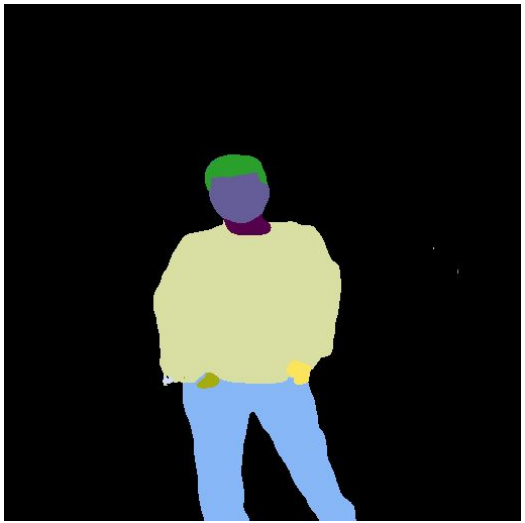
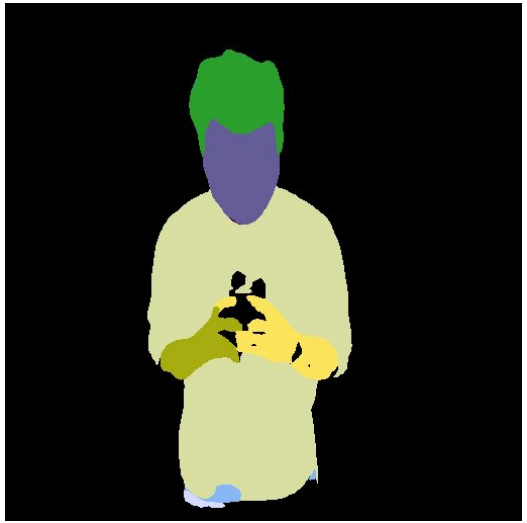
Unveiling Water Bodies through Semantic Segmentation from Satellite Images

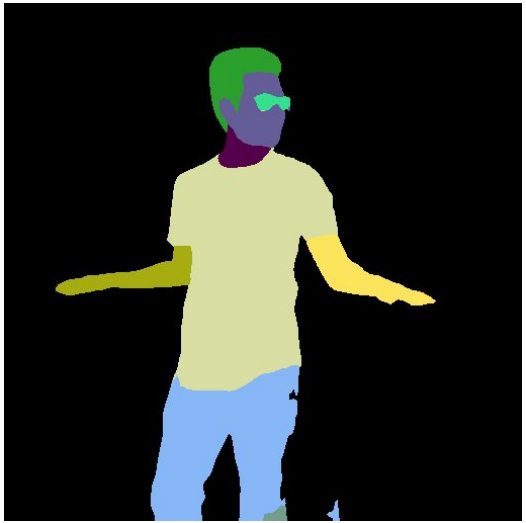
Rounak Sen, Kunal Rustagi and Mayank Sharma
Group 22



Rounak Sen



Kunal Rustagi



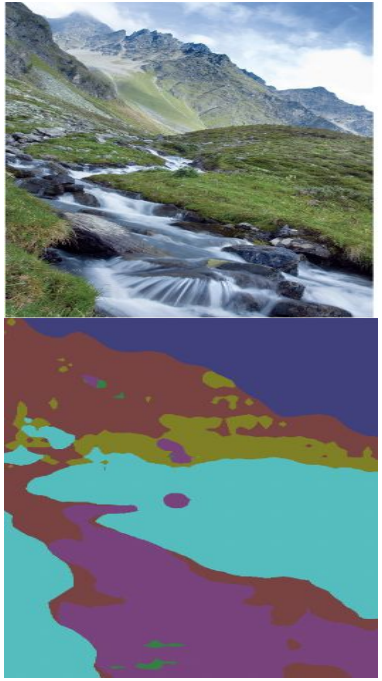
Mayank Sharma

What is semantic segmentation? How does it help with the unveiling of water bodies?

