DLCV HW2 Report

Student: B05901074 陳泓均
Collaborators: 潘彥銘、林奕廷、詹書愷、賴繹文

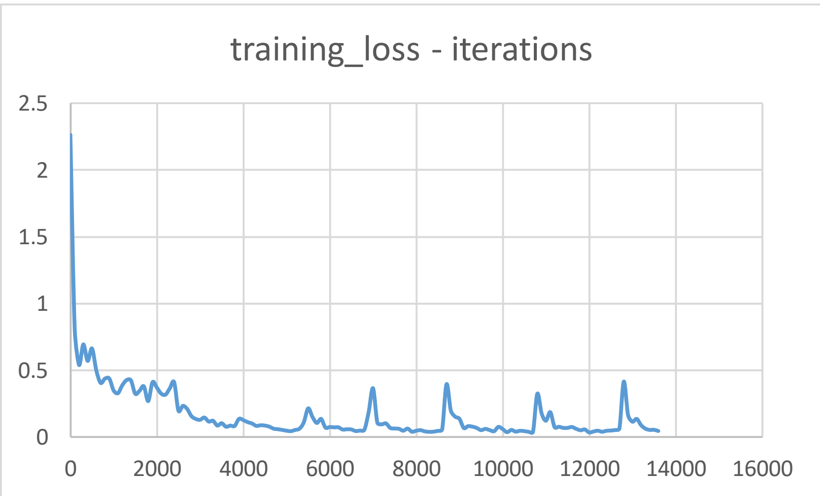
Problem 1.

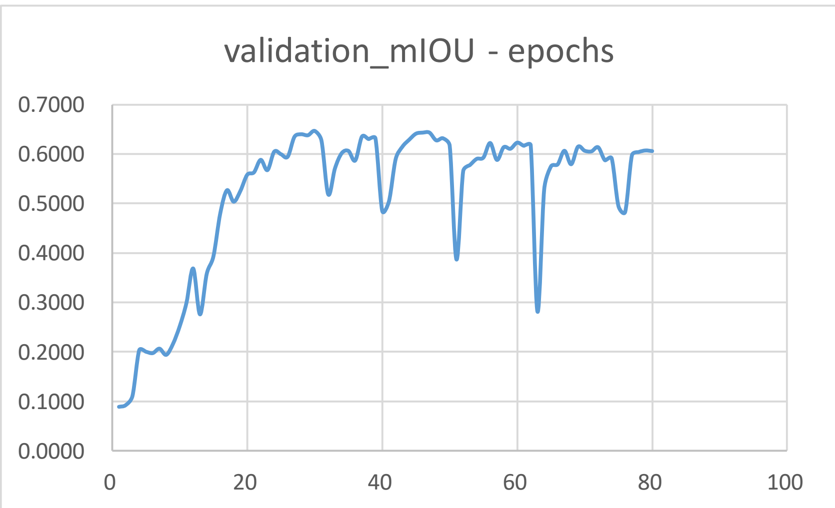
1. Baseline model

(1) Data pre-processing:

- Data Augmentation: For baseline model, I did not use any data augmentation techniques.
- Data Normalization: I normalize the data using transform.normalize to first convert Image from pixel value 0-255 to a tensor with value 0.0 – 1.0, then normalize with mean and std both equals to 0.5. (i.e. mean = [0.5,0.5,0.5], std = [0.5,0.5,0.5])

(2) Two figures:

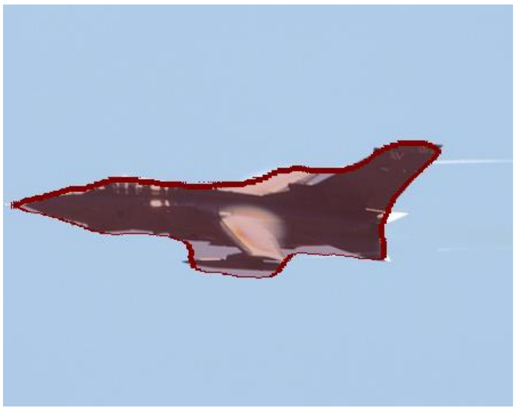




(3) Visualization

a. Person:

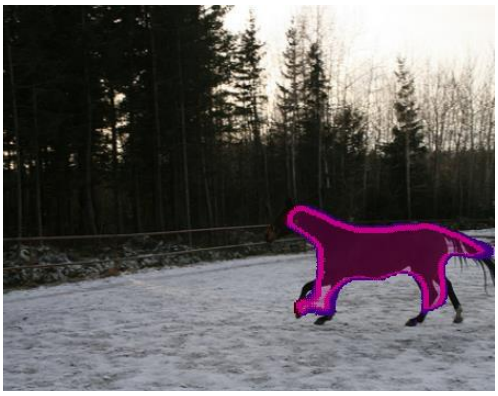
b. Aeroplane:



