# Segmentation using **UNet** Architecture

https://arxiv.org/pdf/1505.04597.pdf

**Team 40 StatBots**

Shivank Saxena - 2022900025

Chandu Chegu - 2022201062

Abhishek Reddy - 2022201025

Ravada Sai Venkatesh - 2022201072

# Table of Contents

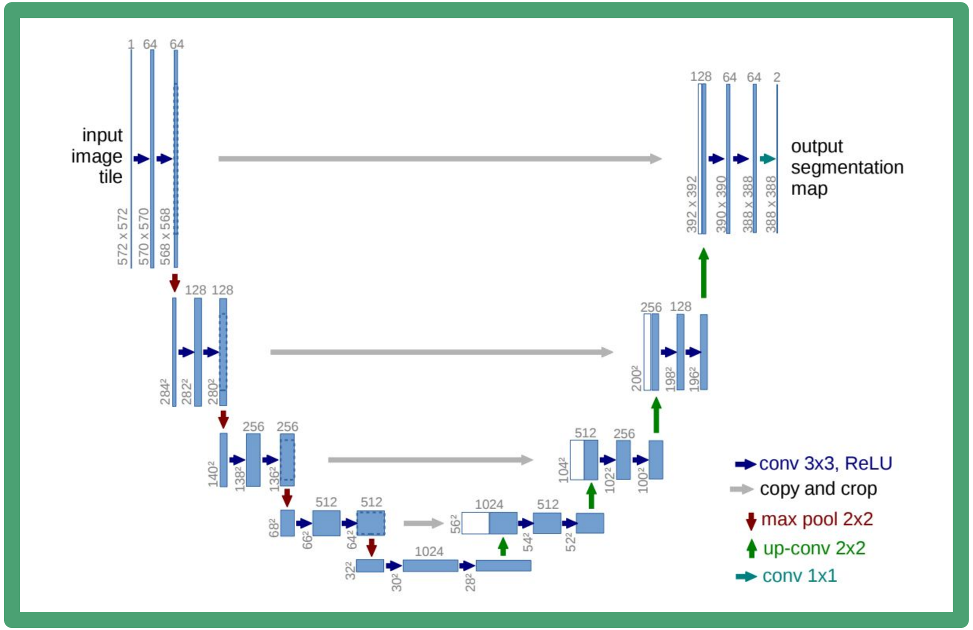
# Problem Statement

- The Goal is to take either a RGB color image (h*w*3) or a grayscale image (h*w*1) and output a segmentation map where each pixel contains a class label represented as an integer.

# UNET: Proposed Architecture

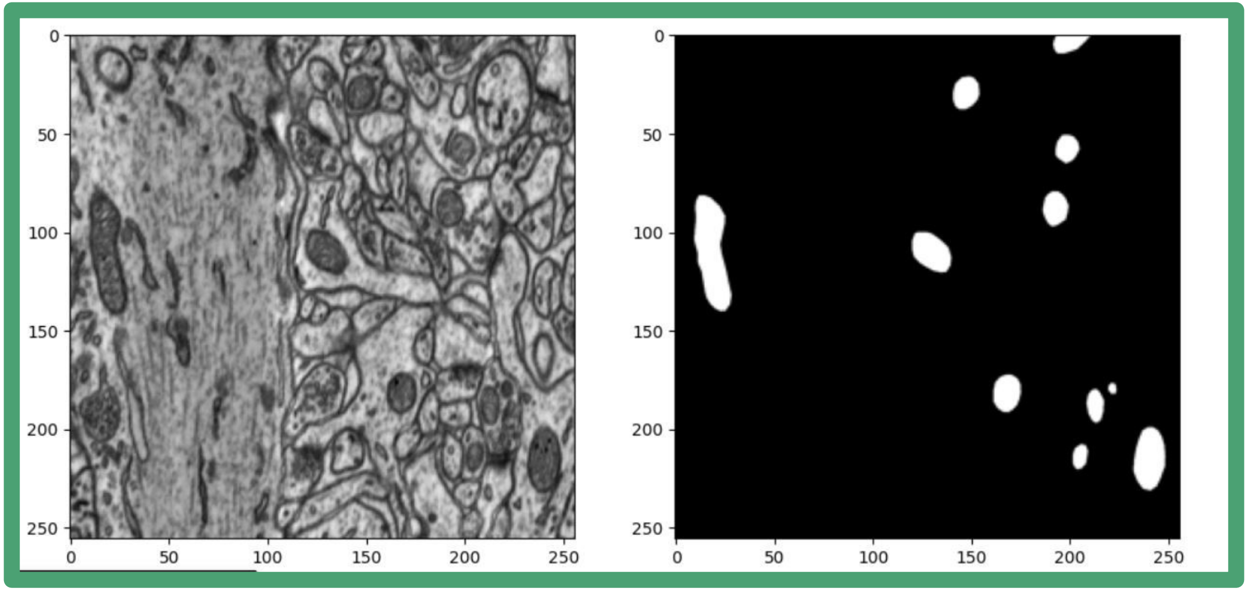
1. Auto-encoder enhanced with residual skip connections.

2. Has both contracting (feature extraction) and expanding paths(precise localization by concatenation), allowing the model to capture both local and global features of the input image.



