Off-road Autonomy: Semantic Scene Segmentation

GHR 2.0 Hackathon - Desert Environment Challenge

Presented by: KINETIC MINDS

Navigating the unseen through high-fidelity digital twins

Methodology & Training Workflow

Deployed Architecture

"Implementing a **SegFormer (B0)** architecture with a lightweight backbone, optimized for rapid inference on synthetic desert textures."

Falcon Cloud Platform

Development environment leveraging EDU Conda setup for reproducible training workflows

Training Objective

"Minimizing **Cross-Entropy + Dice Loss** to handle class imbalance between dominant classes (Sky) and rare classes (Flowers/Logs)."

Hyperparameter Optimization

Fine-tuning learning rates, batch sizes, and network depth while maintaining inference efficiency

Data Overview & Class Breakdown

Synthetic Data Source

High-quality semantic labels generated via FalconEditor's digital twin platform, enabling rapid iteration without physical data collection.

Total Classes: 10 distinct terrain categories

Image Format: RGB color + segmented masks

Target Class Hierarchy

Natural Vegetation

Trees (100) • Lush Bushes (200) • Dry Grass (300) • Dry Bushes (500) • Flowers (600)

Inanimate Objects

Logs (700) • Rocks (800) • Ground Clutter (550)

Background Classes

Landscape (7100) • Sky (10000)

Performance Metrics & Benchmarking

Intersection over Union (IoU)

Primary evaluation metric measuring pixel classification accuracy. Calculated per class and averaged across all categories.

Baseline Establishment

Initial performance measurements on sample dataset enable tracking iterative improvements throughout training cycles.

Generalization Testing

Training Loss vs. Validation Loss visualization ensures models generalize effectively to unseen desert biomes.

Challenges & Strategic Solutions

Challenge: Data Scarcity

Solution: Falcon's digital twin eliminates costly real-world data collection. Synthetic environments provide infinite variation at zero marginal cost.

Challenge: Class Imbalance

Solution: We implemented a **Weighted Loss Function** that assigns a 10x higher penalty for misclassifying rare classes. This forces the model to prioritize learning small objects (like ID: 600) over the dominant background pixels.

Challenge: Context Shifts

Solution: We applied **Strong Geometric Augmentation** (random flips, rotations) and **Color Jittering**. This forces the network to learn object *textures* (e.g., the roughness of a rock) rather than relying on lighting or position.

Failure Analysis & Visual Diagnosis

Ground Truth

Accurate pixel-level annotations from FalconEditor's synthetic environment with precise class boundaries.

Model Prediction

Neural network output highlighting correct classifications and misclassification patterns requiring refinement.

Common Error Patterns

- Logs misclassified as Rocks due to similar geometric profiles
- Flowers confused with Dry Bushes under low-light conditions
- Shadow regions causing boundary ambiguity in segmentation masks
- Occlusion leading to incomplete object recognition

Reproducibility & Repository Instructions

Environment Setup

Activate EDU Conda environment and execute `setup_env.bat` to configure dependencies

Training Execution

Run `train.py` with configuration file specifying model architecture and hyperparameters

Validation Testing

Execute `test.py` on holdout dataset to reproduce final IoU results and generate confusion matrices

Code Availability: Complete implementation including data preprocessing, model definitions, and evaluation scripts available in private GitHub repository. Jupyter notebooks document training progress and failure analysis iterations.