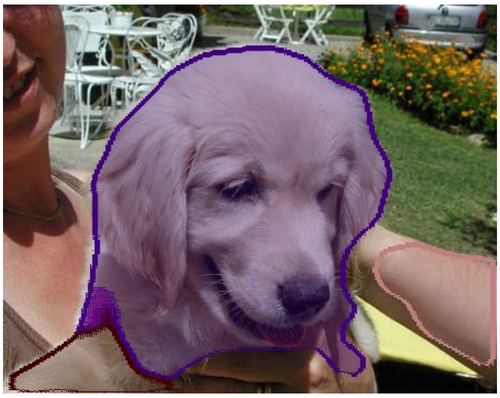
c. Bus

d. TV/monitor

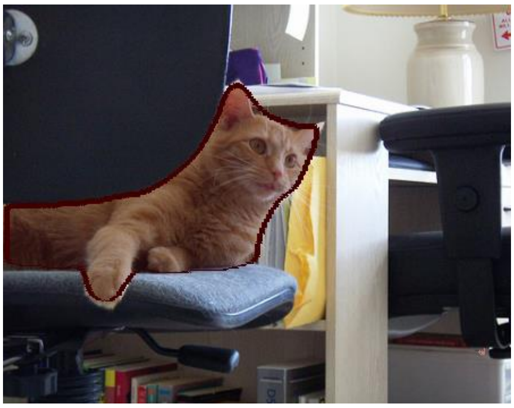
e. Horse



f. Dog



g. Cat



h. Car

(p.s. background is displayed in all images above)

(4) Miou score:

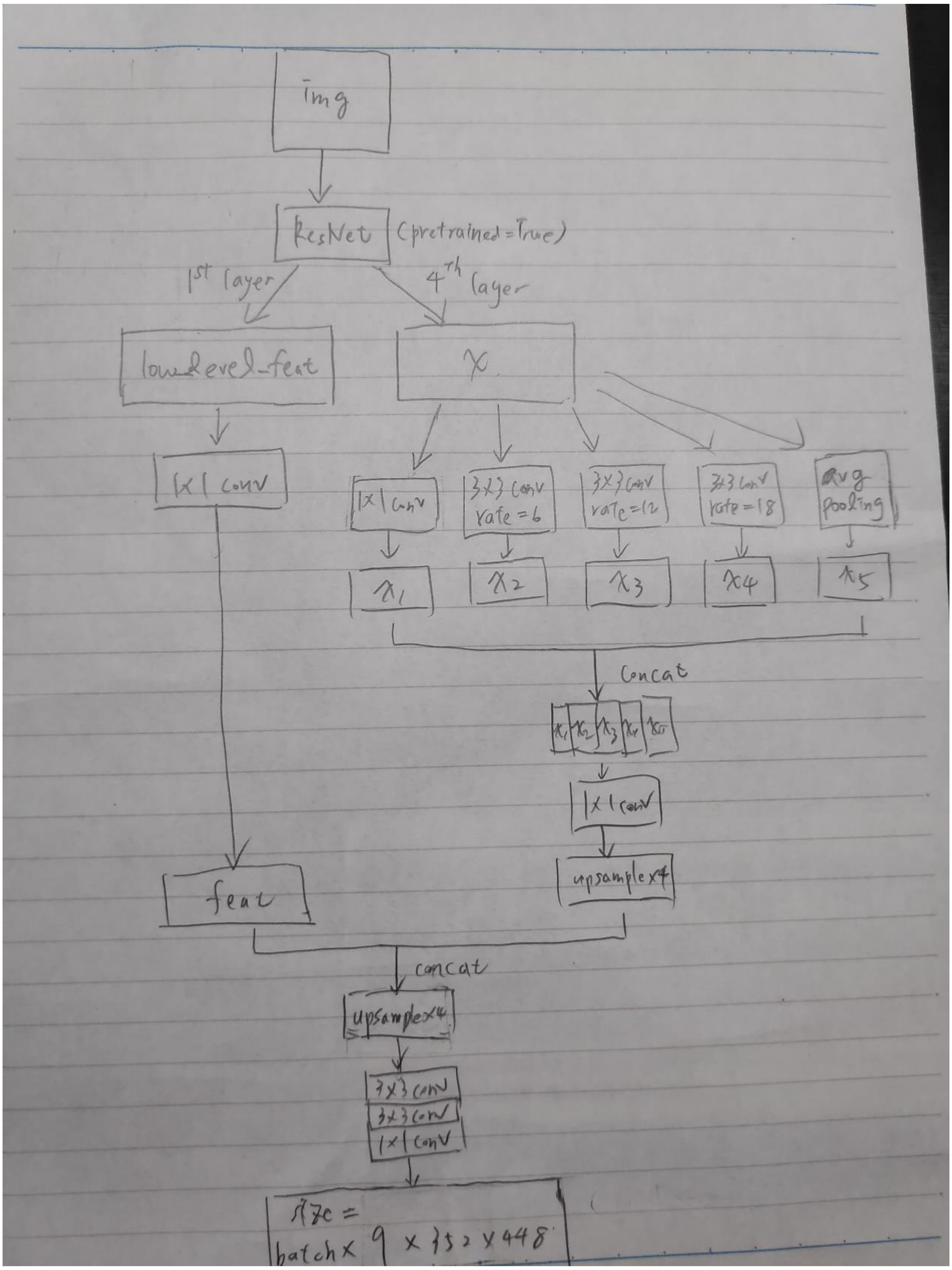
```
class #0 : 0.89513
class #1 : 0.73786
class #2 : 0.65648
class #3 : 0.71856
class #4 : 0.33786
class #5 : 0.51866
class #6 : 0.55926
class #7 : 0.70923
class #8 : 0.61094

mean_iou: 0.638220
```

