory, albeit smooth and efficient, might differ in resolution from the initial trajectory. A subsequent interpolation phase is introduced to map the trajectory back to the original resolution, ensuring it is compatible with the original mask dimensions and maintains high fidelity. Algorithm 1 summarizes the work flow of the visual refinement unit for path extraction.

III. EXPERIMENTAL SETUP

A. Datasets

In this research, we employed three carefully chosen datasets to train our model, each catering to diverse terrains, lighting conditions, and trail types.

Algorithm 1: Visual Refinement Unit for Path Extraction

Input : binary mask (s_i') of size $rows \times cols$
Output: refined path (τ): xn, yn

begin
- Initialize an empty list: path = []
- **for** $i = 1$ *to* $rows$ **do**
  - Extract white_pixels: Locations where s_i'[i, :] equals 255
  - **if** *white_pixels is non-empty* **then**
    - Compute center using:

$$center_x(i) = \frac{\sum_{j=1}^{N_i} w_j x_j}{\sum_{j=1}^{N_i} w_j}$$

    - Append $center_x(i)$ to path
- Apply B-Spline interpolation on path to get T_s:

$$T_s = B\_spline(path, \mathcal{S})$$

- Decimate T_s to get T_d using:

$$T_d = decimate(T_s, step\_size)$$

- Filter T_d using Savitzky-Golay parameters to get T_f:

$$T_f = SG\_filter(T_d, window\_params)$$

- Interpolate T_f to match the length of the original path and get τ
- **return** τ: xn, yn

1) Off-Road Open Desert Trail Detection (O2DTD): This locally-curated dataset contains 5045 desert images, taken at six different times to capture desert-specific light and shadow variations. With its pixel-wise annotations distinguishing sky, ground, and trail, we utilized the trail annotations to identify the *Path*, classifying the others as background.

2) Robot Unstructured Ground Driving (RUGD): This public dataset [29] comprises around 7436 images from 18 sequences. Initially segmented into 24 classes, including eight terrain types, we consolidated them, focusing on the *Path* category and designating the rest as background.

3) Off-Road Freespace Detection (ORFD): With 12197 images, this public dataset [30] showcases varied off-road terrains under different conditions. Primarily annotated into three categories: traversable, non-traversable, and unreachable areas, we identified the traversable area as the *Path*, categorizing the rest as background.

While O2DTD provided path trajectory labels, both RUGD and ORFD lacked them. To ensure consistent analysis, we generated these labels for the test segments in the latter two datasets. For generating ground truth paths in RUGD and ORFD datasets, we initially utilized a proposed automated center-points extraction technique from the segmented masks to outline potential paths. This process is then refined through meticulous manual adjustments to ensure accuracy and reliability of the path. Further, we validated the model's performance through a blind test, which means the model was evaluated using three off-road datasets (2 real-world and one synthetic) that it had not previously seen during training, to impartially test its generalizability and robustness. Comprehensive details of all datasets and their subsets are provided in Table II.

TABLE II
DATASET DETAILS

Dataset	Training	Validation	Test
ORFD	8397	1245	2555
RUGD[a]	5235	280	1921
O2DTD	3235	504	1009
Freiburg Forest[b]	-	-	136
CAT[b]	-	-	544
Forza Horizon 5[b]	-	-	300

[a] Randomly selected sequences for test ('trail-3', 'trail-7', 'trail-12', 'trail-15'). [b] Used only the test set to blind test model performance.

B. Training Setup

1) Model Hyperparameters: The TerrainSense framework was trained using the AdamW optimizer with a base learning rate of 0.00025. Training extended for a total of 200,000 iterations, with each batch consisting of 12 images. The entire model was implemented in Python 3.8.17, leveraging the capabilities of PyTorch 2.0.1. This configuration was chosen to optimize both convergence speed and model accuracy, balancing computational resources with training efficiency.

