Report

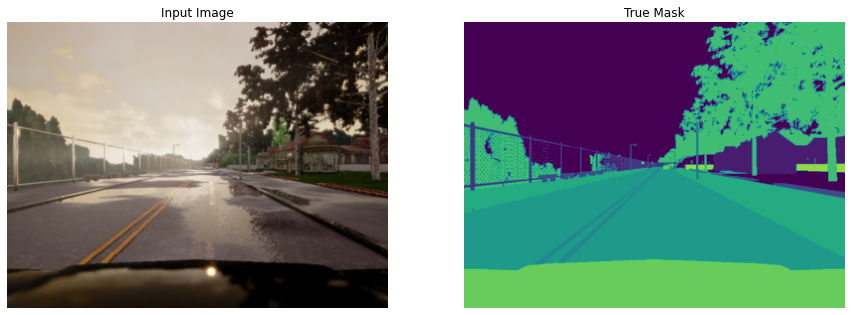
Data Processing

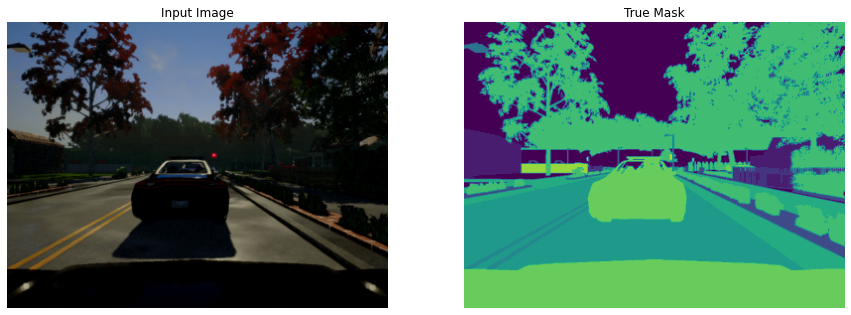
Each image is resized down to 288, 384 (original size: 600, 800) to reserve memory and maintain the aspect ratio.

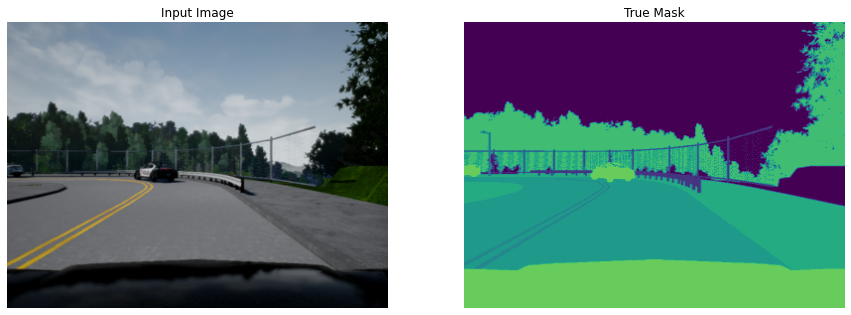
Data Augmentation

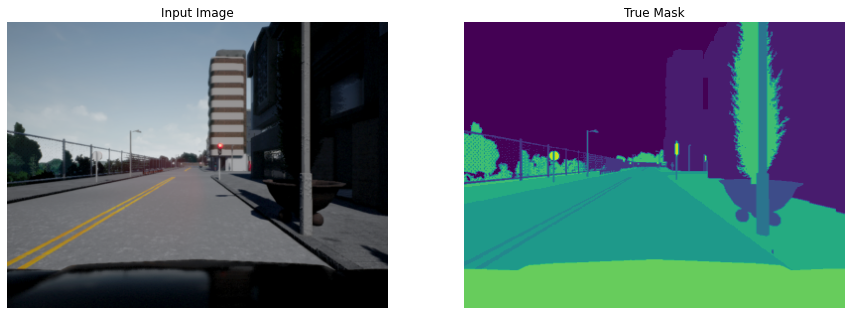
Each train image and it’s correspounding mask is flipped horizontally to increase the data size.