# UNET: Dataset

- Electron Microscopy dataset



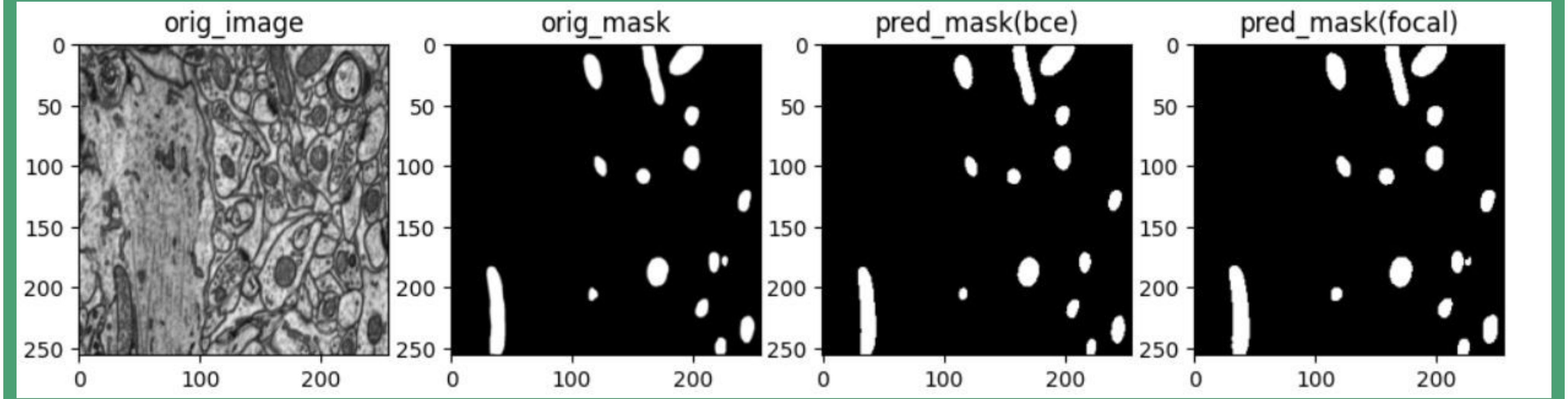
# UNET: Loss Function

- Experimented with Binary Cross-entropy, Focal Loss-Binary on the electron microscopy dataset.

| Cross - Entropy | Focal Loss |
|---|---|
| $C(P,y) = -\sum_i y_i \log P_i$ | $C(P,y) = -\sum_i y_i (1-p_i)^\gamma \log P_i$ |

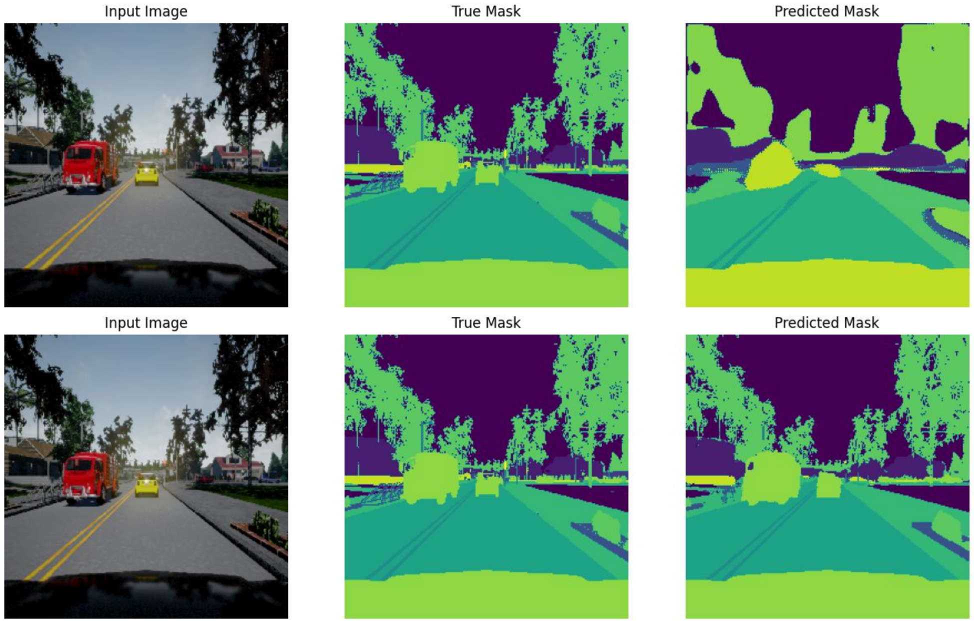
# UNET: Results

- The results with focal loss captured the smaller regions good. BCE(100 epochs) and Focal(50 epochs)



# UNET: Why it works so well

- We have experimented with the architecture and avoided the skip connections and compared the results



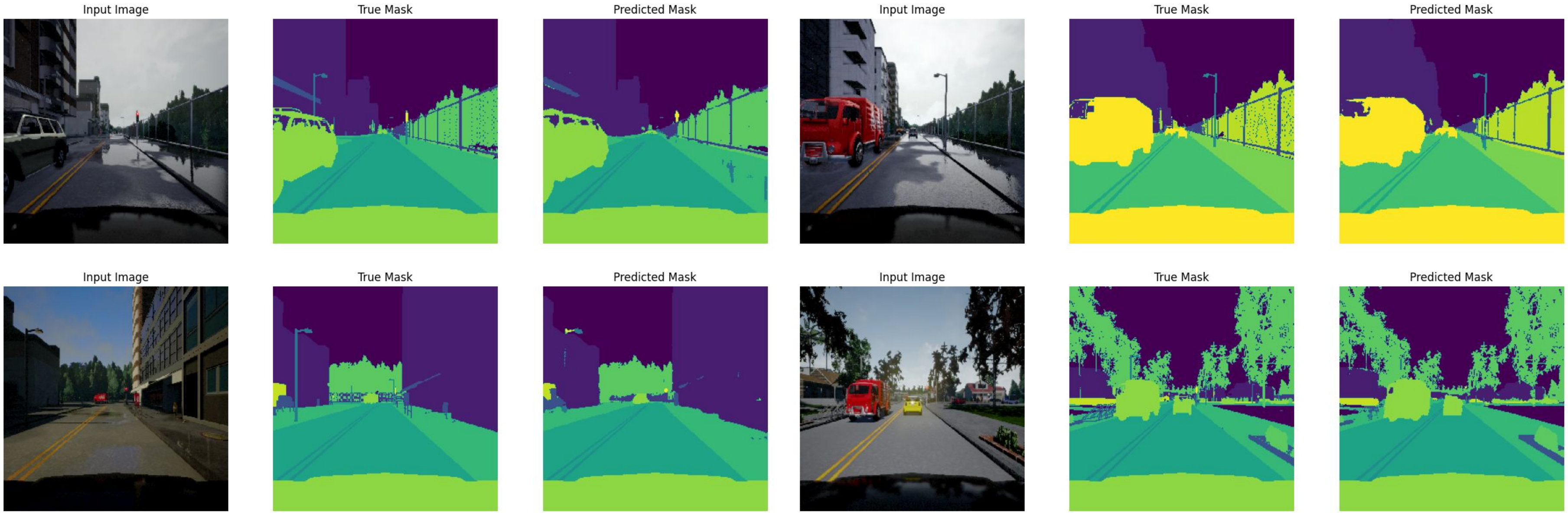

**Mean IOU 42.44%**

**(Autoencoder)**

**Mean IOU 72.55%**
**(UNET)**

# UNET: On Self Driving Car Dataset

- Mean IOU: 72.55 % (UNET)



