CS x476 Project 2

Zhaodong Yang
halyang@gatech.edu
zyang645
903748903
6476

Part 1.1: Image Segmentation

What is image segmentation? Why do we want to do image segmentation?

Image segmentation is dividing an image into different sections based on the object or boundaries in the image. Image segmentation classifies every pixel’s to a particular class in the image. The reason why we want to do image segmentation is to recognize object or group together similar-looking pixels for further processing.

What are some applications that use image segmentation? List at least 2.

Object recognition uses image segmentation to extract the objects of interest from an image. Medical imaging can use image segmentation to locate tumors and measure tissue volumes. And object detection can use image segmentation to detect the object we want, like pedestrian detection for self-driving cars.

Part 1.2: Sigmoid v. Softmax

What is the difference between sigmoid and softmax in terms of how they are used? What is the similarity in terms of their output values?

Sigmoid is used when detecting only one type of class, while softmax is used when detecting multiple types of classes. And when using sigmoid, we just output one probability of the pixel belonging to the only class. While using softmax, we output several probabilities of the pixel belonging to each of the possible classes.

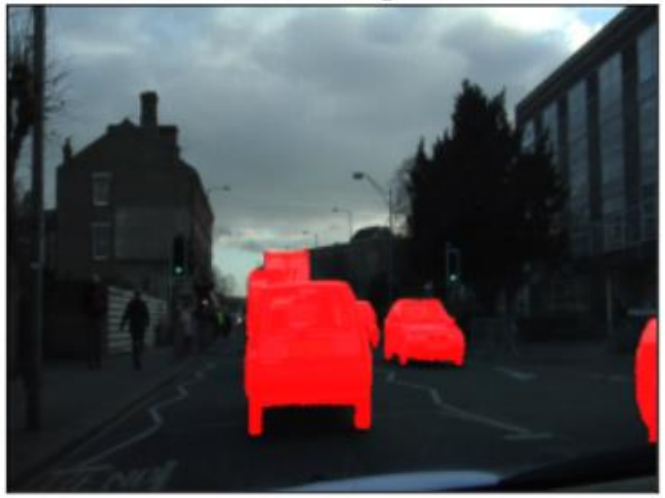
The effect of the outputs of sigmoid and softmax is the same. They both normalize the value of segmentation model output to [0, 1]. So the maximum and minimum output values of sigmoid and softmax are the same.

Part 1.3: Apply Mask to Image

Image

Cars Mask

Final Seg



Part 2.1a: Pre-trained Models

!! Please see the link in the title to help you answer the following questions.

What are some other **available encoders** that are not used in the project 2? List 4.

For ResNet, there are resnet18, resnet34, resnet101, resnet152 which are not used.

For VGG, there are vgg11, vgg13, vgg16, etc.

And there’s other encoder families like ResNeXt, ResNeSt, GERNet, SE-Net, etc.

Part 2.1b: Pre-trained Models

!! Please see the link in the title to help you answer the following questions.

What is the architecture of one of the **segmentation models** that you are interested in that's not covered in the project 2? Provide some details of this architecture from its associated paper.

I am interested in the U-Net, which is built upon FCN with some modifications to make it work with few training images and yield more precise segmentations. One main modification in U-Net is that in the upsampling part the decoder also has a large number of feature channels, which allow the network to propagate context information to higher resolution layers. The network does not have any fully connected layers and only uses the valid part of each convolution.

Part 2.1c: FCN Paper

!! Please see the link in the title to help you answer the following questions.

What is the result and reason of viewing fully connected layers as convolutions with kernels? (Hint: Look into Paper Section 3.1)

The result is the network can take input of any size and output classification maps. While the resulting maps are equivalent to the evaluation of the original net on particular input patches, the computation is highly amortized over the overlapping regions of those patches. And because it outputs spatial maps, it is more appropriate for using on dense problem like semantic segmentation.

What are the number of convolutional layers and parameters of the 3 models used for segmentations? (Hint: Look into Paper Table 1 and Section 4)

FCN-AlexNet has 8 convolutional layers and 57M parameters.

FCN-VGG16 has 16 convolutional layers and 134M parameters.

FCN-GoogLeNet has 22 convolutional layers and 6M parameters.

Part 2.2: VGG

!! Please see the link in the title to help you answer the following questions.

What is the total number of convolutional layers (Conv) in VGG-19? What is the total number of fully connected layers in VGG-19? (Hint: Look into Paper Figure 3)

VGG-19 has 16 convolutional layers and 3 fully connected layers.

What do you notice about the image height and image width as you go through the _encoder_ of the FPN+VGG-19? What about the _decoder_ of the FPN+VGG-19? (This is the question 1 is the Notebook)

As we go through the encoder, the image height and width become half of the input every time it goes through a MaxPool layer. As for the decoder, the image height and width grow twice as before every time it goes through a FPNBlock.

