2026 NDIA MICHIGAN CHAPTER
GROUND VEHICLE SYSTEMS ENGINEERING
AND TECHNOLOGY SYMPOSIUM
CYBERSECURITY TECHNICAL SESSION
AUG. 11–13, 2026 – NOVI, MICHIGAN

EVALUATING THE ROBUSTNESS OF SEGMENTATION MODELS AGAINST ADVERSARIAL PATCHES IN OFF-ROAD ENVIRONMENTS

Christopher Salas[1], Mert D. Pesé, PhD[1], Bing Li, PhD[1], Jonathon Smereka, PhD[2], Long Cheng, PhD[1]

[1]Clemson University, Clemson, SC
[2]US Army DEVCOM GVSC

ABSTRACT

Semantic Segmentation (SS) is critical for autonomous vehicles to navigate off-road environments by identifying drivable terrain. Although models like ResNet34+UNet and EfficientViT have been proposed for these tasks, their susceptibility to localized adversarial patches in unstructured environments remains underresearched. This paper presents a comprehensive robustness evaluation of six real-time SS architectures, including the state-of-the-art YOLOv11 and YOLOv12 segmentation variants against five diverse adversarial patch schemes. Our experiments, conducted on a modified YCOR dataset, demonstrate that EfficientViT is the most resilient architecture, maintaining high accuracy with minimal performance degradation. In contrast, single-stage models like YOLOv11n-seg exhibit significant vulnerability, with pixel accuracy drops reaching 26.55%. We also show how decreases in overall segmentation accuracy impact the segmentation models' ability to discern traversable terrain from non-traversable terrain.

Citation: C. Salas, M. Pesé, B. Li, J. Smereka, L. Cheng, "Evaluating the Robustness of Segmentation Models Against Adversarial Patches in Off-Road Environments," In Proceedings of the Ground Vehicle Systems Engineering and Technology Symposium (GVSETS), NDIA, Novi, MI, Aug. 11-13, 2026.

1. INTRODUCTION

Autonomous vehicles (AVs) operating in off-road environments rely heavily on Semantic Segmentation (SS) to identify drivable terrain and navigate safely. To meet the demand for real-time inference in these complex environments, researchers have deployed various classes of SS architectures. These include conventional encoder-decoder networks that yield pixel-wise predictions (such as ResNet34+UNet [1]), Transformer-based models designed for high-resolution, low-latency predictions (like EfficientViT [2]), and single-stage models (such as YOLOv11 [3] and YOLOv12 [4]). Unlike

traditional architectures, single-stage models use a single neural network to predict bounding boxes and class probabilities for objects in a given scene simultaneously. With the increased use of object recognition and semantic segmentation models comes an increase in the potential for adversaries to take advantage of these systems for their own malicious exploitations. One common attack against these CV models is adversarial perturbations in the form of a localized patch [5] [6] [7]. While the threat of adversarial patches is well-documented, existing research overwhelmingly focuses on structured on-road environments. This leaves a severe gap in our understanding of off-road AV security. Unlike on-road environments, off-road navigation lacks structured visual cues and depends almost entirely on pixel-level terrain classification. Consequently, localized adversarial patches optimized for off-road SS models have the potential to maliciously alter a vehicle's perception of traversable terrain, leading to catastrophic navigational failures.

The work presented in this paper aims to address the research gap in understanding how adversarial patches can affect segmentation models in off-road environments. To the best of our knowledge, this paper would present the first extensive look at how adversarial patches can impact off-road driving environments and their effect on various CNN-based and Transformer-based semantic segmentation models. This work answers the following key research questions. RQ1: How robust are proposed real-time semantic segmentation models for off-road environments to adversarial patches? RQ2: Which adversarial patch optimization schemes are the segmentation models most vulnerable to? RQ3: How does the performance of state-of-the-art single stage segmentation models compare to other proposed models?
Our main contributions are as follows:

1. We perform a comprehensive robustness evaluation across six distinct segmentation model architectures against five distinct adversarial patch optimization schemes.

2. We reveal how adversarial patches uniquely threaten off-road autonomous navigation by explicitly degrading the model's ability to discern traversable terrain from non-traversable terrain, fundamentally blinding the vehicle's path-finding systems.

2. BACKGROUND AND RELATED WORKS

2.1. Segmentation Models

Long *et al.* [8] paved the way for many modern SS models by demonstrating that Fully Convolutional Networks (FCNs) could be used to outperform the state-of-the-art in SS at the time. YOLO [9] is a FCN-based object detection model that uses a single neural network to predict bounding boxes and class probabilities for objects in a given scene. An official YOLO model that supports segmentation was released with YOLOv5 [10] and has undergone continuous improvements over the years, culminating in the recent release of YOLOv12 [4].

Most existing research on SS models focuses heavily on structured on-road environments, optimizing architectures for predictable visual cues like lane markings, paved surfaces, and standardized traffic infrastructure. In contrast, significantly less attention has been given to off-road environments, which are characterized by unstructured terrain sparse landmarks, and highly varying vegetation. To address this gap, researchers have recently proposed models specifically tailored for these challenging conditions. ResNet34+UNet was first pro-

Table 1: Adversarial Patch Scheme Comparison

Adversarial Scheme	White-Box Optimization	Black-Box Transfer	Data-Free	Uses EOT
LaVAN [5]	✓	✗	✗	✗
DiAP [11]	✓	✓	✓	✗
Patch-Wise [12]	✓	✓	✗	✗
SemSegAdv [13]	✓	✓	✗	✓
AdvTransfer [14]	✓	✓	✗	✓

posed by Rahi *et al.* [1] as a suitable SS model to identify off-road traversable terrain in real-time. Furthermore, Pickeral *et al.* [2] demonstrated how the EfficientViT architecture can be utilized to make real-time inferences in off-road environments.