2) Loss Function: In this research, we trained the model for joint detection and segmentation tasks. For this purpose, we follow a similar approach to [27] to integrate three distinctive losses: classification loss L_{cls}, box loss L_{box}, and mask loss L_{mask}. Specifically, we implemented the focal loss for classification and combined L1 loss (L_{L1}) and GIOU loss (L_{giou}) for bounding box regression. Our mask loss was conceived through a blend of crossentropy L_{ce} and dice loss L_{dice}, supplemented by point loss for augmented efficiency. The cumulative loss function is expressed in Eq. (4):

$$\begin{aligned} L = \lambda_{cls} L_{cls} + \lambda_{L1} L_{L1} \\ + \lambda_{giou} L_{giou} + \lambda_{ce} L_{ce} \\ + \lambda_{dice} L_{dice}, \end{aligned} \tag{4}$$

In our implementation, the respective weightings were: $\lambda_{cls} = 4$, $\lambda_{L1} = 5$, $\lambda_{giou} = 2$, $\lambda_{ce} = 5$, and $\lambda_{dice} = 5$.

IV. RESULTS

A. Qualitative

In this section, qualitative results are presented to evaluate the performance of the proposed TerrainSense model.

1) Segmentation Results: Fig. 2 illustrates a comprehensive visual comparison between the original off-road scene, the ground truth, and our model's predicted segmentation results. The model demonstrates proficiency in capturing the majority of the nuances of the off-road environment, with its predictions closely matching the ground truth. This showcases its capability to discern and detect subtle path features vital for navigation. However, instances of under-segmentation, where certain path parts are missed, and over-segmentation, where extraneous areas are marked, highlight areas for model refinement. Addressing these discrepancies will be essential for ensuring enhanced accuracy and safety in off-road navigation applications.

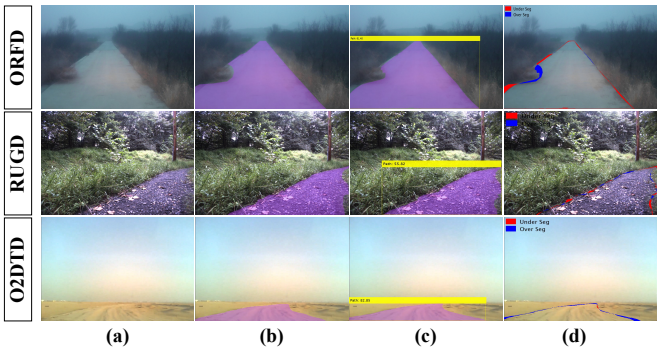

Fig. 2. Path segmentation results are shown on randomly selected images from all three test datasets. (a) Original images, (b) Ground-truth labels, (c) Segmented path predictions, and (d) Under-over segmentation are shown in red and blue color respectively.

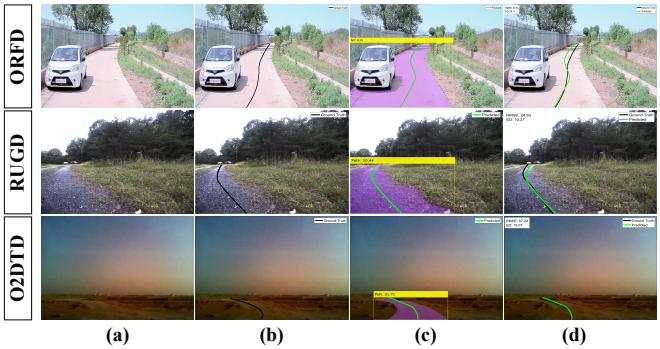

Fig. 3. Path prediction results are shown on randomly selected images from all three test datasets. (a) Original images, (b) Ground-truth labels, (c) Extracted paths, and (d) Overlayed ground-truth and predicted paths.

2) Path Prediction: In Fig. 3, the focus shifts to the model's ability to predict paths within the previously identified segmented regions of the off-road environment. The visual representation clearly depicts the model tracing the contours of the segmented terrains, demonstrating its proficiency in understanding the topography and potential obstructions. This alignment between the predicted path and segmented areas accentuates the model's potential for real-time off-road navigation. However, noticeable deviations between the predicted path and the segmented terrains signal areas where the model might either be heavily reliant on segmentation outcomes or perhaps overlooking crucial environmental cues. Such instances spotlight the need for further refinement to ensure a seamless and safe off-road traversal experience.

B. Quantitative

In this section, quantitative results are presented to evaluate the performance of the proposed TerrainSense model.

