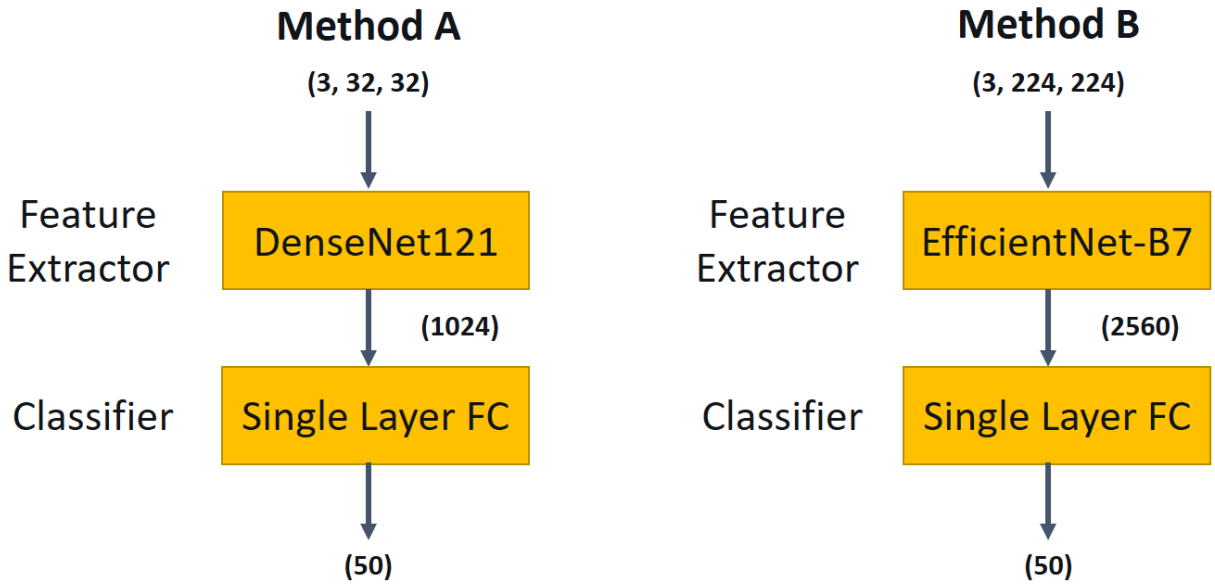
DLCV HW1 REPORT

r12942046 薛仁豪

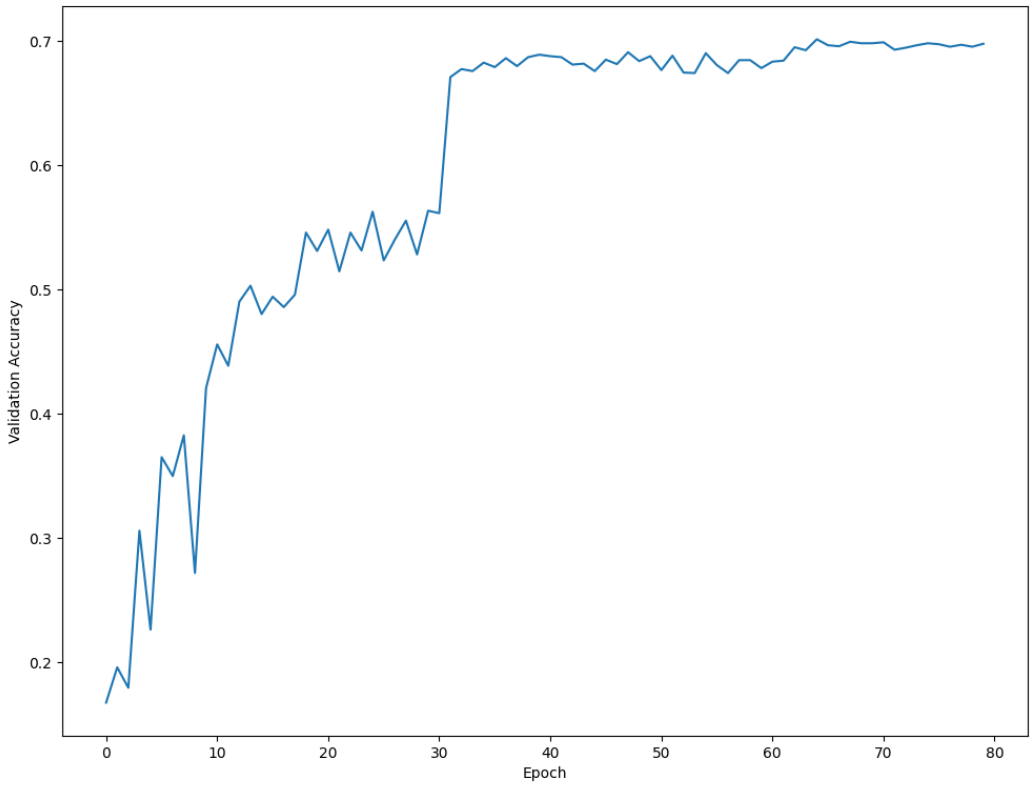
Problem 1

1. Draw the network architecture



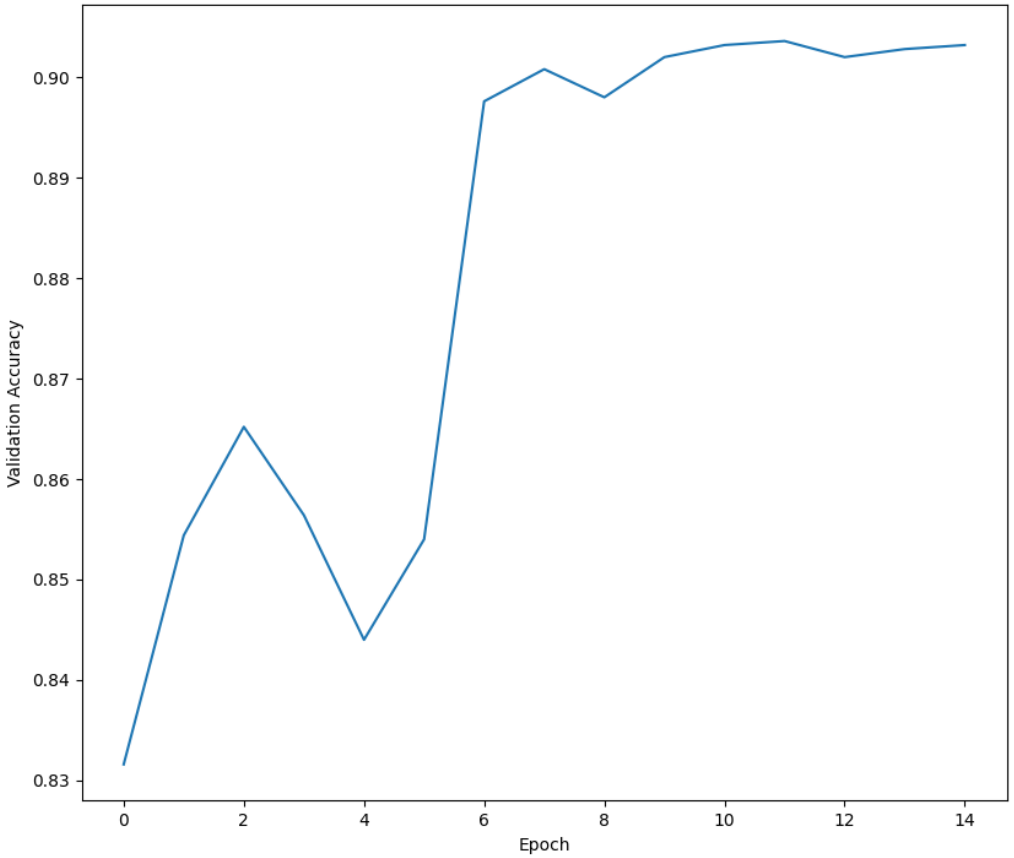
2. Accuracy of my models

Method A



Best Validation Accuracy: 70.16%

Method B



Best Validation Accuracy: 90.32%

3. Implementation details of model A

Optimizer: SGD with momentum = 0.9 and weight decay = 0.001

Loss Function: Cross Entropy

Validation Method: Directly input all the validation data into the model

LR Scheduling: The model is trained for 80 epochs, with the learning rate initially set to 0.01 and decay at epoch 30 and 60 by a factor of 0.1

Data Augmentation:

1. Random Resize Crop: Size 32 and scale ranging from 0.7 to 1.0
2. Random Horizontal Flip: With probability 0.5

3. Random Rotation: Rotation angle ranging from -15° to 15°