2.2. Adversarial Examples for Autonomous Vehicle Perception Modules

Extensive research has been done on how adversarial examples can be optimized to affect various computer vision (CV) tasks such as image classification and object detection. Generally, these adversarial attack methodologies can be classified into two primary categories based on the attacker's knowledge of the victim system: white-box attacks and black-box attacks. In a white-box setting, the adversary has complete knowledge of and access to the target model, including its architecture, trained weights, gradients, and often the training data, allowing for direct optimization of the patch. Conversely, in a black-box setting, the attacker lacks direct access to the model's internal parameters; instead, they must generate patches using data-independent methods or by optimizing perturbations on a known surrogate model with the expectation that the attack will transfer to the unseen victim model.

Early localized adversarial perturbations primarily operated under white-box assumptions. Adversarial examples in the form of localized patches aimed at fooling image classifiers were first presented in Brown *et al.* [6]. Later, Eykholt *et al.* [15] demonstrated how adversarial stickers can be placed on traffic signs to fool Deep Neural Networks (DNNs) in real-world, safety-critical scenarios. Treu *et al.* [16] present the first adversarial example based method targeting human instance segmentation by generating adversarial textures based on fashion style images. The resulting adversarial textures are effectively able to make a person invisible to the detection networks.

DAG (Dense Adversary Generation) [17] presented one of the first novel loss functions for creating visually imperceptible perturbations that target semantic segmentation and object detection. While DAG focuses on per-image adversarial perturbations, [18] extends the threat model by demonstrating the viability of image-agnostic, universal perturbations. LaVAN (Localized and Visible Adversarial Noise) [5] presents one of the earliest iterations of localized adversarial perturbations in the form of a patch that can be used to fool image classifiers. Their methodology focuses on creating visible patches that only take up roughly 2% of the image area. LaVAN is considered a white-box attack because it requires full knowledge of the victim model in order to optimize using gradient descent. Patch-Wise [12] uses a novel

optimization algorithm that focuses on patch regions rather than individual pixels. To maximize the perturbation's impact across these regions, the method proposed by Gao *et al.* [12] utilizes the full gradient access of the victim model to iteratively update the patch, strictly implementing it as a white-box attack.

To move beyond purely digital perturbations and ensure patches remain effective when printed and captured by physical cameras, researchers must introduce specialized optimization techniques. Addressing this need, Rossolini *et al.* [19] and Nesti *et al.* [13] present a novel loss function to optimize adversarial patches that target SS models and are physically realizable. The novel loss function is first proposed for un-targeted attacks in [13] and then later extended for targeted attacks in [19]. We refer to patches optimized using this method in our experiments as SemSegAdvPatch. SemSegAdvPatch is a white-box attack that trains on a surrogate model, but is evaluated as a black-box attack that can be transferred to unseen victim models deployed on physical autonomous vehicles after optimization in [19].

Continuing the focus on cross-model transferability, Shekhar *et al.* [14] present a simplified loss function to optimize adversarial patches. Patches generated using the loss function presented by [14] are referred to as AdvTransferPatch in our experiments. AdvTransferPatch is a white-box attack for the patch generation step, but once generated on a surrogate model, the patch can be used as a black-box attack on victim models it has not seen. Zhou *et al.* [11] present DiAP (Data Independent Adversarial Patch), a black-box patch optimization strategy that does not require the attacker to have access to any of the model training data. DiAP targets the feature extractors of pretrained, publicly available models to create a patch that is optimized specifically for the target model. Table 1 gives an overview of the five patch optimization algorithms that we evaluated in this paper.

2.3. Adversarial Perturbation Defenses

Defending against localized adversarial patches generally falls into two categories: adversarial training and patch detection. Adversarial training, as proposed by Kurakin *et al.* [20], involves directly training a model on adversarial perturbations to make it more robust towards them. It is limited in its application because it requires the victim model to have prior knowledge of the adversarial examples that an attacker will be using in order to train against them.

As an alternative to adversarial training, researchers have explored various patch detection and masking techniques. For image classification tasks, Li *et al.* [21] proposed Random-Local-Ensemble (RLE) to detect and reject patched inputs before classification. For semantic segmentation models, Yatsura *et al.* [22] introduced Demasked Smoothing, a certifiable defense that masks different parts of the input to provide theoretical guarantees against patches up to a predefined size.

Most adversarial patch defense methods, such as those proposed by [21] and [22] introduce an increase in latency to model predictions in exchange for a more robust analysis. From an autonomous vehicle perspective, this latency increase is often unacceptable due to potential critical safety consequences. To address these real-time constraints within autonomous vehicle semantic segmentation, a representative defense strategy is proposed by Rossolini *et al.* [19] known as the Fast Patch Detection Algorithm (FPDA). FPDA works by identifying and counting the presence of over-activated internal features which are common when adversarial patches are present.

2.4. Adversarial Perturbation Evaluations Against Segmentation Models

While adversarial patches have been extensively evaluated against standard image classifiers, comprehensive studies targeting the vulnerabilities of semantic segmentation models remain limited. Among the existing literature, Wang *et al.* [23] presented an in-depth evaluation of how the performance of SS models can be negatively impacted by adversarial patches from an aerial perspective. Several of the adversarial patch optimization schemes analyzed in our study were initially evaluated by [23] against SS models within this aerial context. Specifically, their evaluation assessed the robustness of several distinct architectures, including BSNet, MANet, AFNet, SSAtNet, and MDANet.