- In short, semantic segmentation means gives our machines the power of visual cognition, enabling them to understand images at a deeper level, discerning objects and their meaning
- Used in various applications such as **Object Recognition & Understanding**, **Fine-Grained Image Analysis**, **Efficient Data Annotations**, **Visual Scene Understanding**
- Why do we want to unveil water bodies using segmentation?
    - **Water Body Mapping & Monitoring**: allows for the precise water body identification and delineation
    - **Water Quality Assessment**: identify areas with different water quality characteristics, like turbidity or pollution levels
    - **Change Detection**: Tracks changes in water extent, shoreline erosion, and new water bodies by comparing segmented images over time

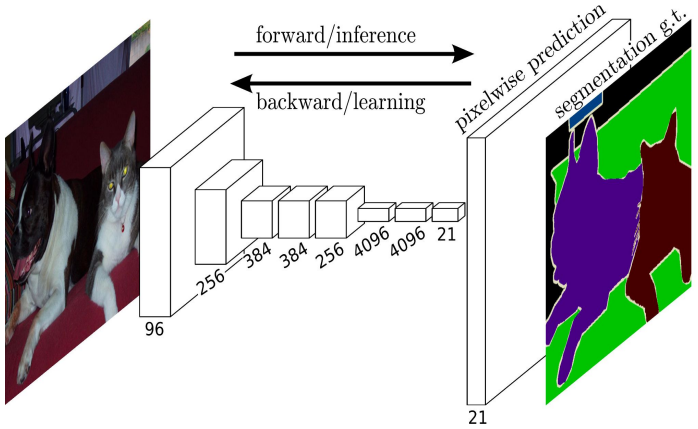
Dataset

- **ATLANTIS** is a benchmark for semantic segmentation of waterbody images.
- This dataset covers a **wide range of natural waterbodies** such as sea, lake, river and man-made (artificial) water-related structures such as dam, reservoir, canal, and pier.
- ATLANTIS includes 5,195 pixel-wise annotated images split to 3,364 training, 535 validation, and 1,296 testing images.
- Data pre-processing includes resizing the images to a specified height and width, horizontal flipping, random brightness and contrast adjustments, and random shift, scale, and rotation transformations.

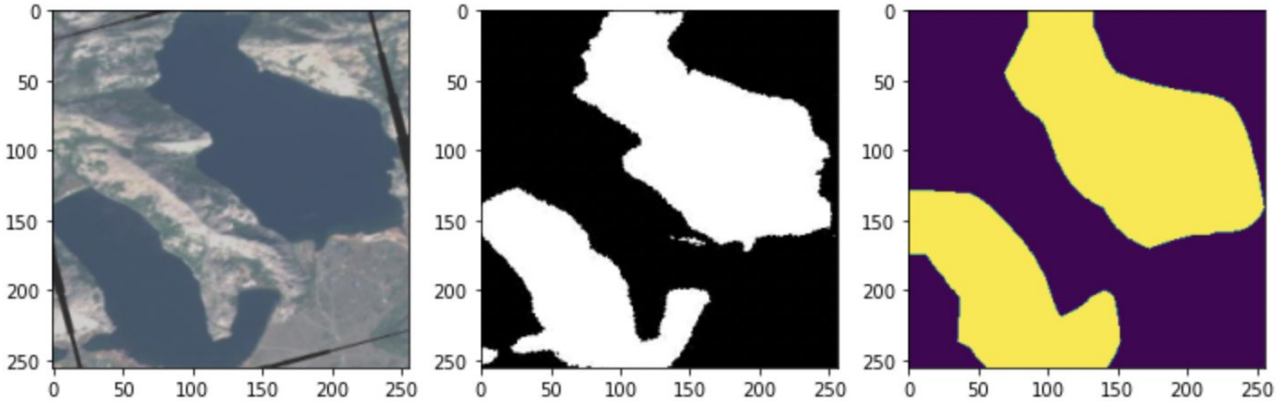


Fully-Connected Networks (FCN)