# Variations: Segnet

It uses an encoder-decoder structure with pooling indices from encoder layers as skip connections to reconstruct the segmentation map

4

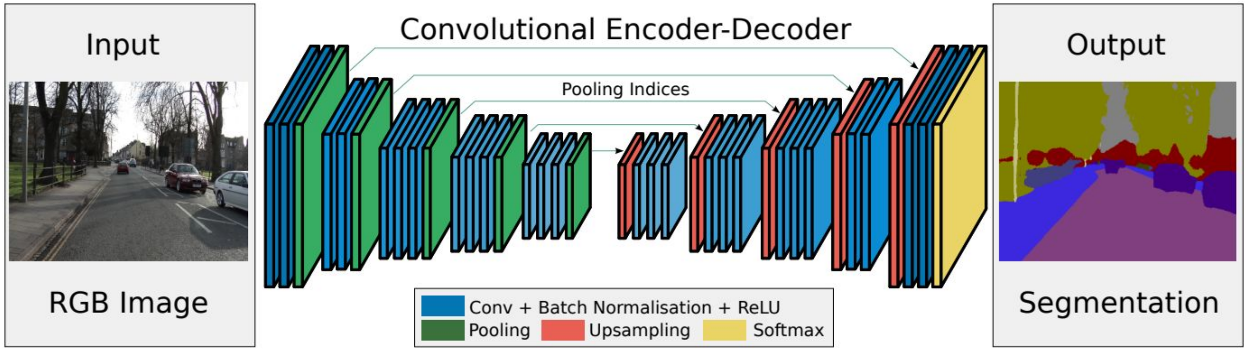
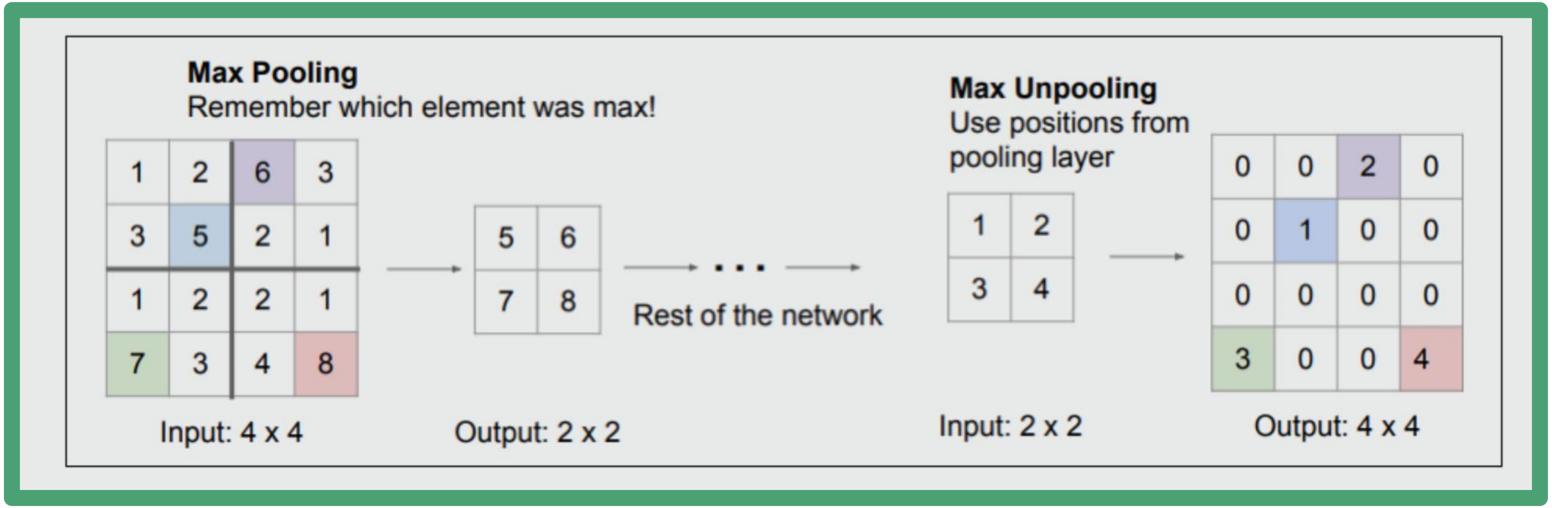