1) Detection Performance on Different Test Datasets: Table III showcases the performance of the TerrainSense framework on three datasets: ORFD, RUGD, and O2DTD. We used Average Precision (AP) at two Intersection over Union (IoU) thresholds (AP50 and AP95) to evaluate the model performance. We also calculated a combined AP score across an IoU range from 0.50 to 0.95 with a step size of 0.5. Here, it is important to emphasize that TerrainSense was jointly trained on three datasets, which included a variety of off-road terrain types, different weather conditions, and times of day. From the results, it is observed that the models perform best on the O2DTD dataset, followed by the ORFD dataset, and then the RUGD dataset. This could point to the challenges specific to the RUGD dataset, such as its complex terrain features and diverse vegetation classes. As the IoU threshold increases, there's a decrease in AP values for all datasets, especially at the AP95 threshold. Overall, the combined AP scores give an idea of the model's performance across different conditions. Given its training on a wide-ranging dataset, the performance of the TerrainSense framework is acceptable. However, there might be a need for further improvements, particularly for challenging datasets like RUGD.

TABLE III
TERRAINSENSE PERFORMANCE EVALUATION ON TEST DATASETS FOR DETECTION, SEGMENTATION, AND PATH PREDICTION

Dataset	Detection			Segmentation		Path	
	AP50	AP95	AP	DSC	IoU	RMSE	ED
ORFD	0.7839	0.3723	0.5013	0.8434	0.7929	113.28	73.28
RUGD	0.7359	0.2737	0.4340	0.7743	0.6447	125.22	84.77
O2DTD	0.9984	0.3407	0.5633	0.9467	0.8966	60.51	23.40

2) Segmentation Performance on Different Test Datasets: Next, we presents the path segmentation results of the TerrainSense framework, utilizing the Dice Similarity Coefficient (DSC) and Intersection over Union (IoU) as metrics. The segmentation performance across all three datasets is shown in Table III, where it can be noted that the framework attained DSC and IoU scores of 0.8434 and 0.7929, respectively, on the ORFD dataset. These results indicate a balanced performance compared to other two datasets, though certain instances could be proving challenging. Specific terrains or lighting conditions in ORFD may be areas for further exploration and model optimization.

The RUGD dataset shows lower performance metrics, with a DSC of 0.7743 and an IoU of 0.6447. This suggests that the RUGD dataset includes a wider variety of terrains or conditions that might not be fully represented in the model's training data. Investigating the unique characteristics of RUGD's terrains could provide insights for improving model training approaches. On the other hand, the O2DTD dataset demonstrates outstanding and consistent results, with a DSC of 0.9467 and an IoU of 0.8966, both exhibiting low standard deviations. The superior performance on the O2DTD dataset can be attributed to the nature of our custom dataset, which exclusively features desert terrains at various times of the day. This specificity likely makes it easier for the model to adapt and excel, as it only needs to learn and recognize a narrower range of terrain features and lighting conditions.

3) Path Prediction Results: We evaluated the path prediction performance of TerrainSense framework, using the root mean square error (RMSE) and Euclidean distance (ED) metrics, as shown in Table III. Given the large image sizes

used for comparison, the values in Table III might appear elevated. The results on the ORFD highlight some deviation between predicted and ground truth paths, with an RMSE of 113.28 and a ED of 73.28. For RUGD, the RMSE and ED are higher at 125.22 and 84.77 respectively, pointing to broader discrepancies.

The framework demonstrates superior performance on O2DTD dataset with an RMSE of 60.51 and remarkably lower ED at 23.40 respectively. This consistency suggests the model's predictions align well with ground truths in O2DTD. Overall, there is variation in path prediction across datasets, driven by dataset-specific challenges and image resolution. While O2DTD showcases the model's capabilities, results on ORFD and RUGD highlight areas for improvement.

C. Comparison with State-of-the-Art

In this section, we present a comparative analysis of our proposed TerrainSense framework against other state-of-the-art (SOTA) methods [31], [32], [33], [34], [35], [36], [37], across three distinct off-road datasets [29], [30], as shown in Table IV. Our comparison focuses on three critical aspects: segmentation accuracy, path prediction, and inference speed. The values in Table IV represent the aggregated results across all datasets.