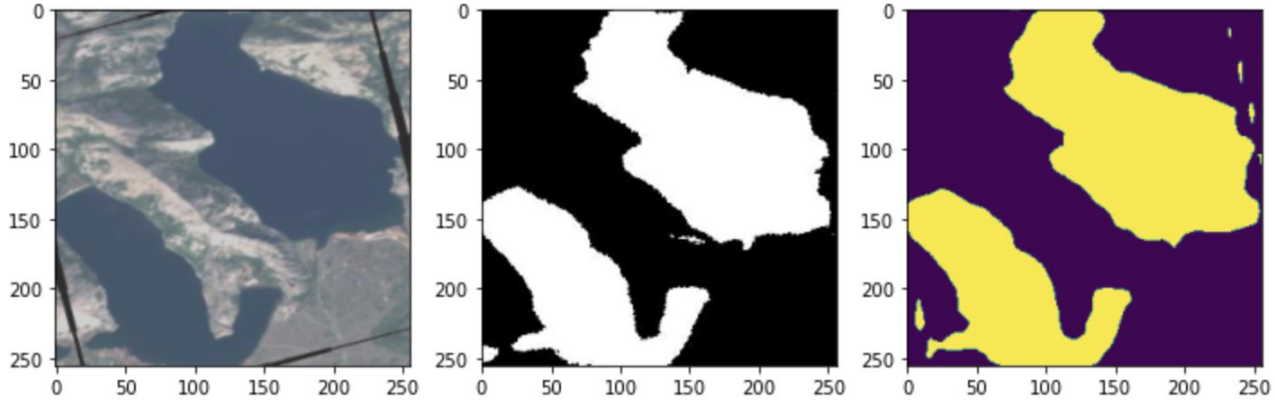
- In FCN, each neuron is connected to every neuron in the previous layer.
- Commonly used in various machine learning tasks like classification, segmentation, etc.
- FCN32 uses downsampling by a factor of 32, whereas the factor is 8 for FCN8. FCN8 uses skip connections to capture deep-level features.
- Both the models have a pre-trained VGG backbone.



FCN32:

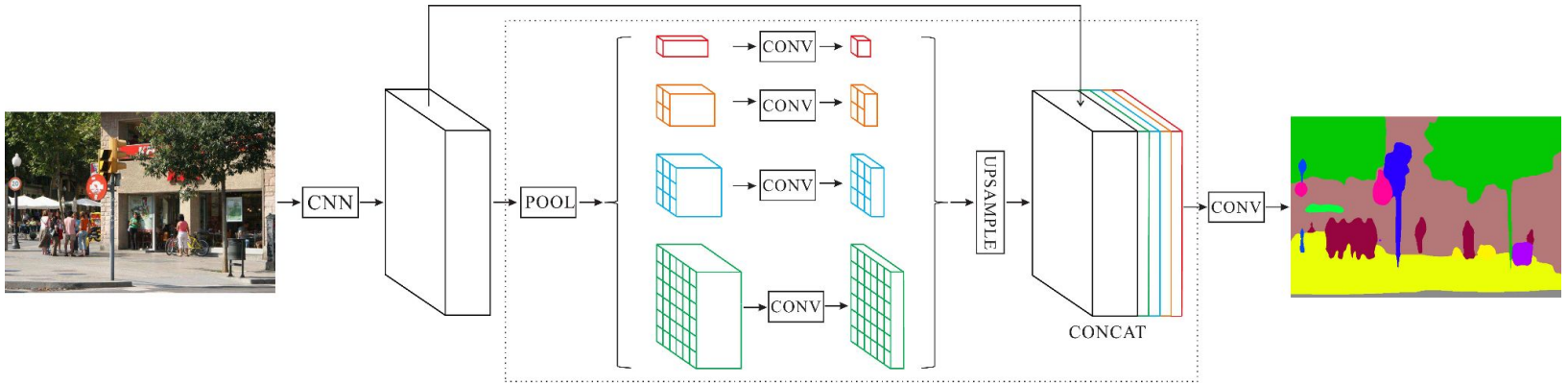


FCN8:



Pyramid Scene Parsing (PSP) Networks

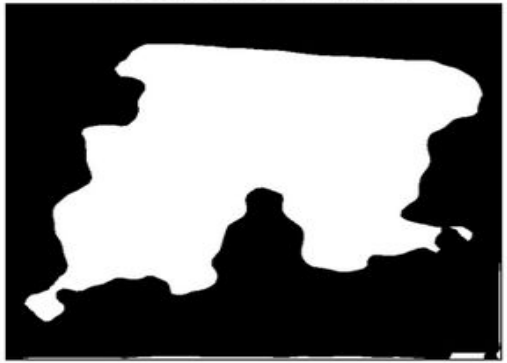
- Utilizes a pyramid parsing module that exploits global context information by different-region based context aggregation.
- Local and global clues together make the final prediction more reliable.
- More suitable for tasks that require **detailed contextual information**, such as object segmentation in crowded scenes or semantic segmentation in large-scale images.