However, the original five SS models in [23] were not evaluated as real-time inference models. Because their work focuses on parameter-heavy models that typically have longer inference times, those findings do not perfectly translate to the strict real-time constraints of autonomous ground vehicles. Furthermore, a significant research gap remains regarding how adversarial attacks affect semantic segmentation models deployed specifically in off-road autonomous driving scenarios. Deoli *et al.* [24] evaluates the robustness of off-road autonomous driving segmentation against Projected Gradient Descent (PGD) attacks [25] with L^2 and L^∞ upper-bounds. Each of these attacks are perturbations made to the entire image and do not abide by the same localization and spatial constraints as adversarial patches. The most common attack goal for adversarial patches in autonomous vehicles is how they can affect the perception of various traffic signs [26]. This threat is unique to on-road environments and not as applicable to off-road environments that do not have such traffic signs.

3. THREAT MODEL

For an off-road autonomous vehicle that utilizes semantic segmentation in its navigation system, the main goal of an adversarial patch should be to affect the vehicle's perception of the trail or drivable area.

3.1. Patch Optimization Assumptions

In this work, we tested five different adversarial patch optimization schemes (LaVAN [5], Patch-Wise [12], DiAP [11], SemSegAdvPatch [13], and AdvTransferPatch [14]) across various segmentation models. A common assumption across all five of these techniques is access to the victim model. The assumption is that the attacker will have access directly to the victim model, or to a surrogate model with equivalent architecture to the victim model. Because they require this level of access, all five methods fundamentally operate under white-box optimization principles.

While these methods rely on white-box access during the generation phase, prior research indicates that patches optimized via DiAP, SemSegAdvPatch, and AdvTransferPatch exhibit strong cross-model transferability. Consequently, once generated, these patches can be deployed as black-box attacks against unseen victim models.

Beyond victim model access, these optimization schemes vary in their specific training data requirements. For LaVAN, SemSegAdvPatch, Patch-Wise, and AdvTransfer, the attacker must have access to both the SS model they are attacking and the dataset on which it was trained. DiAP stands out from the others because it uses a dataset independent approach that only needs access to the victim SS model architecture. With these differences kept in mind, it is worth noting that all patches optimized in our experiments operate under a white-box setting where the attacker has access to the custom trained victim

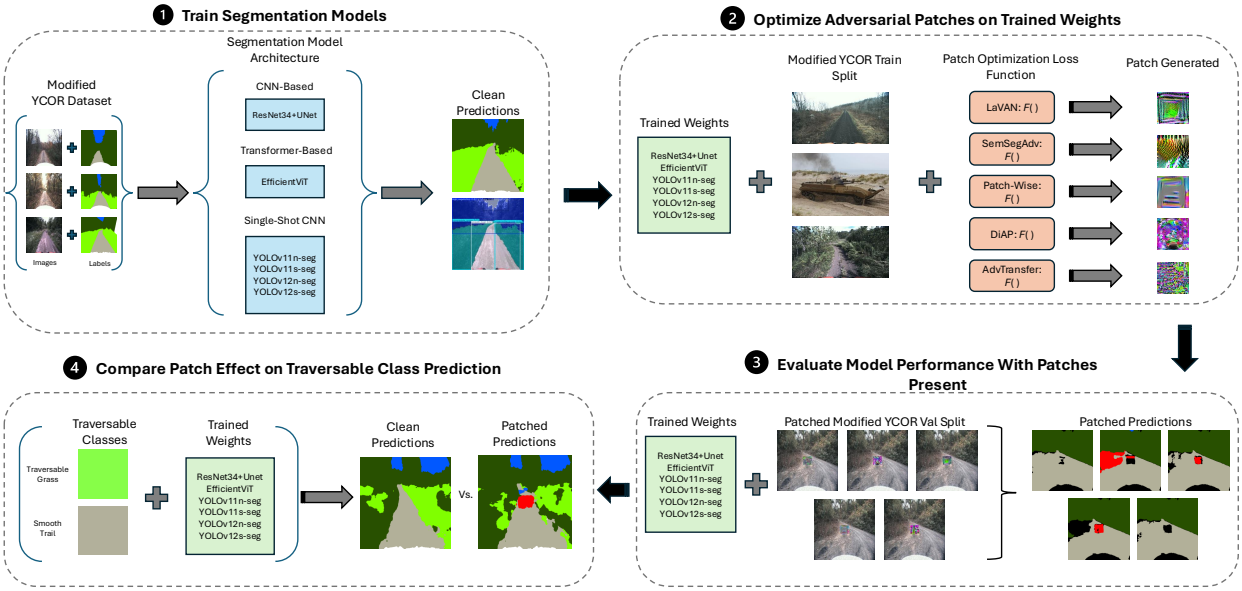


Figure 1: Robustness Evaluation of Segmentation Models

model, as well as the dataset it was trained on.

3.2. Systems Perspective

From a systems level perspective we assume that the attacker has the ability to inject digital adversarial patches into the victims camera feed. This assumption aligns with recent research demonstrating that attackers can compromise over-the-air (OTA) update mechanisms or implant malware to execute runtime stealthy perception attacks, which covertly modify camera image frames before they reach the deep neural network [27]. By executing a video injection attack, the adversary bypasses the physical camera sensor entirely, feeding manipulated digital content directly in the perception pipeline.

In our evaluations, these digital patches are injected in a fixed position at the center of the benign image. The attacker does not need to deploy the patches in the physical environment, as such physical patch deployment requires generating high-quality printed patches and predicting the target vehicle's exact driving path for effective placement.