Our findings highlight that the TerrainSense framework notably surpasses other SOTA methods in all evaluated aspects. Specifically, regarding the segmentation of free-space regions, TerrainSense achieves a 4.29% improvement over its nearest competitor. Furthermore, when focusing on the precision of path prediction, our framework demonstrates a significant enhancement, reducing the RMSE by 9.32% in comparison to the second-best performing method. Additionally, the TerrainSense framework exhibits superior efficiency, with an 18.28% faster inference time than the closest alternative.

This comparative analysis underscores the effectiveness of the TerrainSense framework in handling off-road navigation challenges, showcasing its strengths in both accuracy and operational efficiency.

TABLE IV

TERRAINSENSE PERFORMANCE COMPARISON WITH SOTA

Method	Mask (IoU)	Path (RMSE)	Time (msec)
Mask2Former [31]	0.7385	109.92	385
clipseg [32]	0.6429	131.81	352
CGNet [33]	0.5541	155.18	268
OneFormer [34]	0.6475	131.38	435
PSPNet [35]	0.7461	110.11	376
GroupViT [36]	0.6899	122.80	419
OCRNet [37]	0.6119	148.16	336
Ours	0.7781	99.67	219

D. Generalization Evaluation

Generalization performance is paramount when deploying deep models to tackle real-world challenges. In our study, we sought to validate the adaptability of TerrainSense on out-of-distribution data. To this end, an experiment was conducted where the model, post-training, was evaluated on two real-world datasets: Freiburg Forest (F-Forest) [19] and CAT [9], in addition to a synthetic dataset derived from the Forza Horizon 5 (FH5) game. Our findings, presented in Table V, not only showcase that TerrainSense exhibits commendable generalization on unseen datasets but also underscore the potential of our model to rapidly adapt to novel terrains with minimal training. This inherent flexibility suggests the promise of TerrainSense for applications demanding incremental learning and swift adaptability to evolving terrains. Additionally, Fig. 4 shows the qualitative results of TerrainSense domain adaptive performance on unseen datasets.

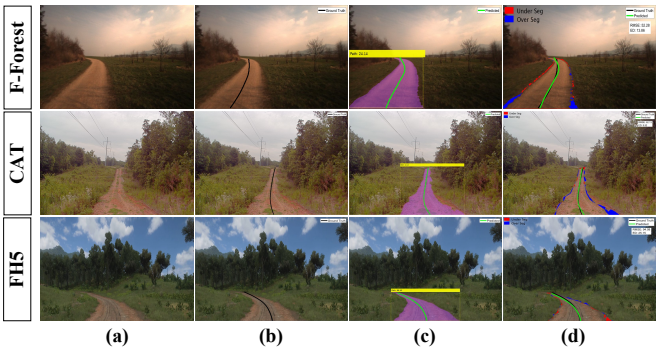

Fig. 4. Path segmentation and prediction results on unseen datasets. (a) Original images, (b) Ground-truth labels, (c) Segmented free space and extracted path, and (d) Overlayed ground-truth and predicted paths.

TABLE V

TERRAINSENSE PERFORMANCE EVALUATION ON UNSEEN DATASETS

Dataset	Detection			Segmentation		Path	
	AP50	AP95	AP	DSC	IoU	RMSE	ED
F-Forest [19]	0.5413	0.1032	0.1564	0.7840	0.6274	134.68	86.08
CAT [9]	0.7021	0.2108	0.3025	0.8191	0.7009	95.73	53.53
FH5	0.7465	0.1847	0.3442	0.8332	0.7304	103.78	65.82

V. CONCLUSION

The challenges of navigating autonomous vehicles in unstructured and off-road terrains have long been a bottleneck in the progress of autonomous driving. While traditional methods show proficiency in structured urban settings, they often falter when faced with the unpredictabilities of off-road environments. This paper introduced TerrainSense, a novel framework focusing on vision-centric, mapless navigation strategies tailored for these challenging terrains. As we delved into the details, it became evident that the strength of TerrainSense lies in its ability to bridge the gap between the robustness of deep learning techniques and the dynamic requirements of off-road navigation. Furthermore, recognizing the importance of vehicle dynamics, future work will incorporate dynamic constraints to ensure the navigation strategies are not only vision-centric but also reflective of the vehicle's physical capabilities and limitations.

