Final Report: Road Extraction from Satellite Images

1. Introduction:

Extracting roads from satellite images is not just a technical challenge but also a step toward smarter cities and efficient disaster responses. In this project, the focus was on training a deep learning model to identify and segment roads from satellite images.

2. Methodology

2.1 Dataset Preparation

The dataset comprised satellite images and their corresponding road masks:

- Training Data: Include satellite images and Their masks.
- Test Data: Contain only images for evaluating the model's predictions.
- **We used different type of Data Augmentation**:
  - **Type 1**: No augmentation applied.
  - **Type 2**:

```
transform = transforms.Compose([
       transforms.Resize((256, 256)),
       transforms.RandomHorizontalFlip(),
       transforms.RandomRotation(10),
       transforms.ColorJitter(brightness=0.2, contrast=0.2, saturation=0.2, hue=0.1),
       transforms.ToTensor(),
       transforms.Normalize(mean=[0.485, 0.456, 0.406], std=[0.229, 0.224, 0.225])
   ])
```

- **Type 3:**

```
transforms.Normalize(mean=[0.485, 0.456, 0.406], std=[0.229, 0.224, 0.225])
```

2.2 Model Architecture

For this project, the pretrained **deeplabv3_resnet101** model was selected due to its robust performance in segmentation tasks. This model integrates several key components to achieve high-quality segmentation results:

- **Feature Extraction with ResNet-101**: The backbone of the model is ResNet-101, a deep convolutional neural network that extracts rich, high-level features from the input images. This network uses layers of convolutions and pooling operations to reduce the spatial dimensions while capturing essential details.
- **Atrous Spatial Pyramid Pooling (ASPP)**: A unique feature of the DeepLab architecture, ASPP employs atrous convolutions with varying rates to capture multi-scale contextual information. This allows the model to identify objects at different scales, enhancing its segmentation capabilities.
- **Upsampling and Output**: After feature extraction and context capture, the model upsamples the lower-resolution feature maps to match the original image dimensions. This ensures that the segmentation output maintains high spatial resolution and accuracy.

2.3 Training Strategy

The training dataset was divided into an 80% training set and a 20% validation set to balance model training and evaluation. The training strategy incorporated several key techniques to optimize performance:

- **Learning Rate Scheduling**: The learning rate was dynamically adjusted during training to enhance convergence and stability.
- **Weight Decay**: A regularization method applied to prevent overfitting by penalizing large weights.

- **Cross-Entropy Loss**: This loss function was used to evaluate the performance of the model by measuring the difference between predicted and true labels.
- **Adam Optimizer**: The primary optimizer used for training, known for its adaptive learning rate capabilities.
- **StepLR Scheduler**: Occasionally used to adjust the learning rate at specific intervals, further refining the training process.
- **Number of Epochs**: Defined the total number of training iterations to ensure adequate learning.

2.4 Evaluation Process

The F1 score was chosen as the primary metric for evaluation, providing a balanced measure of precision and recall. This metric ensured that the model's predictions were both accurate (precision) and comprehensive (recall), essential for effective road detection in segmentation tasks.

3. Results:

3.1 Quantitative Metrics

L_R	Weight_decay	Batch Size	Num Epochs	Data_ Aug	Training Accuracy	Validation Accuracy	Optimizer	Loss
0.0002	0.0005	8	35	Type 2	96.31	0.8357	Normal Adam Optim	Cross-Entropy
0.0002	0.0005	64	60	Type 2	0.9491	0.8103	Normal Adam Optim	Cross-Entropy
0.0002	0.0005	32	50	Type 2	0.9578	0.7969	Normal Adam Optim	Cross-Entropy
0.0001	0.001	16	40	Type 2	0.9499	0.8155	Normal Adam Optim	Cross-Entropy
0.0002	0.0005	16	40	Type 2 except resize	0.99	0.91	Normal Adam Optim	Cross-Entropy
0.0002	0.0005	16	40	Type 1	1.0	0.98	Normal Adam Optim	Cross-Entropy
0.0002	0.0005	8	35	Type 1	0.9669	0.9247	Normal Adam Optim	Cross-Entropy
0.0002	0.0005	32	55	Type 3	0.9245	0.8940	Here we used step optimizer Step=0.01	Cross-Entropy
0.0002	0.0005	8	35	Type 3	0.9583	0.9178	Here we used step optimizer Step=0.1	Cross-Entropy
0.0002	0.0005	8	35	Type 1	0.94	0.90	Normal Adam Optim	Cross-Entropy
0.0003	0.0001	8	35	Type 3	0.9687	0.9300	Normal Adam Optim	Cross-Entropy

3.2 Qualitative Analysis

The model performed well on most test images, accurately segmenting roads. However, specific challenges included:

- Misclassifications in areas with visually similar features, such as parking lots.

- Difficulty in detecting roads obscured by trees or buildings.

4. Conclusion

This project demonstrated the effective use of the **deeplabv3_resnet101** model for road extraction from satellite imagery, achieving a strong F1 score. Data augmentation and hyperparameter tuning were key to enhancing performance. However, the model struggled with occlusions and visually similar features. When we tried data augmentation, the model's performance initially declined, so we relied on normalization instead. DeepLabv3's ability to handle any input image size without resizing simplified our workflow. With more training images and alternative loss functions like Dice Loss, further improvements could be achieved. While other models might have offered better performance, we stuck with DeepLabv3 due to its initial success. Due to time constraints and heavy coursework, we couldn't explore these options fully.