4. METHODOLOGY

Figure 1 provides a comprehensive overview of our experimental pipeline, detailing the four primary stages of our evaluation. In this section, we evaluate five diverse adversarial patch optimization schemes against six real-time segmentation models spanning hybrid CNN, Vision Transformer, and single-stage architectures. To evaluate the effect of a patch on a segmentation model, we measure the percent of unperturbed pixels that are misclassified compared to the ground truth clean prediction. We specifically focus on pixels located entirely outside the patched region. This spatial evaluation approach adapts the methodology established by Nesti *et al.* [13], who similarly isolated model degradation by computing the mean Intersection over Union (IoU) exclusively on regions outside the adversarial patch.

4.1. Experimental Setup

To systematically evaluate the robustness of the proposed SS models in off-road environments, our experimental setup follows the four-stage pipeline illustrated in Figure 1. First, we train the selected segmentation models on the modified YCOR dataset (Step 1). Next, we optimize five adversarial patches (LaVAN, SemSegAdvPatch, PatchWise, DiAP, and AdvTransferPatch) directly on the trained model weights (Step 2). We then evaluate the performance of the selected models when these optimized patches are introduced to images from the validation split of the modified YCOR dataset (Step 3). Finally, we isolate and compare the specific effect of these patches on the models' ability to accurately predict traversable classes (Step 4). As part of this final evaluation, we also vary the size of the most devastating patch for each model to mirror the setup used in an earlier on-road evaluation study provided by Rossolini *et al.* [19], to provide a direct comparison of our off-road results against established on-road results.

4.2. Dataset Description

To execute the training and evaluation phases of our pipeline (Steps 1 and 3), we utilized the modified YCOR dataset, first provided by Rahi *et al.* [1]. The dataset contains eight segmentation classes: Unlabeled, Background, High Vegetation, Traversable grass, Traversable trail, Obstacle, Sky, and Puddle. The main modifications made to the original YCOR dataset by [1] were re-annotating mislabeled images, class reduction, data addition, and data preprocessing and augmentation to achieve a more balanced representation between drivable classes and obstacles. Notably, this is the same dataset originally used to train the hybrid ResNet34+UNet architecture proposed in [1] for off-road environments. The modified YCOR dataset contains a total of 15,414 image and label pairs. 1,541 image and label pairs were used for the validation set. 13,873 image and label pairs were used for the training set. To prevent overfitting, an early stopping mechanism was employed during the training phase. Training convergence was reached, and the process was halted, if the model's performance on the validation set showed no improvement for 10 consecutive epochs. Table 2 details the number of instances for each segmentation class in the training and validation splits.

4.3. Victim Segmentation Models

To give a fair comparison of how various adversarial patches can affect segmentation models, we chose to evaluate their effects on six different models. The chosen models were ResNet34+UNet [1], EfficientViT [28], YOLOv11n-seg, YOLOv11s-seg [3], YOLOv12n-seg, and YOLOv12s-seg [4].

ResNet34+UNet for off-road SS was first proposed and evaluated by Rahi et al. [1]. ResNet34+UNet deploys a conventional encoder-decoder CNN architecture by utilizing ResNet [29] as its encoder and UNet [30] as the decoder that yields dense pixel-wise predictions.

EfficientViT is a Vision Transformer based segmentation model first proposed by Cai et al. [28]. Its feasibility for real-time inference in off-road scenarios was subsequently validated by Pickeral et al. [2]. EfficientViT is a hybrid CNN-Vision Transformer model that is designed to make high resolution predictions with very low-latency.

YOLOv11/12(n/s)-seg are single stage detectors that deploy a CNN backbone with a segmentation head.

Class	Total dataset	Train split	Val split
Total images	**15414**	**13873**	**1541**
unlabeled	962	867	95
background	6174	5534	640
high_vegetation	12627	11377	1250
traversable_grass	8649	7776	873
smooth_trail	12003	10799	1204
obstacle	2557	2275	282
sky	7357	6613	744
puddle	257	231	26

Table 2: Number of images in which each semantic class appears for the full modified YCOR dataset, and the train and validation splits, along with the total number of labeled images in each split.

4.4. Adversarial Patch Methods

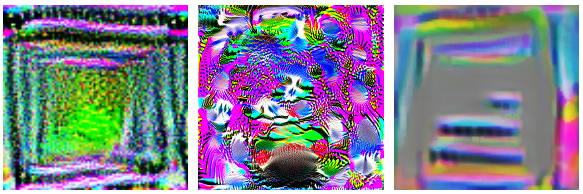
During the patch optimization phase (Step 2), we optimize five selected adversarial patch schemes (LaVAN, SemSegAdvPatch, Patch-Wise, DiAP, and AdvTransferPatch) specifically for each victim model. For the subsequent evaluation phase (Step 3), we also introduce a baseline un-optimized "DummyPatch" to compare against the optimized patches. Figure 2 provides a side-by-side visualization of each patch used in our experiments. LaVAN is a localized, visible patch that can fool a classifier even when placed outside of the main object region [5]. SemSegAdvPatch utilizes a novel Expectation over Transformation (EOT) formula to optimize itself for SS models [13]. Patch-Wise focuses on creating pixel-wise noise so that it has strong transferability [12]. DiAP crafts data-free patches without the need for the model's training set [11]. AdvTransferPatch refers to the patch presented by Shekhar *et al.* [14] that focuses on cross-model transferability. The optimization problem for generating an adversarial patch δ across all five schemes can be described by the following general formulation:

$$\min_{\delta} \mathbb{E}_{x\sim\mathcal{D},t\sim\mathcal{T}} \left[\mathcal{L}_{\text{adv}}\big(f(\mathcal{A}(x,\delta,t)),y\big) + \lambda\mathcal{L}_{\text{reg}}(\delta)\right] \tag{1}$$