Original Image

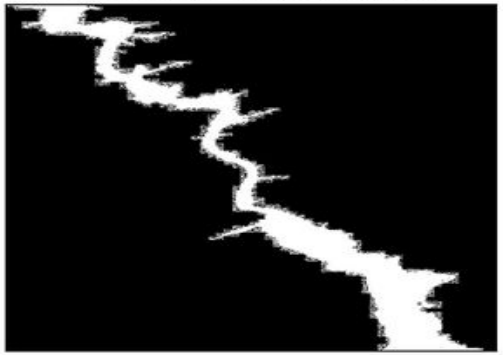
Ground Truth Mask

Predicted Mask

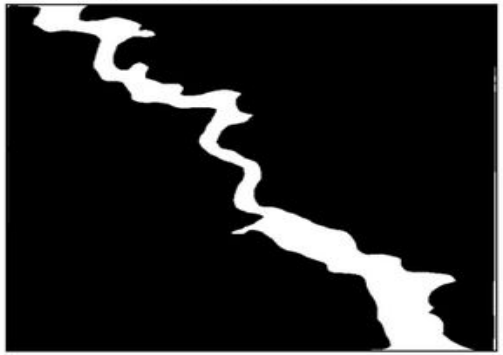


Original Image

Ground Truth Mask



Predicted Mask



U-net

- U-Net is a fully-connected convolutional neural network developed for biomedical image segmentation at the Computer Science Department of the University of Freiburg, Germany. However, its usage extends beyond this application. U-Net is particularly effective in situations where the dataset is small.

- U-Net performs better with pre-trained ResNet backbone, however this model can be quite memory intensive. The original UNet model is lighter and can be a good choice for binary semantic segmentation. We have implemented this architecture and will now explain it.