REFERENCES

[1] S. Kuutti, R. Bowden, Y. Jin, P. Barber, and S. Fallah, "A survey of deep learning applications to autonomous vehicle control," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 2, pp. 712–733, 2020.

[2] G. Bresson, Z. Alsayed, L. Yu, and S. Glaser, "Simultaneous localization and mapping: A survey of current trends in autonomous driving," *IEEE Transactions on Intelligent Vehicles*, vol. 2, no. 3, pp. 194–220, 2017.

[3] H. Ye, J. Mei, and Y. Hu, "M2f2-net: Multi-modal feature fusion for unstructured off-road freespace detection," in *2023 IEEE Intelligent Vehicles Symposium (IV)*, pp. 1–7, IEEE, 2023.

[4] O. Mayuku, B. W. Surgenor, and J. A. Marshall, "A self-supervised near-to-far approach for terrain-adaptive off-road autonomous driving," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 14054–14060, IEEE, 2021.

[5] D. W. Carruth, C. T. Walden, C. Goodin, and S. C. Fuller, "Challenges in low infrastructure and off-road automated driving," in *2022 Fifth International Conference on Connected and Autonomous Driving (MetroCAD)*, pp. 13–20, IEEE, 2022.

[6] X. Liang, T. Wang, L. Yang, and E. Xing, "Cirl: Controllable imitative reinforcement learning for vision-based self-driving," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 584–599, 2018.

[7] C. Xu, W. Zhao, J. Liu, C. Wang, and C. Lv, "An integrated decision-making framework for highway autonomous driving using combined learning and rule-based algorithm," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 4, pp. 3621–3632, 2022.

[8] B. Gallazzi, P. Cudrano, M. Frosi, S. Mentasti, and M. Matteucci, "Clothoidal mapping of road line markings for autonomous driving high-definition maps," in *2022 IEEE Intelligent Vehicles Symposium (IV)*, pp. 1631–1638, IEEE, 2022.

[9] S. Sharma, L. Dabbiru, T. Hannis, G. Mason, D. W. Carruth, M. Doude, C. Goodin, C. Hudson, S. Ozier, J. E. Ball, *et al.*, "Cat: Cavs traversability dataset for off-road autonomous driving," *IEEE Access*, vol. 10, pp. 24759–24768, 2022.

[10] D. Maturana, P.-W. Chou, M. Uenoyama, and S. Scherer, "Real-time semantic mapping for autonomous off-road navigation," in *Field and Service Robotics: Results of the 11th International Conference*, pp. 335–350, Springer, 2018.

[11] J. S. Berrio, M. Shan, S. Worrall, and E. Nebot, "Camera-lidar integration: Probabilistic sensor fusion for semantic mapping," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 7637–7652, 2021.

[12] R. Ahmed, A. Al Shehhi, B. Hassan, N. Werghi, and M. L. Seghier, "An appraisal of the performance of ai tools for chronic stroke lesion segmentation," *Computers in Biology and Medicine*, p. 107302, 2023.

[13] B. Hassan, S. Qin, T. Hassan, M. U. Akram, R. Ahmed, and N. Werghi, "Cdc-net: Cascaded decoupled convolutional network for lesion-assisted detection and grading of retinopathy using optical coherence tomography (oct) scans," *Biomedical Signal Processing and Control*, vol. 70, p. 103030, 2021.

[14] R. Ahmed, Y. Chen, B. Hassan, L. Du, T. Hassan, and J. Dias, "Hybrid machine-learning-based spectrum sensing and allocation with adaptive congestion-aware modeling in cr-assisted iov networks," *IEEE Internet of Things Journal*, vol. 9, no. 24, pp. 25100–25116, 2022.

[15] X. Cai, M. Everett, J. Fink, and J. P. How, "Risk-aware off-road navigation via a learned speed distribution map," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2931–2937, IEEE, 2022.

[16] D. Feng, C. Haase-Schütz, L. Rosenbaum, H. Hertlein, C. Glaeser, F. Timm, W. Wiesbeck, and K. Dietmayer, "Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 3, pp. 1341–1360, 2020.

[17] S. Triest, M. Sivaprakasam, S. J. Wang, W. Wang, A. M. Johnson, and S. Scherer, "Tartandrive: A large-scale dataset for learning off-road dynamics models," in *2022 International Conference on Robotics and Automation (ICRA)*, pp. 2546–2552, IEEE, 2022.