Where:

- f: The target deep neural network.
- $\mathbb{E}$: The expected value computed across the sampled distributions.
- $\mathcal{D}$: The data distribution from which clean input images x are sampled.
- $\mathcal{T}$: The distribution of transformations t applied to the patch to ensure robustness (e.g., Expectation over Transformation).
- $\mathcal{A}(x,\delta,t)$: The application function that applies patch δ to input x under transformation t.
- y: The label (ground truth for untargeted, target class for targeted).
- $\mathcal{L}_{\text{reg}}$: Regularization term (e.g., Non-Printability Score, Total Variation).

The specific adversarial loss term $\mathcal{L}_{\text{adv}}$ varies by the implemented patch optimization scheme.



(a) LaVAN **(b)** DiAP **(c)** Patch-Wise

(d) SemSegAdv **(e)** AdvTransfer **(f)** Dummy

Figure 2: Six Different Patches Used in Evaluation (Optimized for YOLOv12n-seg)

4.5. Implementation Details

Each patch used in our experiments was optimized specifically for the model it was evaluated on. To determine the initial vulnerability of each model, all generated patches were given fixed dimensions of 90x90 pixels and were optimized in an equivalent manner across each model. Input images are resized to 640x640 pixels for evaluation, so patch regions cover approximately 2% of the total image area. The optimized patches are placed in the center of a scene from the modified YCOR validation set. This static placement was chosen because the off-road trail scenes in the YCOR dataset predominantly converge toward the center of the camera's perspective. Our dataset analysis confirms this strategic placement, revealing that a fixed 90x90 pixel central patch successfully overlays traversable terrain classes in 66.51% of the validation images, serving as a highly efficient attack vector that effectively affects the vehicle's perception of the drivable area without the need for complex dynamic placement algorithms.

Once all patch schemes were evaluated, we assessed how the most devastating patch for each model degraded its ability to discern traversable from non-traversable classes. Subsequently, we performed an additional evaluation with these specific patches, dynamically scaling the patches to relative area thresholds of 2.15% (Small), 3.81% (Medium), and 8.58% (Large) to directly compare our results with earlier on-road evaluations conducted by Rossolini *et al.* [19]. Clean predictions of the validation images are used as the ground truth prediction to compare the patched image predictions against.

(a) Clean Image **(b)** Patched Image

Figure 3: Clean Image Vs Patched Image

4.6. Evaluation Metrics

Similarly to the accuracy measurement tactic presented in [13], we focus on the accuracy at the pixel level of the segmentation predictions that occur outside the patch region. We report the percent of pixels accurately segmented in the non-patch region across clean images and patched images, and then measure the accuracy drop percent between the two. It makes more sense to measure the degradation of pixel accuracy that takes place outside of the patch region since the patch itself would be considered out-of-distribution data, and is already prone to low prediction accuracy in SS models. Figure 3 shows an example of how adversarial patches were applied to the scene for evaluation.

Table 3: Inference speed and resource usage for all evaluated models at 544×1024 resolution. All models are evaluated in FP32. Results are averaged over 200 timed iterations on a single Nvidia V100 GPU.

Model	Time (ms)	FPS	Params (M)	Mem (MB)
ResNet34+UNet	12.30	81.3	24.44	329.9
EfficientViT-b0	4.65	215.3	0.71	65.0
YOLOv11n-seg	8.20	121.9	2.84	67.4
YOLOv11s-seg	9.00	111.1	10.07	135.9
YOLOv12n-seg	11.96	83.6	2.81	94.4
YOLOv12s-seg	12.32	81.2	9.77	160.6

5. RESULTS

5.1. Baseline Performance and Viability

Traditional semantic segmentation models such as ResNet34+UNet and EfficientViT have been proposed as viable models for off-road AV scenarios. YOLO segmentation variants should also be considered as viable segmentation models for time sensitive AV off-road scenarios. Through our evaluations, YOLOv11n-seg, YOLOv11s-seg, YOLOv12n-seg, and YOLOv12s-seg all showed a high prediction accuracy of above 90% on clean images. We evaluated the inference speed of all YOLO models used in our evaluations compared to the other two SS models to show their feasibility as real-time SS models that can be used in AV environments. Table 3 shows a direct comparison of inference speed, FPS, number of parameters, and peak GPU memory usage between the models. From this evaluation, we can see that YOLOv11 and YOLOv12 segmentation models demonstrate comparable inference speeds to other proposed SS models. YOLOv11n-seg, YOLOv11s-seg, and YOLOv12n-seg all demonstrated faster inference speeds compared to ResNet34+UNet. YOLOv11n-seg had the fastest average inference time at 8.20 ms.

5.2. Overall Vulnerability and Pixel Accuracy Degradation

Building upon the clean baseline performance and real-time viability established in Section 5.1, we evaluated the security impact of the optimized adversarial patches. To isolate the effect of the adversarial perturbations, we measured the percentage of accurately segmented pixels strictly outside the patched region. The patch area was excluded from this metric because localized adversarial noise would be considered out-of-distribution data that inherently resists correct classification. Table 4 details the pixel accuracy degradation caused by each adversarial patch scheme across every segmentation model.