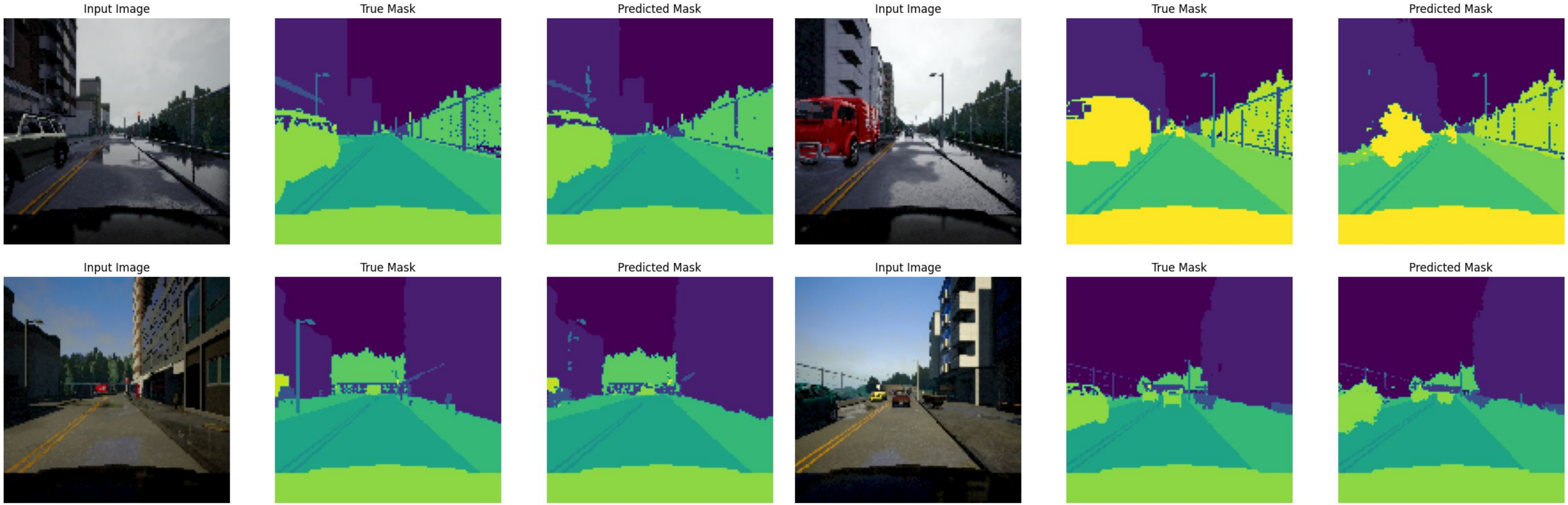

Fig. 2. An illustration of the SegNet architecture. There are no fully connected layers and hence it is only convolutional. A decoder upsamples its input using the transferred pool indices from its encoder to produce a sparse feature map(s). It then performs convolution with a trainable filter bank to densify the feature map. The final decoder output feature maps are fed to a soft-max classifier for pixel-wise classification.

# Variations: Segnet

During the max pooling operation, the pooling indices indicate which pixel was selected as the maximum value in the pooling window. These indices are then used while performing unpooling operation at decoder layers.

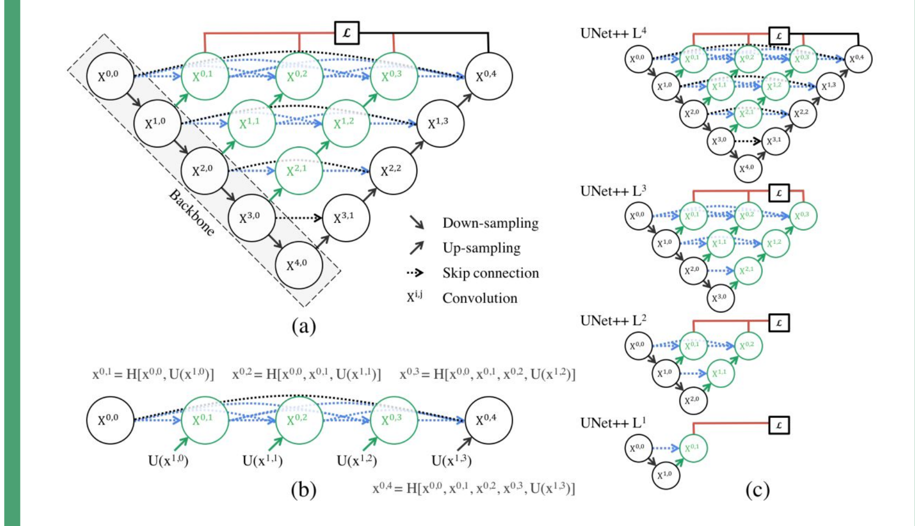


# Variations: Segnet
# mean IOU : 65.14%



# Variations: UNET++

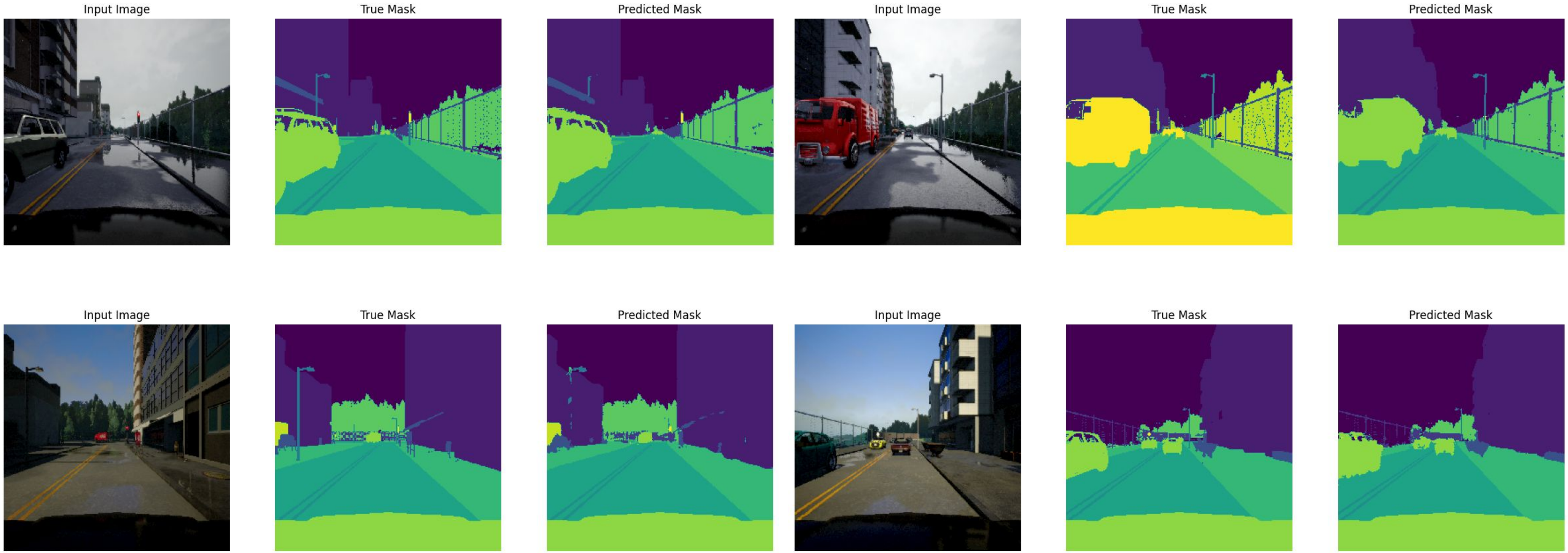
- It uses a series of nested, dense skip pathways that combines feature maps from different scales.
- This enables the model to capture both fine-grained and high-level features across multiple scales of the input image.
- The re-designed skip pathways aim at reducing the semantic gap between the feature maps of the encoder and decoder sub-networks.