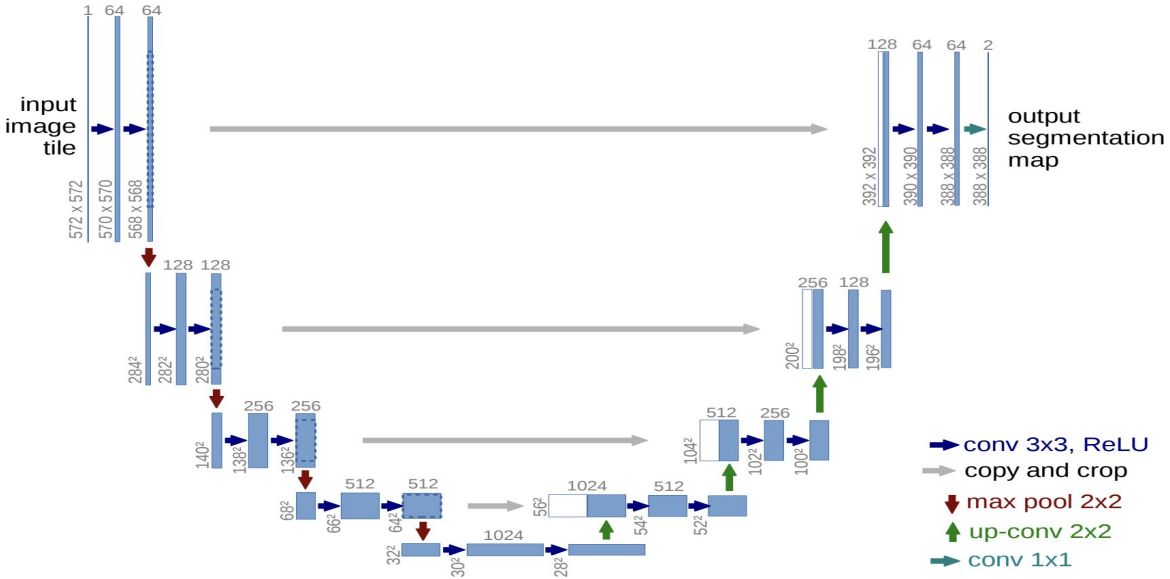
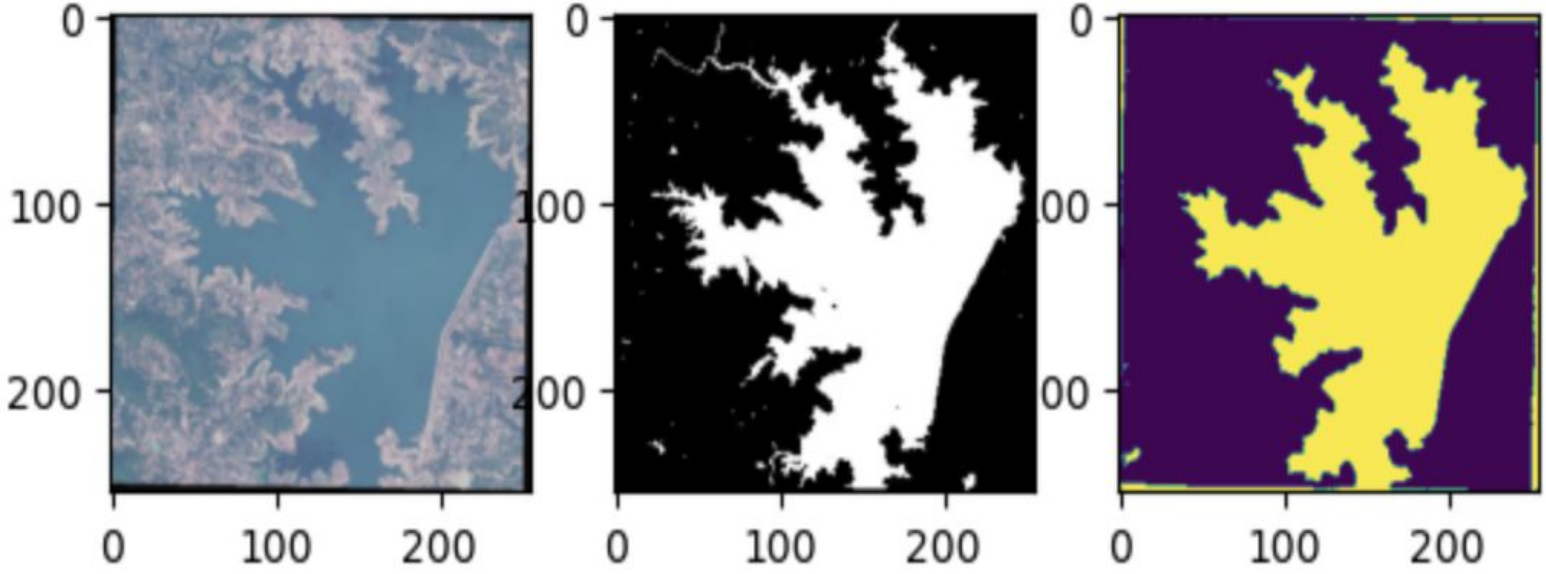