When introduced to the optimized patches, the architectures exhibited stark differences in their overall robustness. EfficientViT proved to be the most robust model in our evaluation, experiencing a maximum accuracy drop of only 3.52% when introduced to Patch-Wise. The ResNet34+UNet hybrid model demonstrated the second highest resilience, experiencing its largest degradation of 8.65% from AdvTransferPatch.

In contrast, the single-stage detectors exhibited severe vulnerabilities to adversarial patches. YOLOv11n-seg was the most susceptible architecture among all the models we evaluated. Despite its strong baseline

Table 4: Percent of pixels outside the patch region that are accurately segmented under patch attacks on the modified YCOR dataset (value in parentheses is the drop vs. clean; ↓ = lower is worse). Largest drops in pixel accuracy per model are shown in bold. Reworked template from [23].

Patch Attack	ResNet34+ UNet	Efficient ViT	YOLOv11n-seg	YOLOv11s-seg	YOLOv12n-seg	YOLOv12s-seg
Clean Sample	87.69	96.85	92.88	90.99	95.97	94.35
Dummy Patch	87.01 (0.68↓)	96.23 (0.62↓)	92.27 (0.61↓)	90.40 (0.59↓)	95.03 (0.94↓)	93.72 (0.63↓)
LaVAN	85.59 (2.10↓)	95.81 (1.04↓)	**66.33 (26.55↓)**	82.87 (8.12↓)	**87.13 (8.84↓)**	85.79 (8.56↓)
SemSegAdvPatch	85.86 (1.83↓)	96.11 (0.74↓)	88.33 (4.55↓)	**73.24 (17.75↓)**	91.78 (4.19↓)	**75.37 (18.98↓)**
Patch-Wise	86.94 (0.75↓)	**93.32 (3.53↓)**	90.61 (2.27↓)	89.80 (1.19↓)	92.87 (3.10↓)	92.32 (2.03↓)
DiAP	85.47 (2.22↓)	95.05 (1.80↓)	89.52 (3.36↓)	87.78 (3.21↓)	93.81 (2.16↓)	92.62 (1.73↓)
AdvTransferPatch	**79.04 (8.65↓)**	96.31 (0.54↓)	91.20 (1.68↓)	89.42 (1.57↓)	93.87 (2.10↓)	92.23 (2.12↓)

Table 5: Traversable Class Prediction Accuracy Drop Per Model (Outside Patch Region)

		Traversable Grass			Traversable Trail			
Model	**Method**	**Clean**	**Patch**	**Drop**	**Clean**	**Patch**	**Drop**	**Avg Drop**
YOLOv11n	LaVAN	83.88%	57.02%	26.86%	97.15%	52.37%	44.78%	**35.82%**
YOLOv11s	SemSeg	79.28%	58.87%	20.41%	96.05%	52.38%	43.67%	**32.04%**
YOLOv12n	LaVAN	94.85%	70.35%	24.50%	98.29%	89.01%	9.28%	**16.89%**
YOLOv12s	SemSeg	88.84%	55.68%	33.16%	97.72%	59.19%	38.53%	**35.85%**
ResNet34+UNet	AdvPatch	71.80%	81.59%	-9.79%	94.73%	88.86%	5.87%	**-1.96%**
EfficientViT	P-Wise	95.75%	89.50%	6.25%	99.13%	95.41%	3.72%	**4.99%**

performance, when introduced to the LaVAN patch it experienced a catastrophic 26.55% drop in overall pixel accuracy for this model. The newer YOLOv12n-seg model demonstrated improved architectural resilience compared to its predecessor, limiting its maximum accuracy drop to 8.84% under an identical LaVAN patch attack.

5.3. Impact on Traversable Terrain

While overall pixel degradation highlights the broad vulnerability of these models, the practical threat to autonomous vehicles lies in how these errors impact path-finding capabilities. For off-road autonomous navigation, the primary security concern is the ability of a perception model to reliably identify traversable terrain from non-traversable terrain during an active attack. Table 5 details the specific impact of the most damaging patch for each model on traversable class predictions.

The overall accuracy drops observed in vulnerable models directly translate to potential critical navigation failures. Under the LaVAN attack, the accuracy of YOLOv11n-seg for the Traversable Grass class dropped from 83.88% to 57.02%. Furthermore, its accuracy for the Traversable Trail class dropped from 97.15% to 52.37%. These results demonstrate how the localized placement of an adversarial patch fundamentally disrupts the ability of models to classify traversable terrain across the entire visual scene.

Model	Small (2.15% area)				Medium (3.81% area)				Large (8.58% area)			
	Clean		Patched		Clean		Patched		Clean		Patched	
	mIoU	mAcc	mIoU	mAcc	mIoU	mAcc	mIoU	mAcc	mIoU	mAcc	mIoU	mAcc
YOLOv11n	0.731	0.808	0.446	0.633	0.731	0.808	0.361	0.552	0.731	0.809	0.336	0.527
YOLOv11s	0.576	0.665	0.479	0.597	0.576	0.665	0.426	0.556	0.577	0.665	0.338	0.495
YOLOv12n	0.826	0.877	0.612	0.766	0.826	0.877	0.435	0.620	0.826	0.877	0.352	0.537
YOLOv12s	0.740	0.795	0.617	0.710	0.740	0.795	0.544	0.646	0.740	0.795	0.484	0.607
ResNet34UNet	0.564	0.663	0.462	0.619	0.564	0.663	0.514	0.641	0.565	0.663	0.527	0.638
EfficientViT	0.889	0.922	0.798	0.865	0.889	0.923	0.761	0.835	0.890	0.924	0.668	0.770

Table 6: Overall-patch mIoU and mAcc for clean vs. patched evaluations across models and patch sizes. Patched variants: YOLOv11n: LaVAN; YOLOv11s: SemSegAdv; YOLOv12n: LaVAN; YOLOv12s: SemSegAdv; ResNet34UNet: AdvTransferPatch; EfficientViT: Patch-Wise.

