CS x476 Project 2

Bipin Koirala
GT email: bkoirala3@gatech.edu
GT username: bkoirala3
GT ID: 9037.15.285
Section (6476)

Part 1.1: Image Segmentation

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

Answer:

Image segmentation is the process of division of an image into sub regions or categories within the image. It is done by classifying each pixel to belong to a particular class. This results in an image mask which contains pixel belonging to same class with the same value.

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

Answer:

Image segmentation is used in many areas such as;

- Locating tumors in patients
- Face detection
- Autonomous driving
- Detecting objects in satellite imagery etc.

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?

Answer:

When converting a classifier's raw output values into probabilities through sigmoid function, they fall in the range of [0,1]. Sigmoid function treats these output as independent events i.e., occurrence of any one class doesn't affect the other classes.

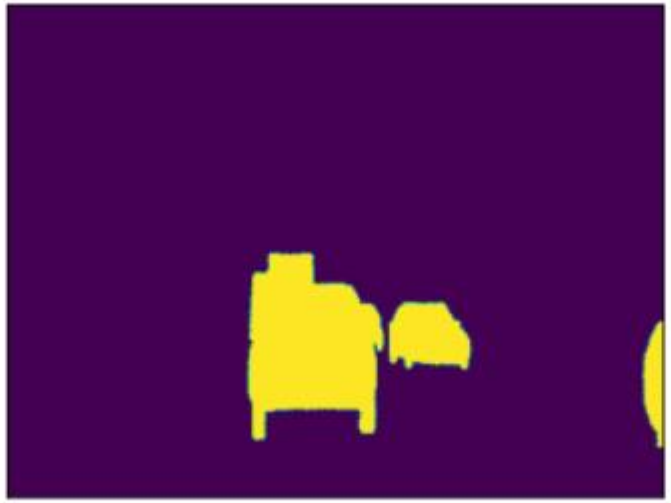
However, if the SoftMax function is used to convert the raw output of classifier into probabilities then the sum of each probability is equal to one. This means that sigmoid function treats these output as mutually exclusive events.

One key similarity between these two functions is that they generate values that fall between 0 and 1 so that any arbitrary raw output from the classifier is made comprehensive to the general understanding.

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.

Answer:

- DenseNet
- SE-Net
- Inception
- GERNet

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.

Answer: LinkNet Architecture

LinkNet is efficient in sharing information from the encoder with the decoder after each down-sampling block, which is proven to be better than using pooling indices in decoder or just using FCN in decoder. Furthermore, this feature forwarding technique facilitates for fewer parameters in decoder.
The first block contains convolution layer with filter size 7x7 and a stride of 2; this is followed by a max-pool layer of window size 2x2 and stride of 2. The last block does the full convolution by taking the feature map from 64 to 32, followed by 2D convolution. At last, this full convolution is used as the classifier with a kernel size of 2x2.

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)

Answer:

The authors have created ‘taby cat heatmap’ in RGB format (3 channels). Output from convolution network has many channels and therefore they need a convolution with 3 kernels of size 1x1 to make the output in RGB format. This converts a CNN to an FCN.

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)

Answer

	FCN-AlexNet	**FCN-VGG16**	**FCN-GoogLeNet**
Convolution Layers	8	16	22
Parameters	57 M	134 M	6 M

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)

Answer

- No. of Convolution layers = 16
- No. of Fully Connected layers = 3

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)

Answer:

Encoders in convolution networks progressively brings an original image size to a reduced size. On the other hand, decoder does the opposite by converting the encoded image tensor to a larger tensor (original size of the image). Max Pooling is associated with encoder in a sense that after a convolutional layer; a pool filter is applied so that it down samples the image into lower resolution. Decoders help in reconstructing the image by merging up-sampled map until finest resolution map is generated.

Part 2.3: Resnet

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)

Answer:

- No. of Convolution layers = 49
- No. of Fully Connected layers = 1

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

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)

Answer:

•The model that uses FPM+ResNet-50 has higher size in comparison to the model with FPN+VGG-19.
•Model with VGG-19 has 16 convolutional layers that are followed by 3 fully connected layers. Furthermore, the width of network starts with a value of 64 and progressively increases by a factor of 2 after each pooling layer.
•ResNet-50 contains 49 convolutional layers and 1 fully connected layer at the end of the network. It solves the problem of vanishing gradient unlike VGG-19. Therefore, the model FPN+ResNet-50 should theoretically perform better but in practice the performance depends on dataset, model parameters, data preprocessing choices etc.

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?

Answer:

1st Layer encoder : It maps the environment as a whole
Last layer encoder : vectorizes (outlines) the shapes
Last layer decoder : localizes the feature we want (car)

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

Answer:

It tells that the learning model is an iterative process where a filter is applied to the previous output (inside the layers) which helps in identifying features in an image such as edges, vertical lines, blends etc.

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)

Answer:

Normalization makes the IoU invariant to the scale of the problem under consideration. This normalized measure focuses on the area/volume for object similarity evaluation.

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

Answer

In simple terms, IOU is the ratio of size of intersection between the true mask and predicted mask to the union of true mask and predicted mask.

As per the definition of IOU, Pred Mask 1 would have higher IOU score. It is because the union of true mask and Pred Mask 2 has more size than compared to the union of True Mask and Pred Mask 1 since there is an extra predicted mask in case II.

This increased size of the union of two True Mask and Pred Mask 2 would lower the IOU score even if the intersection of (true mask + pred mask 1) and (true mask + pred mask 2) are the same.

Part 3.2: Apply IoU

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

Answer

```
In [20]: vgg_iou = student_code.applyIoU(vgg19_model, test_dataset)
         resnet_iou = student_code.applyIoU(resnet50_model, test_dataset)

         print('vgg19 IoU score is: ',vgg_iou)
         print('ResNet50 IoU score is: ', resnet_iou)

         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?

Answer:

In this case, FCN+VGG-19 model has better performance. The total number of trainable parameters for ResNet50 is greater than VGG-19 backbone, this might have caused overfitting of data in our model.

Part 3.3: Performance

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

Answer:

CNN architecture depends on the amount of dataset i.e., how large the dataset is. With the increase in the number of parameters; more feature could be extracted up to a certain extent. Beyond this limit, there is a possibility of overfitting the data. Overfitting will eventually give errors like false positive.

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)

Answer

- Lack of suitable strategy to utilize global scene category clues
- Loss of spatial information

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

Answer:

PSPNet is favored over FCN when it comes to analyzing the global context of the image in order to predict at a local level. FCN based classifiers do not perform well in capturing the context of the whole image.

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)

Answer:

According to the paper, size of receptive field can indicate how much we use context information. There are instances where the empirical receptive field of CNN is smaller than theoretically predicted size on high layers. This would make the networks not contextualize the global prior input. PPM removes the fixed sized constraint of the CNN.

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

Answer

```
In [21]: psp_iou, fpn_iou = student_code.compare_psp_fpn(test_dataset)
         print('PSPNet-ResNet50 IoU score is: ',psp_iou)
         print('FPN-ResNet50 IoU score is: ', fpn_iou)

         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)]
```