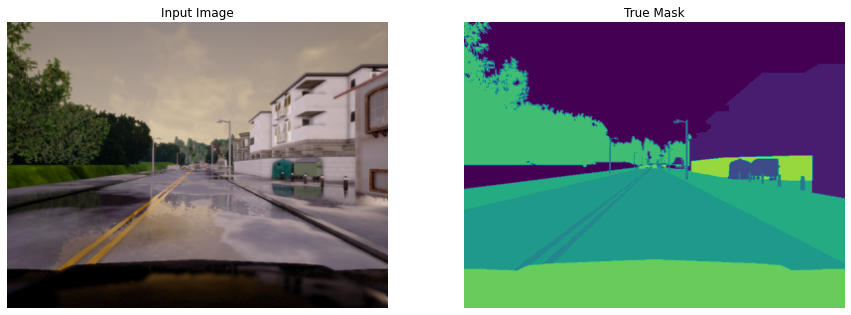
Data and Mask Visualization











Model Architecture (UNet)

----------------------------------------------------------------

Layer (type) Output Shape Param #

================================================================

Conv2d-1 [-1, 32, 288, 384] 896

Conv2d-2 [-1, 32, 288, 384] 9,248

MaxPool2d-3 [-1, 32, 144, 192] 0

Conv2d-4 [-1, 64, 144, 192] 18,496

Conv2d-5 [-1, 64, 144, 192] 36,928

MaxPool2d-6 [-1, 64, 72, 96] 0

Conv2d-7 [-1, 128, 72, 96] 73,856

Conv2d-8 [-1, 128, 72, 96] 147,584

MaxPool2d-9 [-1, 128, 36, 48] 0

Conv2d-10 [-1, 256, 36, 48] 295,168

Conv2d-11 [-1, 256, 36, 48] 590,080

MaxPool2d-12 [-1, 256, 18, 24] 0

Conv2d-13 [-1, 512, 18, 24] 1,180,160

Conv2d-14 [-1, 512, 18, 24] 2,359,808

ConvTranspose2d-15 [-1, 256, 36, 48] 524,544

Conv2d-16 [-1, 256, 36, 48] 1,179,904

Conv2d-17 [-1, 256, 36, 48] 590,080

ConvTranspose2d-18 [-1, 128, 72, 96] 131,200

Conv2d-19 [-1, 128, 72, 96] 295,040

Conv2d-20 [-1, 128, 72, 96] 147,584

ConvTranspose2d-21 [-1, 64, 144, 192] 32,832

Conv2d-22 [-1, 64, 144, 192] 73,792

Conv2d-23 [-1, 64, 144, 192] 36,928

ConvTranspose2d-24 [-1, 32, 288, 384] 8,224

Conv2d-25 [-1, 32, 288, 384] 18,464

Conv2d-26 [-1, 32, 288, 384] 9,248

Conv2d-27 [-1, 13, 288, 384] 429

================================================================

Total params: 7,760,493

Trainable params: 7,760,493

Non-trainable params: 0

----------------------------------------------------------------

Input size (MB): 1.27

Forward/backward pass size (MB): 280.12

Params size (MB): 29.60

Estimated Total Size (MB): 310.99

----------------------------------------------------------------

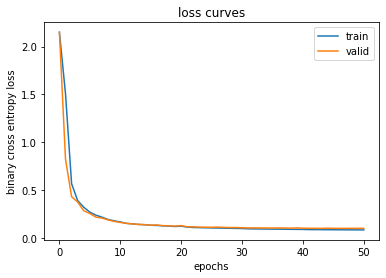
Loss Function:

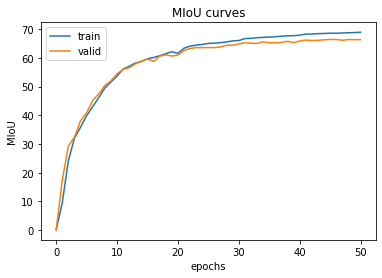
This project uses Focal Loss to mitigate data imbalance (under-represented class such as “Pole”, “Fence”, etc).

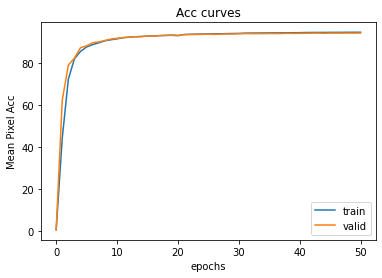
Evaluation:

1. Mean Pixel Accuracy (Acc)
2. Mean Intersect over Union (MIoU)

Training curves



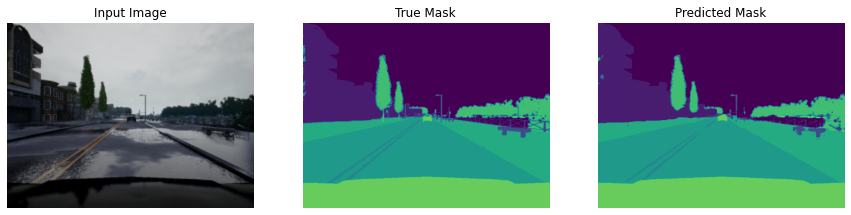




Test dataset Performance

Test Loss: 0.092 | MIoU: 68.08%| Acc: 94.43%



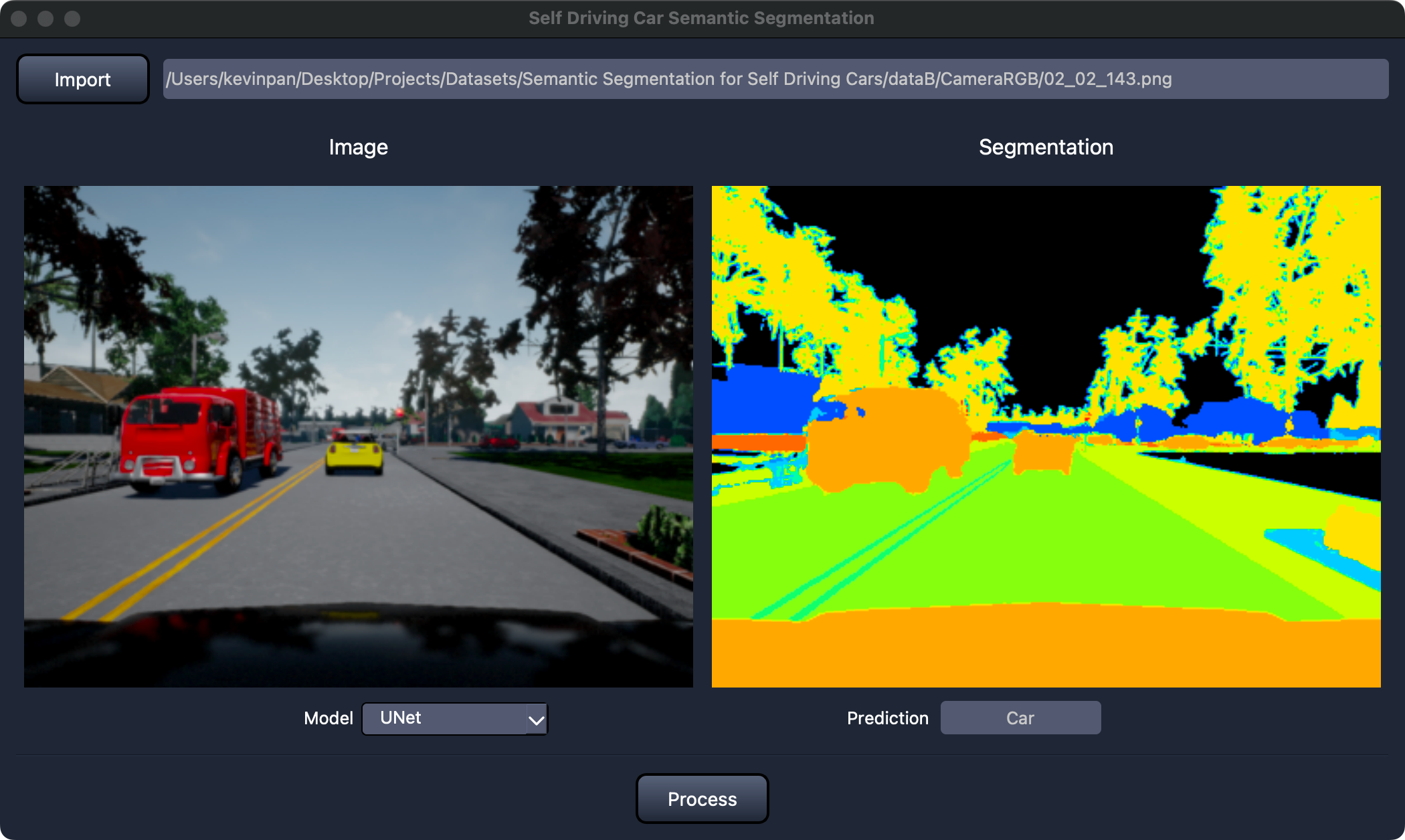








GUI



User Manual:

1. Press “Import” button to load an image.
2. The Dropdown Model menu lets you pick which model to use.
3. Then press “Process: button to generate a segmentation map.
4. Hover the mouse over the Segmentation map to see the Prediction of that pixel.