Class 0 is the highest, and there is no surprise in there since background is easiest to identify. Also, class 4 is the lowest, and it is the class of TV/monitor. I think the reason is that sometimes TV or monitor contains some content inside that would make the machine misinterpret the information of the image. I demonstrate this below using data/val/img/0001.png as example:

As we can see, the content inside the monitor is identified as other classes, which would decrease the segmentation score.

2. Improved model

(1) Model architecture:

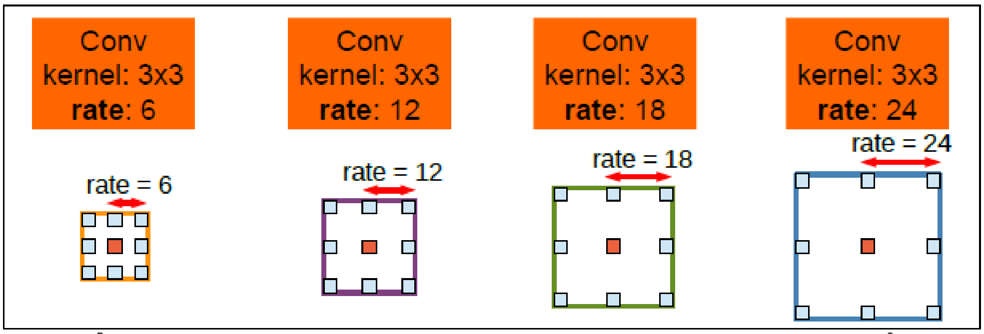


Here I am using the architecture of DeepLabv3+, which concats the low-level feature of the image and the feature after ASPP on 4^{th}-layer feature of resnet. I will illustrate the reasons why this model works on the next question.

(2) Reason:

First we need to know that this module use an important technique called Atrous Convolution, which is doing convolution with holes(an image is displayed below for better understanding), and the rate decides how wide we want to do our