$$\mathcal{L}(Y,\hat{Y}) = -\frac{1}{N}\sum_{b=1}^{N}\left(\frac{1}{2}\cdot Y_b\cdot\log\hat{Y}_b + \frac{2\cdot Y_b\cdot\hat{Y}_b}{Y_b+\hat{Y}_b}\right)$$

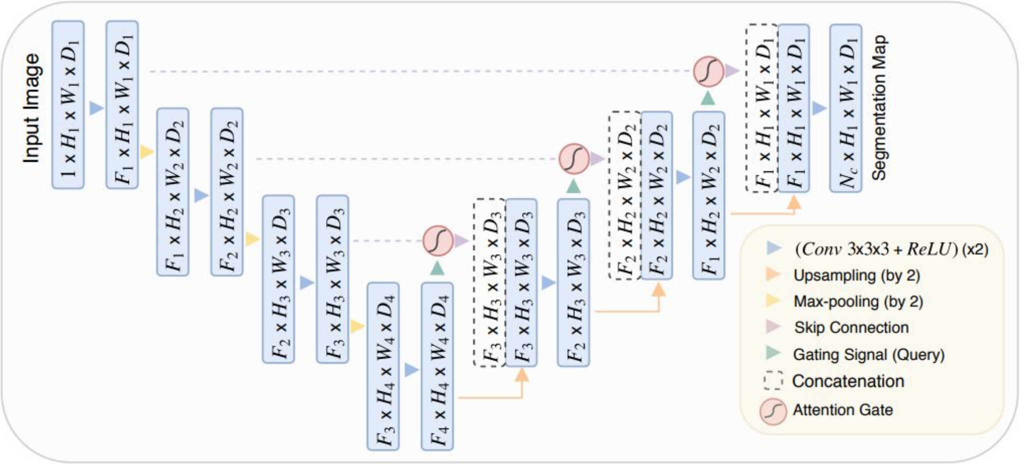
# Variations: UNET++

# Mean IOU: 73.97%



# Variations: UNET with Attention

- Skip connections in UNET contain many un-useful low-level features which are concatenated to the decoder. Attention Gate learns which of these features worth taking a look at and which are just noise.

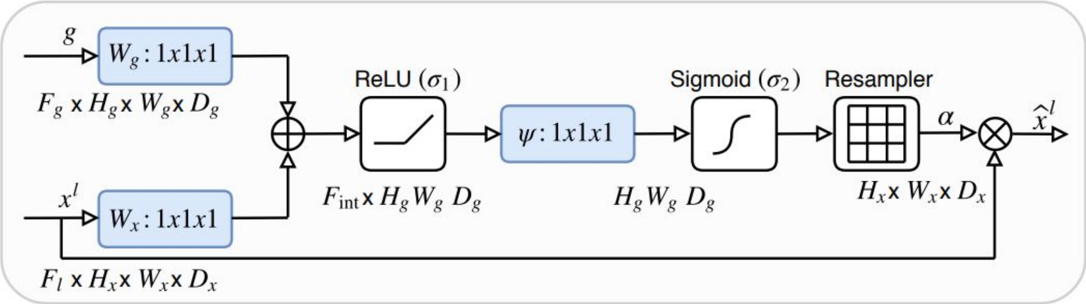


# Variations: UNET with Attention

Attention gate takes as a input a skip connection and the output from the previous decoder layer. Matematically attention gate does the following operation

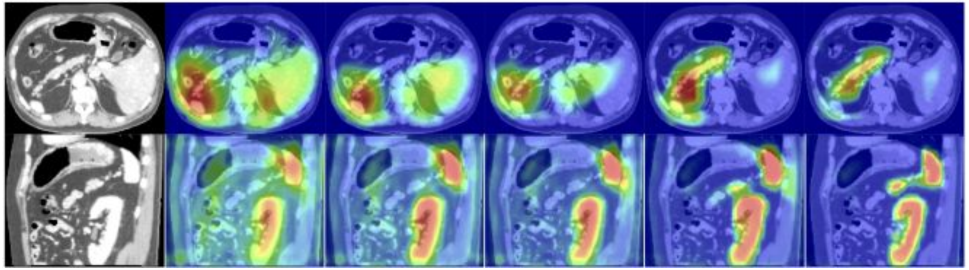
$$q_{att}^{l} = \upsilon^{T}(\sigma_1(W_x^T x_i^l + W_g^T g_i + b_g)) + b_\upsilon$$
$$\alpha_i^l = \sigma_2(q_{att}^l(x_i^l, g_i; \Theta_{att}))$$

Where $\sigma_2(x_{i,c}) = \frac{1}{1+\exp(-x_{i,c})}$, $\Theta_{att}$ contains linear transformations $W_x, W_g$, which are computed using 1×1×1 convolutions for the input tensors. Let's see how we can implement this.