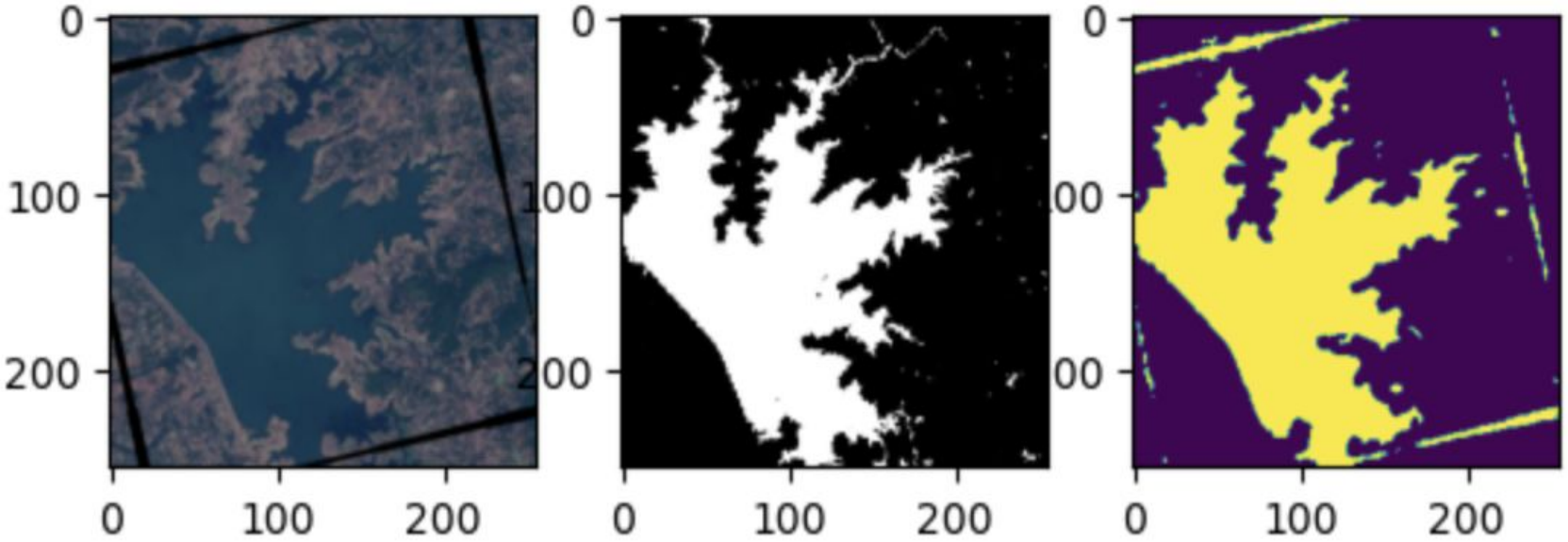