5.4. Off-Road Results Compared to Prior On-Road Study

Rossolini *et al.* [19] evaluated universal adversarial patches of varying sizes on semantic segmentation models in on-road environments, reporting mIoU and mAcc under a standardized evaluation protocol. To compare our off-road findings to theirs, we mirrored their relative patch-size settings, specifically patches covering 2.15% (Small), 3.81% (Medium), and 8.58% (Large) of the total image area, as outlined in Section 4.5. However, rather than replicating their on-road placement strategy, we evaluated each model under clean and patched conditions using a single centrally placed patch, which better aligns with the visual convergence of off-road trails. Following [19], we report mIoU and mAcc computed strictly on out-of-patch pixels so that the observed degradation reflects patch-induced misclassification rather than physical occlusion.

Table 6 summarizes these results for each model and patch size. Our off-road results corroborate the main trend reported by Rossolini *et al.* for on-road scenes, which shows that adversarial patches induce a consistent degradation in segmentation performance that increases with patch size, even when evaluation excludes patch pixels.

6. DISCUSSION

Interestingly, in a few special cases, pixel accuracy outside the patch region actually increased upon the introduction of an adversarial patch. As illustrated with EfficientViT in Figure 4, this anomaly occurs when an adversarial perturbation, in this case AdvTransferPatch, inadvertently corrects baseline misclassifications. A similar phenomenon was observed when the LaVAN patch was applied to the ResNet34+UNet hybrid model. In that instance, the clean model initially misclassified a large portion of the High Vegetation class as Traversable Grass, and the introduction of the patch caused the model to revert this region back to its ground-truth High Vegetation label. These instances imply that while these architectures effectively segment distinct drivable surfaces like dirt trails, they inherently struggle to visually differentiate between similar textures, such as traversable grass and non-traversable high vegetation.

7. LIMITATIONS

Although our evaluations demonstrate the severe vulnerabilities of single-stage models in off-road environments, several limitations exist. First, our threat model restricts patch injection to a fixed central location. A more sophisticated digital attacker could uti-

(a) Ground-truth labels

(b) Prediction (clean)

(c) Prediction with AdvTransferPatch

Figure 4: Comparison of segmentation results for EfficientViT when accuracy increases were recorded.

lize gradient-based search methods to dynamically place the patch over critical navigational features, likely inducing even higher pixel accuracy degradation [31] [32]. Second, because off-road semantic segmentation models inherently struggle with domain generalization [33]; the transferability of these specific adversarial patches across diverse, out-of-distribution environments remains an open question. Third, this work focuses exclusively on attack efficacy in a digital context; the feasibility of physically realizable patches and the effectiveness of proposed adversarial defense mechanisms, such as Fast Patch Detection Algorithm (FPDA), were not investigated and are left for future work.

8. CONCLUSION

In this work, we evaluated the robustness of six unique segmentation models against five different adversarial patch optimization schemes. EfficientViT proved to be the most robust architecture evaluated, maintaining both the highest clean prediction accuracy (96.85%) and the highest resilience against adversarial patches. While lightweight single-stage models like YOLOv11-seg and YOLOv12-seg are viable for real-time off-road navigation due to their competitive inference speeds, they exhibit a significantly higher vulnerability to localized perturbations than hybrid or Transformer-based models. Specifically, YOLOv11n-seg demonstrated the least robustness, suffering a 26.55% accuracy drop when introduced to the LaVAN patch. YOLOv12n-seg emerged as a superior alternative to YOLOv11n-seg for off-road scenarios, offering improved robustness-to-latency ratios.

Furthermore, this study highlights that off-road perception systems face unique risks; unlike on-road models that prioritize traffic sign recognition, off-road models are susceptible to attacks that specifically degrade the identification of traversable terrain, such as low vegetation and trails. The robustness of the segmentation model is crucial for time-sensitive AV scenarios and should be heavily considered when deciding on which model an AV should implement.

ACKNOWLEDGMENTS

This work was supported by Clemson University's Virtual Prototyping of Autonomy Enabled Ground Systems (VIPR-GS), under Cooperative Agreement W56HZV-21-2-0001 with the US Army DEVCOM Ground Vehicle Systems Center (GVSC).

References

[1] A. Rahi, O. Elgeoushy, S. H. Syed, H. El-Mounayri, H. Wasfy, T. Wasfy, and S. Anwar, "Deep semantic segmentation for identifying traversable terrain in off-road autonomous driving," *IEEE Access*, 2024.

[2] A. Pickeral, E. Marquez, M. C. Smith, J. C. Calhoun, *et al.*, "Efficient vision transformers for autonomous off-road perception systems," *Journal of Computer and Communications*, vol. 12, no. 09, 2024.

[3] Ultralytics, "Ultralytics YOLOv11 Segmentation Documentation." https://docs.ultralytics.com/models/yolo11/, 2024. Accessed: 23 November 2025.

[4] Ultralytics, "Ultralytics YOLOv12 documentation." https://docs.ultralytics.com/models/yolo12/, 2024. Official online documentation for Ultralytics YOLOv12 models. Accessed: 2025-11-19.

[5] D. Karmon, D. Zoran, and Y. Goldberg, "Lavan: Localized and visible adversarial noise," in *International conference on machine learning*, pp. 2507–2515, PMLR, 2018.

[6] T. B. Brown, D. Mané, A. Roy, M. Abadi, and J. Gilmer, "Adversarial patch," 2018.