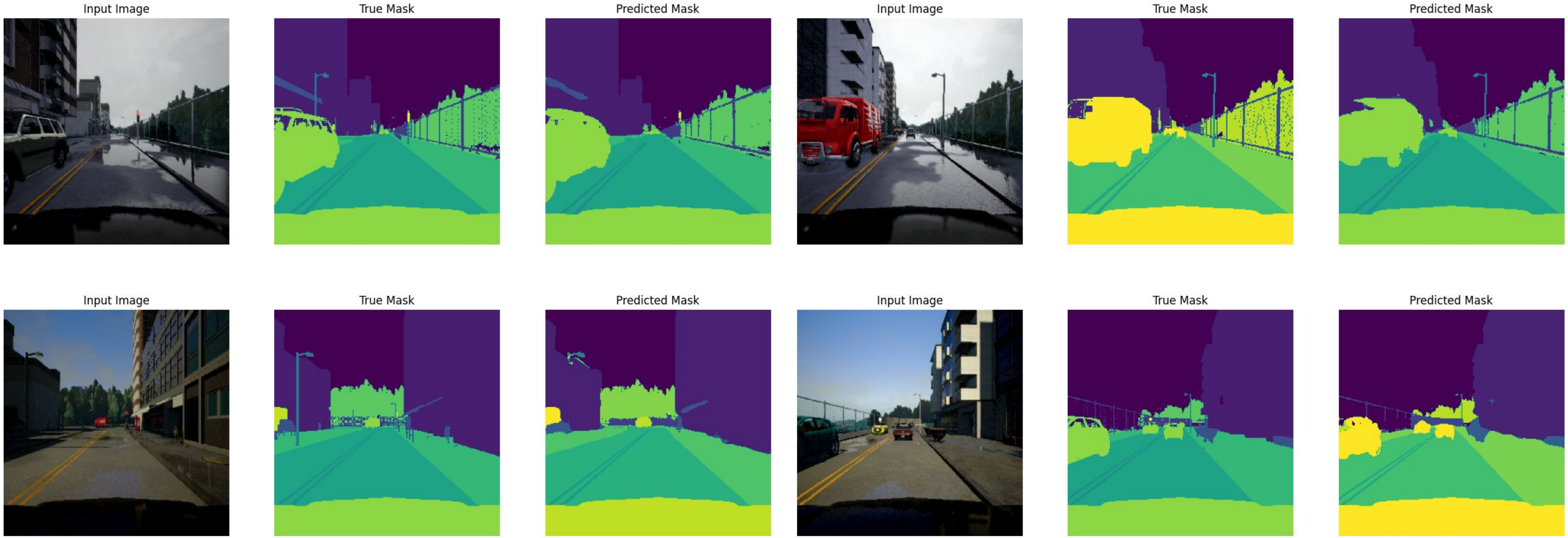


# Variations: UNET with Attention

- Input g (previous decoder layer output) and x (skip connection)
- Convolve x with 1×1 filter and stride = 2, and g with 1×1 filter and stride = 1
- Add together x and g
- Apply ReLU activation function
- $\psi = $ 1×1×1 convolution
- Apply sigmoid activation function
- Upsample sigmoid output to original input size (2×2)
- $att = multiply(upsample, x_{input})$
- 1×1 convolution with n_filters = n_input_x_filters and batch normalization