Fig. 1. U-net architecture (example for 32x32 pixels in the lowest resolution). Each blue box corresponds to a multi-channel feature map. The number of channels is denoted on top of the box. The x-y-size is provided at the lower left edge of the box. White boxes represent copied feature maps. The arrows denote the different operations.

Output of U-net from scratch



U-net pre-trained with Resnet101



Deep Lab v3

- An extension of the DeepLab semantic segmentation architecture, which is known for its robustness and accuracy in dealing with issues such as object scales and image boundaries.
- It uses Atrous Spatial Pyramid Pooling (ASPP): Effectively captures multi-scale context by employing parallel atrous convolutions with different rates.
- DeepLab v3 integrates image-level features right before the final classification layer, which helps capture long range information and improve performance.

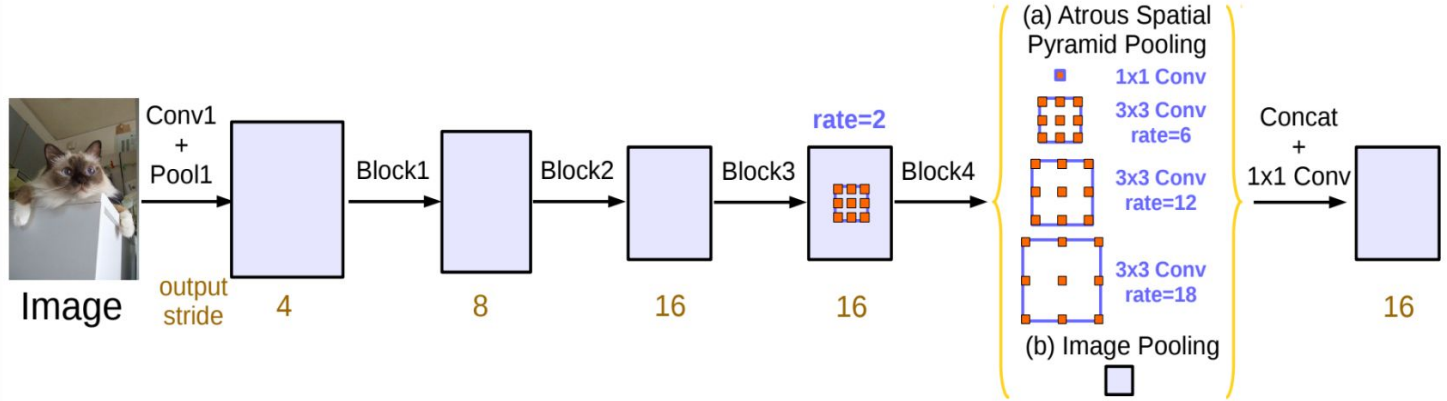
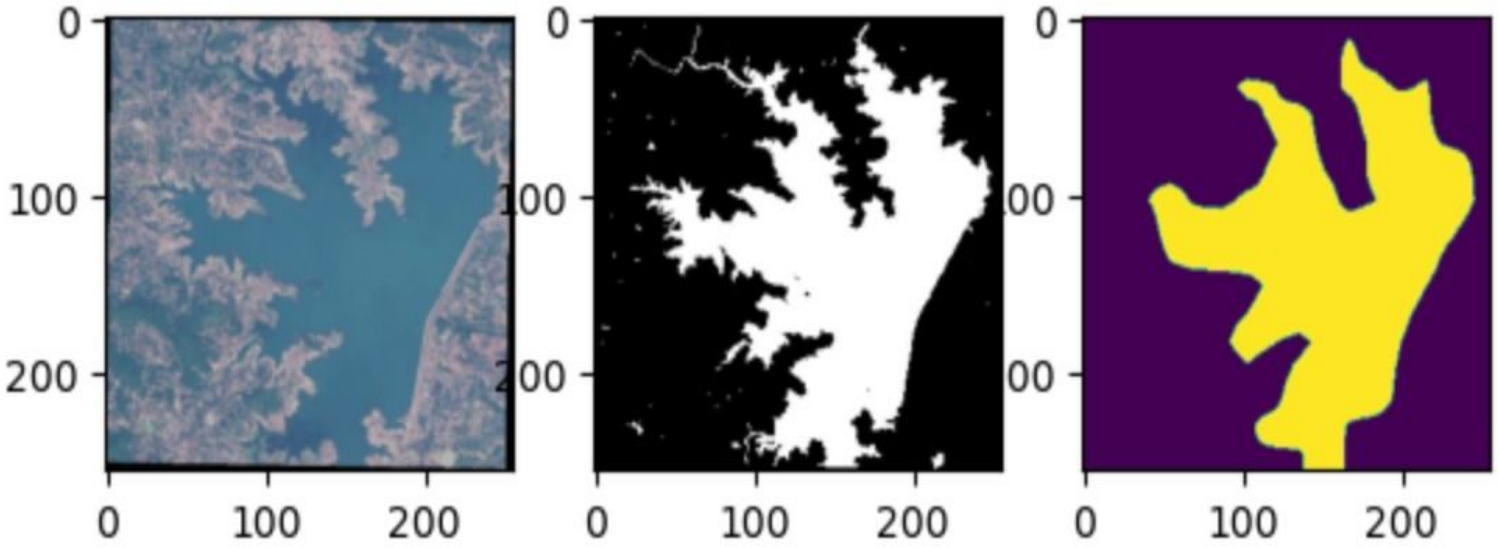


Figure 5. Parallel modules with atrous convolution (ASPP), augmented with image-level features.