[18] M. G. Castro, S. Triest, W. Wang, J. M. Gregory, F. Sanchez, J. G. Rogers, and S. Scherer, "How does it feel? self-supervised costmap learning for off-road vehicle traversability," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 931–938, IEEE, 2023.

[19] A. Valada, G. L. Oliveira, T. Brox, and W. Burgard, "Deep multispectral semantic scene understanding of forested environments using multimodal fusion," in *2016 International Symposium on Experimental Robotics*, pp. 465–477, Springer, 2017.

[20] K. Wong, Y. Gu, and S. Kamijo, "Mapping for autonomous driving: Opportunities and challenges," *IEEE Intelligent Transportation Systems Magazine*, vol. 13, no. 1, pp. 91–106, 2020.

[21] M. Haklay and P. Weber, "Openstreetmap: User-generated street maps," *IEEE Pervasive computing*, vol. 7, no. 4, pp. 12–18, 2008.

[22] L. Caltagirone, S. Scheidegger, L. Svensson, and M. Wahde, "Fast lidar-based road detection using fully convolutional neural networks," in *2017 ieee intelligent vehicles symposium (iv)*, pp. 1019–1024, IEEE, 2017.

[23] L. Sun, Z. Yan, A. Zaganidis, C. Zhao, and T. Duckett, "Recurrent-octomap: Learning state-based map refinement for long-term semantic mapping with 3-d-lidar data," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3749–3756, 2018.

[24] P. Jiang, P. Osteen, M. Wigness, and S. Saripalli, "Rellis-3d dataset: Data, benchmarks and analysis," in *2021 IEEE international conference on robotics and automation (ICRA)*, pp. 1110–1116, IEEE, 2021.

[25] A. Valada, J. Vertens, A. Dhall, and W. Burgard, "Adapnet: Adaptive semantic segmentation in adverse environmental conditions," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4644–4651, IEEE, 2017.

[26] X. Wang, Z. Zhu, Y. Zhang, G. Huang, Y. Ye, W. Xu, Z. Chen, and X. Wang, "Are we ready for vision-centric driving streaming perception? the asap benchmark," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9600–9610, 2023.

[27] F. Li, H. Zhang, H. Xu, S. Liu, L. Zhang, L. M. Ni, and H.-Y. Shum, "Mask dino: Towards a unified transformer-based framework for object detection and segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3041–3050, 2023.

[28] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

[29] M. Wigness, S. Eum, J. G. Rogers, D. Han, and H. Kwon, "A rugd dataset for autonomous navigation and visual perception in unstructured outdoor environments," in *International Conference on Intelligent Robots and Systems (IROS)*, 2019.

[30] C. Min, W. Jiang, D. Zhao, J. Xu, L. Xiao, Y. Nie, and B. Dai, "Orfd: A dataset and benchmark for off-road freespace detection," in *2022 International Conference on Robotics and Automation (ICRA)*, pp. 2532–2538, IEEE, 2022.

[31] B. Cheng, I. Misra, A. G. Schwing, A. Kirillov, and R. Girdhar, "Masked-attention mask transformer for universal image segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 1290–1299, 2022.

[32] T. Lüddecke and A. Ecker, "Image segmentation using text and image prompts," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7086–7096, 2022.

[33] T. Wu, S. Tang, R. Zhang, J. Cao, and Y. Zhang, "Cgnet: A light-weight context guided network for semantic segmentation," *IEEE Transactions on Image Processing*, vol. 30, pp. 1169–1179, 2020.

[34] J. Jain, J. Li, M. T. Chiu, A. Hassani, N. Orlov, and H. Shi, "Oneformer: One transformer to rule universal image segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2989–2998, 2023.

[35] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2881–2890, 2017.

[36] J. Xu, S. De Mello, S. Liu, W. Byeon, T. Breuel, J. Kautz, and X. Wang, "Groupvit: Semantic segmentation emerges from text supervision," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18134–18144, 2022.

[37] Y. Yuan, X. Chen, and J. Wang, "Object-contextual representations for semantic segmentation," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VI 16*, pp. 173–190, Springer, 2020.