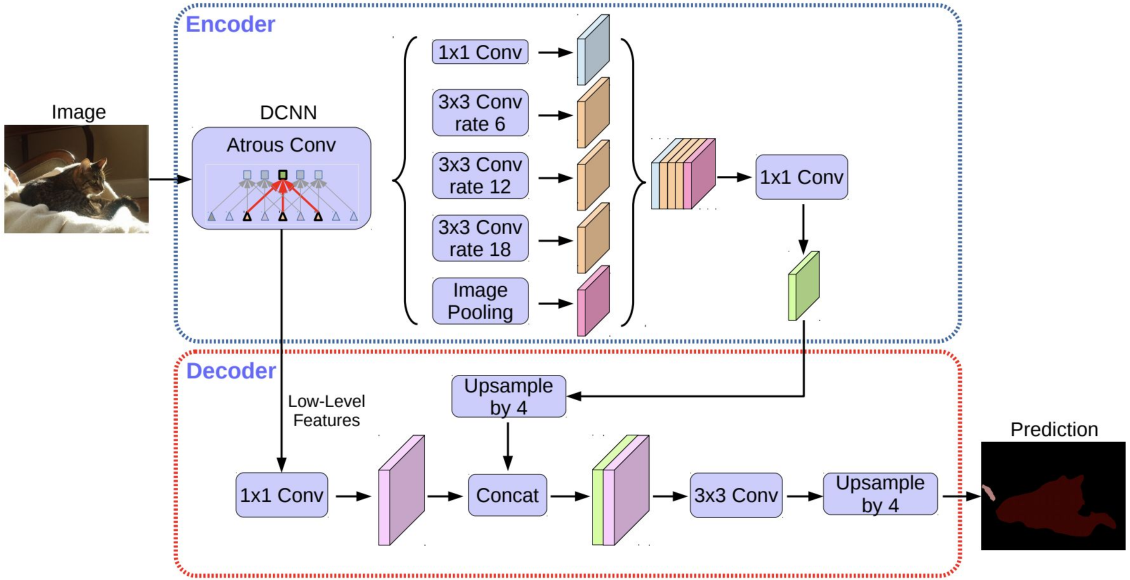


# Variations: UNET with Attention
# mean IOU: 69.73%



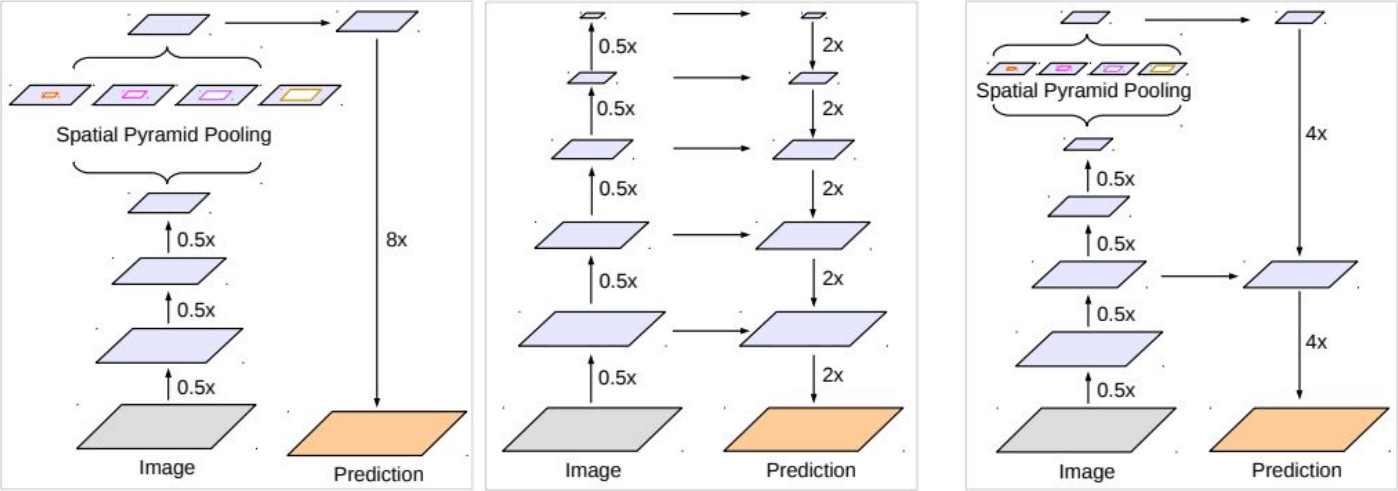
# Variations: Deeplab V3+

- Encoder Network is a pretrained model (ResNet) with atrous convolutions.



# Variations: Deeplab V3+

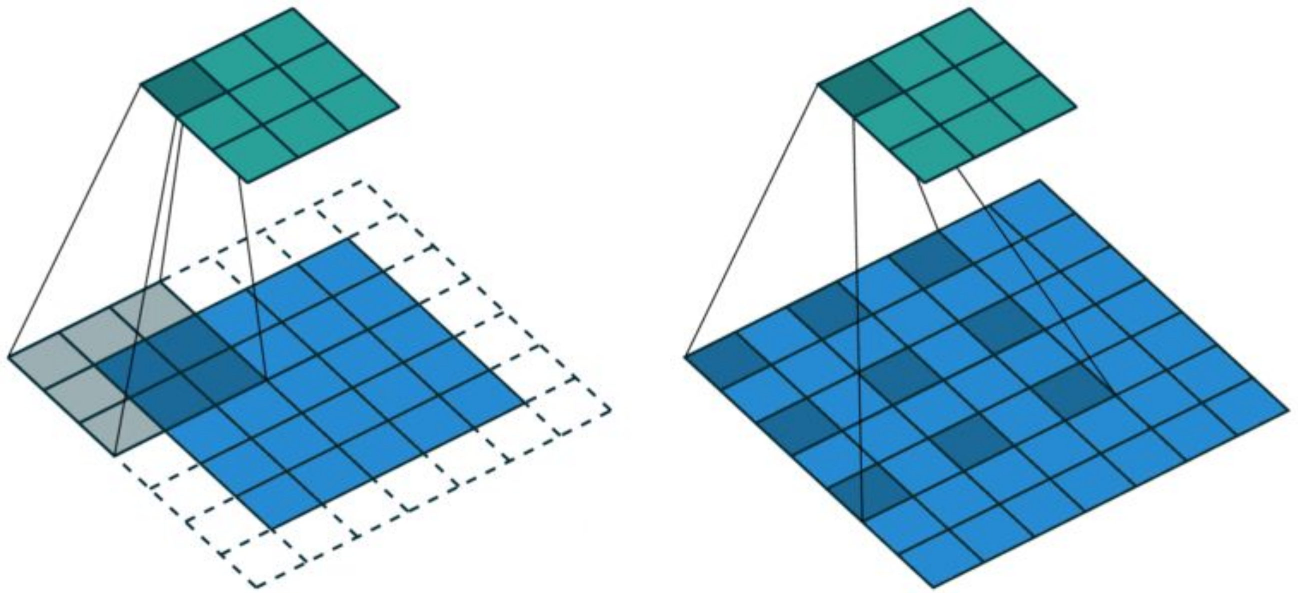
- We use the encoder-decoder architecture with atrous spatial pyramid pooling resulting in faster and stronger encoder-decoder network.



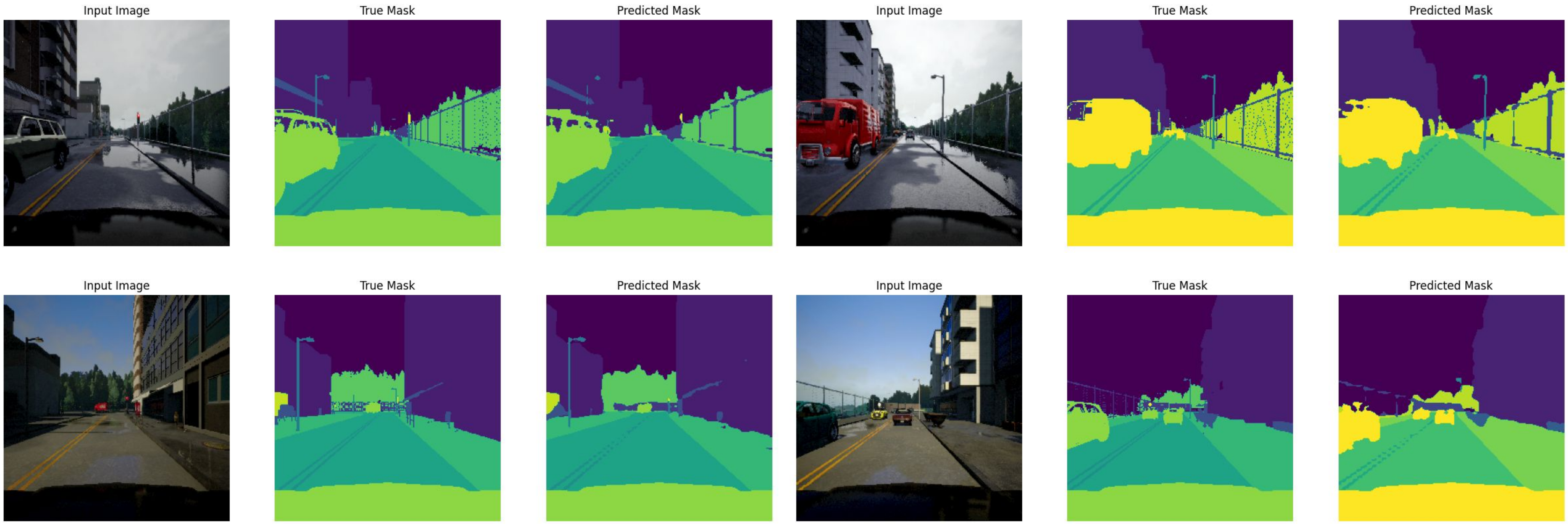

(a) Spatial Pyramid Pooling (b) Encoder-Decoder (c) Encoder-Decoder with Atrous Conv