[7] Y. Huang, Y. Ren, J. Wang, L. Huo, X. Bai, J. Zhang, and H. Yu, "Advreal: Adversarial patch generation framework with application to adversarial safety evaluation of object detection systems," *arXiv preprint arXiv:2505.16402*, 2025.

[8] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3431–3440, 2015.

[9] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788, 2016.

[10] Ultralytics, "Ultralytics YOLOv5 documentation." https://docs.ultralytics.com/models/yolov5/, 2020. Accessed: 2025-11-19.

[11] X. Zhou, Z. Pan, Y. Duan, J. Zhang, and S. Wang, "A data independent approach to generate adversarial patches," *Machine Vision and Applications*, vol. 32, no. 3, p. 67, 2021.

[12] L. Gao, Q. Zhang, J. Song, X. Liu, and H. T. Shen, "Patch-wise attack for fooling deep neural network," in *European Conference on Computer Vision*, pp. 307–322, Springer, 2020.

[13] F. Nesti, G. Rossolini, S. Nair, A. Biondi, and G. Buttazzo, "Evaluating the robustness of semantic segmentation for autonomous driving against real-world adversarial patch attacks," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2280–2289, 2022.

[14] P. Shekhar, B. Devkota, D. Samaraweera, L. N. Kandel, and M. Babu, "Do adversarial patches generalize? attack transferability study across real-time segmentation models in autonomous vehicles," in *2025 IEEE*

Security and Privacy Workshops (SPW), pp. 322–328, IEEE, 2025.

[15] K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, C. Xiao, A. Prakash, T. Kohno, and D. Song, "Robust physical-world attacks on deep learning visual classification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1625–1634, 2018.

[16] M. Treu, T.-N. Le, H. H. Nguyen, J. Yamagishi, and I. Echizen, "Fashion-guided adversarial attack on person segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 943–952, 2021.

[17] C. Xie, J. Wang, Z. Zhang, Y. Zhou, L. Xie, and A. Yuille, "Adversarial examples for semantic segmentation and object detection," in *Proceedings of the IEEE international conference on computer vision*, pp. 1369–1378, 2017.

[18] J. Hendrik Metzen, M. Chaithanya Kumar, T. Brox, and V. Fischer, "Universal adversarial perturbations against semantic image segmentation," in *Proceedings of the IEEE international conference on computer vision*, pp. 2755–2764, 2017.

[19] G. Rossolini, F. Nesti, G. D'amico, S. Nair, A. Biondi, and G. Buttazzo, "On the real-world adversarial robustness of real-time semantic segmentation models for autonomous driving," *IEEE Transactions on Neural Networks and Learning Systems*, 2023.

[20] A. Kurakin, I. Goodfellow, and S. Bengio, "Adversarial machine learning at scale," *arXiv preprint arXiv:1611.01236*, 2016.

[21] B. Li, J. Xu, S. Wu, S. Ding, J. Li, and F. Huang, "Detecting adversarial patch attacks through global-local consistency," in *Proceedings of the 1st International Workshop on Adversarial Learning for Multimedia*, pp. 35–41, 2021.

[22] M. Yatsura, K. Sakmann, N. G. Hua, M. Hein, and J. H. Metzen, "Certified defences against adversarial patch attacks on semantic segmentation," *arXiv preprint arXiv:2209.05980*, 2022.

[23] Z. Wang, B. Wang, C. Zhang, and Y. Liu, "Defense against adversarial patch attacks for aerial image semantic segmentation by robust feature extraction," *Remote Sensing*, vol. 15, no. 6, p. 1690, 2023.

[24] P. Deoli, R. Kumar, A. Vierling, and K. Berns, "Evaluating the robustness of off-road autonomous driving segmentation against adversarial attacks: A dataset-centric analysis," *arXiv preprint arXiv:2402.02154*, 2024.

[25] A. Kurakin, I. J. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," in *Artificial intelligence safety and security*, pp. 99–112, Chapman and Hall/CRC, 2018.

[26] N. Wang, S. Xie, T. Sato, Y. Luo, K. Xu, and Q. A. Chen, "Revisiting physical-world adversarial attack on traffic sign recognition: A commercial systems perspective," *arXiv preprint arXiv:2409.09860*, 2024.

[27] X. Zhou, A. Chen, M. Kouzel, H. Ren, M. McCarty, C. Nita-Rotaru, and H. Alemzadeh, "Runtime stealthy perception attacks against dnn-based adaptive cruise control systems," in *Proceedings of the 20th ACM Asia Conference*

on Computer and Communications Security, pp. 1065–1082, 2025.

[28] H. Cai, J. Li, M. Hu, C. Gan, and S. Han, "Efficientvit: Multi-scale linear attention for high-resolution dense prediction," *arXiv preprint arXiv:2205.14756*, 2022.

[29] 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.

[30] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241, Springer, 2015.

[31] S. Kimhi, M. Kimhi, and A. Mendelson, "Benchmarking adversarial patch selection and location," *Mathematics*, vol. 14, no. 1, p. 103, 2025.

[32] X. Wei, Y. Guo, J. Yu, and B. Zhang, "Simultaneously optimizing perturbations and positions for black-box adversarial patch attacks," *IEEE transactions on pattern analysis and machine intelligence*, vol. 45, no. 7, pp. 9041–9054, 2022.

[33] Z. Zhao, G. Li, C. Min, and K. Lu, "Ot-drive: Out-of-distribution off-road traversable area segmentation via optimal transport," *arXiv preprint arXiv:2601.09952*, 2026.