Deep Lab v3 pre-trained with Resnet 50 Output



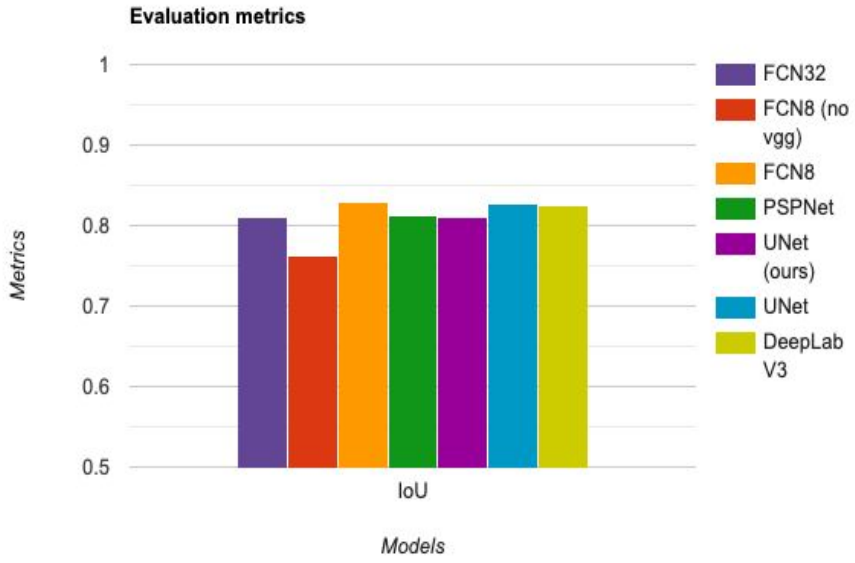
Evaluation metrics

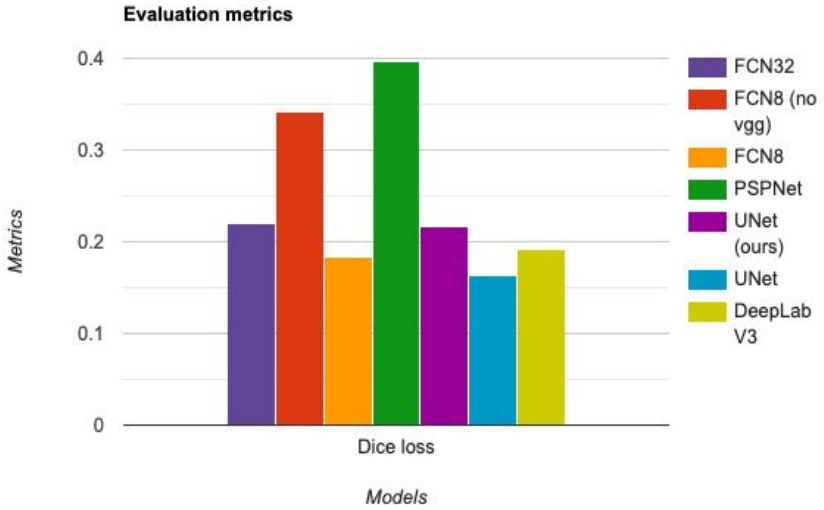
- **Binary Cross Entropy (BCE) loss**, is a commonly used loss function in binary classification tasks. It measures the dissimilarity between predicted probabilities and true binary labels.
  Limitation: Considers loss information in a micro sense, rather than considering it globally

- **Intersection over Union (IoU)** is a metric used to evaluate the performance of semantic segmentation or object detection algorithms. It measures the similarity between the predicted region and the ground truth region by calculating the ratio of their intersection to their union.

- **Dice Loss** is a loss function commonly used in semantic segmentation tasks to measure the dissimilarity between predicted segmentation masks and ground truth masks. It is based on the Dice coefficient, which is a similarity metric that quantifies the spatial overlap between two sets.
  Advantage: Considers loss information both locally and globally

Models \ Metrics	Dice_Loss	IoU	BCE_Loss
FCN32 (with vgg16)	0.220	0.810	0.278
FCN8	0.342	0.762	0.350
FCN8 (with vgg16)	0.184	0.829	0.231
PSPNet (with Resnet)	0.396	0.812	0.573
U-net (ours)	0.216	0.810	0.258
U-Net (with Resnet)	0.163	0.828	0.241
Deep Lab v3	0.192	0.825	0.254


Evaluation metrics
1
0.9
0.8
0.7
0.6
0.5
Metrics
IoU
Models
FCN32
FCN8 (no vgg)
FCN8
PSPNet
UNet (ours)
UNet
DeepLab V3


Evaluation metrics
0.4
0.3
0.2
0.1
0
Metrics
Dice loss
Models
FCN32
FCN8 (no vgg)
FCN8
PSPNet
UNet (ours)
UNet
DeepLab V3

Conclusion and Future Steps

- U-Net performs better (ideally) than FCN, PSPNet and DeepLab due to the presence of skip connections between the encoder-decoder model.
- Deep Lab is expected to perform better if we use the actual Deep Lab Model instead of deeplabv3_mobilenet_v3_large.
- Equipping models with a pre-trained backbone like ResNet or VGG gives better performance.
- Pre-trained backbones can make the model memory-heavy, and for a simpler use-case like binary semantic-segmentation, a distilled down version makes more sense.
- **We are currently optimizing our custom U-net (which is lighter) and trying to make it perform better by incorporating channel attention and leaky ReLU.**

THANK YOU!!
Any questions?