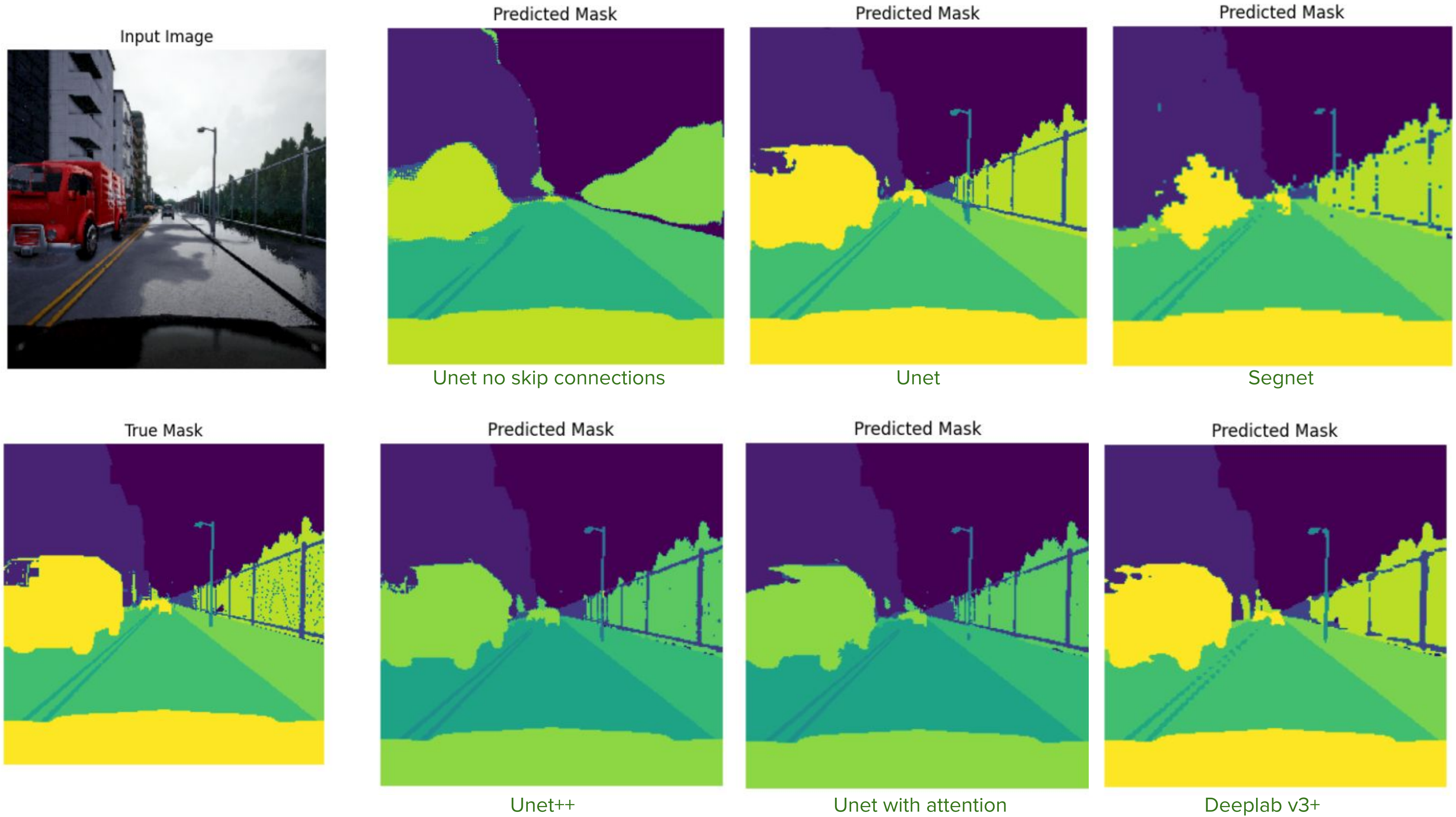
# Variations: Deeplab V3+

- Atrous Convolution Visualization



# Variations: Deeplab V3+
# Mean IOU: 72.86%




Input Image
Predicted Mask
Unet no skip connections
Predicted Mask
Unet
Predicted Mask
Segnet
True Mask
Predicted Mask
Unet++
Predicted Mask
Unet with attention
Predicted Mask
Deeplab v3+

| Model | Dataset | Input Dimension | Epochs | Mean IOU |
|---|---|---|---|---|
| UNET | Electron Microscopy | 256*256*1 | 50<br>100 | 82%<br>84% |
| UNET | Self Driving Car | 256*256*3 | 50 | 72.55% |
| UNET without skip connections | Self Driving Car | 256*256*3 | 50 | 42.44% |
| Segnet | Self Driving Car | 128*128*3 | 50 | 65.14% |
| UNET++ | Self Driving Car | 256*256*3 | 50 | 73.97% |
| UNET++ | Electron Microscopy | 256*256*1 | 50 | 84% |
| UNET with attention | Self Driving Car | 256*256*3 | 50 | 69.73% |
| Deeplab v3+ | Self Driving Car | 256*256*3 | 50 | 72.86% |

# Contributions

- We worked together to understand the paper
- **Shivank Saxena** - UNET with Attention and Deeplab v3+ on Self Driving Car dataset
- **Chegu Sai Poorna Chandu** - UNET, UNET++ on Mitochondria dataset
- **Abhishek Reddy Gaddam** - UNET, SEGNET on Self-Driving Car dataset
- **Ravada Sai Venkatesh** - UNET without skip connections, UNET++ on Self Driving Car dataset