Machine Learning UE23CS352A

Airbus Ship Detection

Likhith M Irani - PES1UG23CS329
Meghan Singhal - PES1UG23CS355

The Core Problem: Semantic Segmentation

Goal: Accurately identify and delineate the exact pixel area of ships in high-resolution satellite imagery.

Why Segmentation?

This project goes beyond simple classification or detection:

- Image Classification: Is a ship present? (Too coarse)
- Object Detection: Where is the ship? (Bounding Box)
- Semantic Segmentation: What pixels belong to the ship? (Pixel-wise mask)

  This task requires a model that understands context and provides precise localization.

Dataset and Data Challenges

Dataset: Airbus Ship Detection (Kaggle Competition)

Source Data: Images in train_v2 and test_v2 folders.

Annotations: train_ship_segmentation.csv provides ground truth.

Data Size: Over 192,000 images were analyzed in the training directory.

The Key Data Challenge: The ground-truth segmentation masks are stored in a compressed format called Run Length Encoding (RLE).

RLE efficiently stores the location and length of continuous 'ship' pixels, but must be decoded into a full 2D binary mask for model training.

Model Architecture: The UNET

The UNET model is ideal for semantic segmentation due to its unique structure:

- **Encoder Path (Contraction):** Captures the context of the image (what is being seen). Uses repeated convolutions and max-pooling to downsample the image.
- **Decoder Path (Expansion):** Enables precise localization (where it is). Uses upsampling layers to reconstruct the segmentation mask.
- **Skip Connections (The Secret Sauce):** Directly connects features from the encoder to the decoder. Crucial for recovering fine-grained spatial details lost during downsampling.

Approach

1. Data Preprocessing:

- Loaded the training images and the corresponding train_ship_segmentations_v2.csv file containing RLE-encoded ship masks.

2. Model Architecture:

- Implemented a U-Net style convolutional neural network for pixel-wise segmentation. Used TensorFlow/Keras for model construction and training.

3. Training and Evaluation:

- Split the data into training and validation subsets.
- Trained the model to predict binary masks for ship regions.

4. Prediction and Submission:

- Applied the trained model to test images to generate predicted masks.

Conclusion

This project demonstrated a complete deep learning workflow for object segmentation in satellite imagery. The final model effectively detects ships and generates valid submission files for the Kaggle challenge. The techniques and insights gained can be extended to other remote sensing and image analysis problems.

Future Work

Further Evaluation: Test the model's performance on the hidden test set to get an objective measure of generalization.

Model Improvement: Explore advanced techniques such as multi-scale training, fine-tuning the UNET's hyperparameters, or experimenting with alternative segmentation models like DeepLabV3+ for potential performance gains.

Real-World Application: Apply the trained model to real-world, live satellite feeds for operational maritime monitoring, including tracking vessel traffic and detecting unlisted ships.

Efficiency Analysis: Conduct a thorough analysis of the model's inference speed to ensure it meets operational requirements for near real-time deployment.