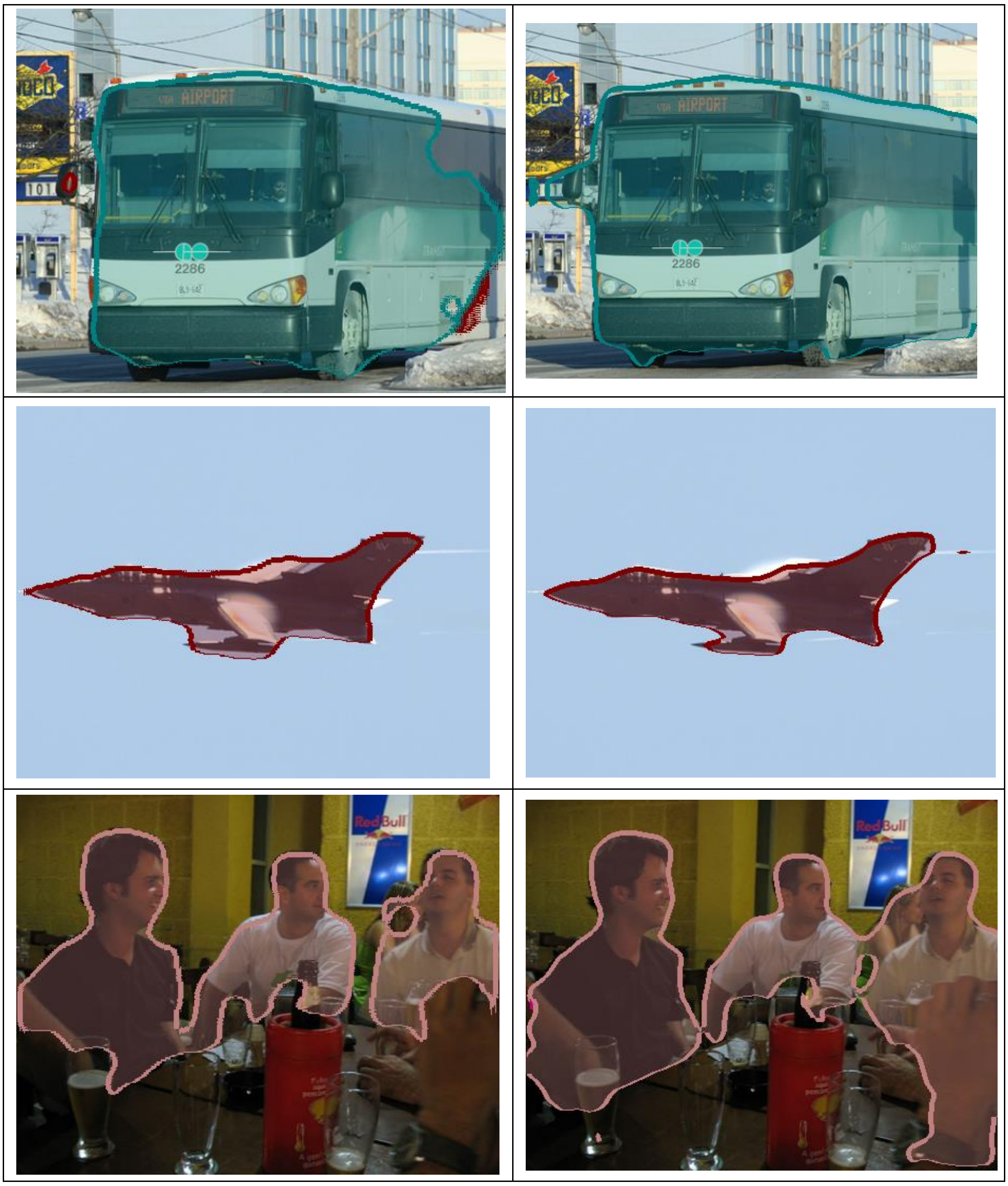
convolution. We want to do Atrous Convolution because this algorithm allows us to compute the responses of any layer at any desirable resolution. It makes the field-of-view larger and able to capture features of different resolution. If we use traditional convolution without atrous convolution, continue striding discards important information in the images. In image segmentation task, some image details such as edges is important, but convolutions could decimate the detail information needed. ASPP, Atrous Spatial Pyramid Pooling is then used in the module. This inspiration comes from the image pyramid, and using the pyramid concept here helps us not only better identify all detail information of all scale but also identify object at different scale.
Also, low-level feature is used here since we still need some low-level features to know the general semantics of the image. (Avg-pooling in the ASPP is of similar idea.)



(3) mIOU score on validation set:

```
class #0 : 0.92170
class #1 : 0.79225
class #2 : 0.76656
class #3 : 0.82841
class #4 : 0.48666
class #5 : 0.79918
class #6 : 0.76985
class #7 : 0.82035
class #8 : 0.79829

mean_iou: 0.775916
```

Baseline_model	Improved_model



As we can see here, improved model generally perform better with regard to deciding the edges and other details of the image.

Problem 2.

1. Reason:

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2}{2\sigma^2}} \cdot \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{y^2}{2\sigma^2}}$$

$$= G(x) \cdot G(y)$$

2. Gaussian filter implementation:

Original image	

After Gaussian filter	

Generally, the image becomes smoother and more blur. Therefore, it could be used in image denoising or smoothing.

Show that convolving with a 2D Gaussian

filter is equivalent to sequentially convolving with a 1D Gaussian filter in both vertical and

horizontal directions.

3. 1D convolution kernel implementation

$K_x = [0.5, 0, -0.5]$

$K_y = K_x^T$

Ix = 

Iy = 

4. Gradient magnitude Image

Original	

After gaussian	

The main difference is that the gradient magnitude image of lena.png has higher pixel value (i.e. brighter) at the edges, and I think that is because that original image has sharper edges, and therefore after convolving with edge-detecting filter, it gets higher pixel values. While in the case of Gaussian-filtered image, it is smoothened and thus will get lower pixel values at edges after convolving with a edge-detecting filter.