Part 2.3: Resnet

!! Please see the link in the title to help you answer the following questions.

What is the total number of convolution layer (Conv) in ResNet-50? What is the total number of fully connected layers in ResNet 50? (Hint: Look into the Figure linked in notebook)

ResNet-50 has 49 convolution layer and 1 fully connected layers.

What do you notice about the size of the FPN+ResNet-50 network/model in comparison with the FPN+VGG-19 network/model? What are other major differences that you notice between the two model architectures? (List at least 2.) (This is the question 2 is the Notebook)

The size of the FPN+ResNet-50 is much bigger than FPN+VGG-19. And FPN+ResNet-50 has more convolutional layers. And encoder of FPN+ResNet-50 has been divided into different stages and every three of its convolutional layers are combined into a bottleneck.

And I think FPN+ResNet-50 might perform better because it is deeper in depth and has more layers, which enable it to understand features better.

Part 2.4: Feature Map

What feature in the input image does the FCN-ResNet50 model appear to focus on:

- In the first layer of its encoder,
- In the last layer of its encoder
- In the last layer of its decoder?

The model focus on the big contours in the first layer of its encoder.
The model focus on the texture or the details of the image in the last layer of its encoder.
The model focus on the whole object, which is the car, in the last layer of its decoder.

What does this tell you about the learning process of the model?

The model learns from the coutours of an image. Then it looks deeper and learns the details of the image. Then it shapes its own concept about the image, which segments the image into different object.

Part 3.1: IoU

IoU encodes the shape properties of the object into the region property with normalized measure focusing on the area.What is the benefit of such property of IoU? (Hint: Check out the section 1 of paper linked in the title)

This property makes IoU invariant to the scale of the problem under consideration.

Which prediction result would have higher IoU score? Please Explain the reason. (This is the question 3 is the Notebook)

Pred Mask 1 would have higher IoU score, because Pred Mask 2 have an big extra mask area which both Pred Mask 1 and ground truth don’t have. So the denominator of IoU of Pred Mask 2 would be higher, which causes its IoU scores to be lower.

Part 3.2: Apply IoU

What is the IoU score for VGG-19 and ResNet-50? (Output from your Jupyter Notebook)

vgg19 IoU score is: [tensor(0.9223), tensor(0.9312), tensor(0.9415)]

ResNet50 IoU score is: [tensor(0.8864), tensor(0.9095), tensor(0.9393)]

Which FCN backbone has better performance? Based on your understanding, why does one FCN backbone perform better than the other?

VGG-19 performs better. The models were trained for 80 epochs on the exact same dataset with the "save best weights" strategy. So the models iterated few times every epoch and didn’t iterate through all the epochs. As a result, because the model with fewer layers iterates faster, in other word, it takes fewer iterations to reach optimal weights, the VGG can reach optimal weights. But in limited iterations ResNet may not reach its optimal weights for every epoch.

Part 3.3: Performance

What is the relationship between the number of parameter and the performance?

According to the result of this experiment, models with fewer parameters performs better.

Extra Credit 1: PSPNet

!! Please see the link in the title to help you answer the following questions.

What are some shortcoming of FCN mentioned in the PSPNet Paper? (Hint: Look into Paper Section 1)

FCN has problem dealing with diverse scenes and unrestricted vocabulary. It is hard for FCN to recognize objects of similar appearance.

And the major issue for current FCN based models is lack of suitable strategy to utilize global scene category clues.

What is the main difference between FCN and PSPNet? (Hint: Look into Paper Section 1)

PSPNet incorporates suitable global features and FCN doesn’t.

In addition to traditional dilated FCN for pixel prediction, PSPNet extends the pixel-level feature to the specially designed global pyramid pooling one. The local and global clues together make the final prediction more reliable.

PSPNet uses a different optimization strategy with deeply supervised loss.

Extra Credit 2: PSPNet

!! Please see the link in the title to help you answer the following questions.

What is the reason for using PPM based on the PSPNet Paper? (Hint: Look into Paper Section 3.2)

Because PPM empirically proves to be an effective global contextual prior. And the empirical receptive field of CNN is much smaller than the theoretical one especially on high-level layers. This makes many networks not sufficiently incorporate the momentous global scenery prior. So we need PPM to address this issue.

What is your IoU score for PSPNet-ResNet50 and FPN-ResNet50?

PSPNet-ResNet50 IoU score is: [tensor(0.8106), tensor(0.9054), tensor(0.9005)]

FPN-ResNet50 IoU score is: [tensor(0.8864), tensor(0.9095), tensor(0.9393)]