4. Normalize: Mean = [0.507,0.487,0.44], STD = [0.267,0.256,0.276]

Model Architecture:

In order to adapt DenseNet121 to a smaller input image size of 32x32, we make a couple of modifications. First, we replace the initial convolution layer, which originally have a kernel size of 7x7, with a new layer having a 3x3 kernel size.

Furthermore, we remove the first max pooling layer, batch normalization and ReLU activation function. These adjustments are essential to retain important features.

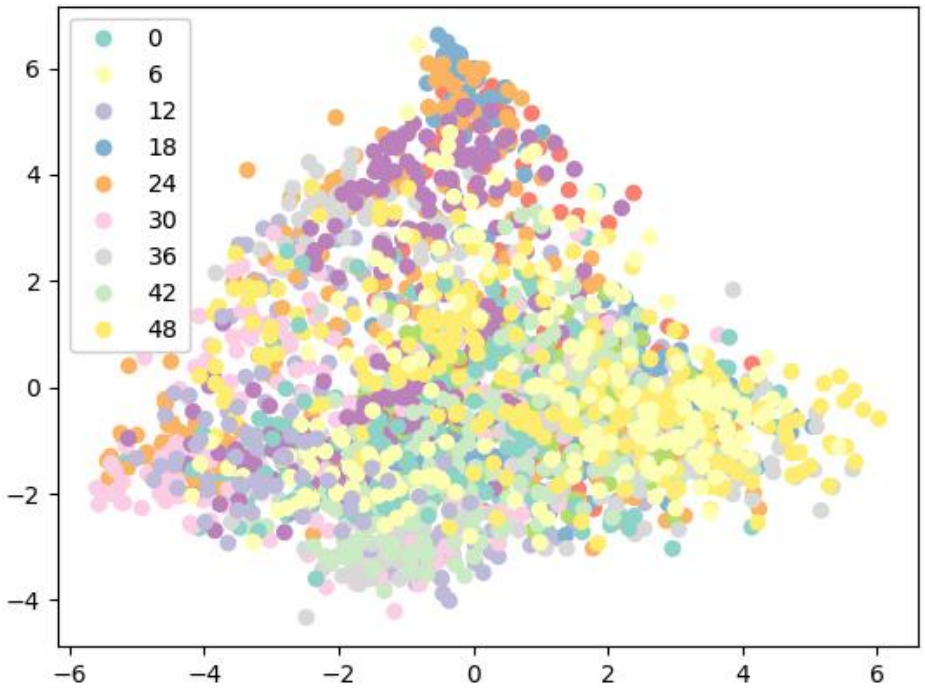
4. Implementation details of model B

There are two key distinctions between A and B.

Firstly, the model employed in B has undergone pre-training on ImageNet, significantly reducing its training time. Consequently, the total training epochs have been reduced to 15.

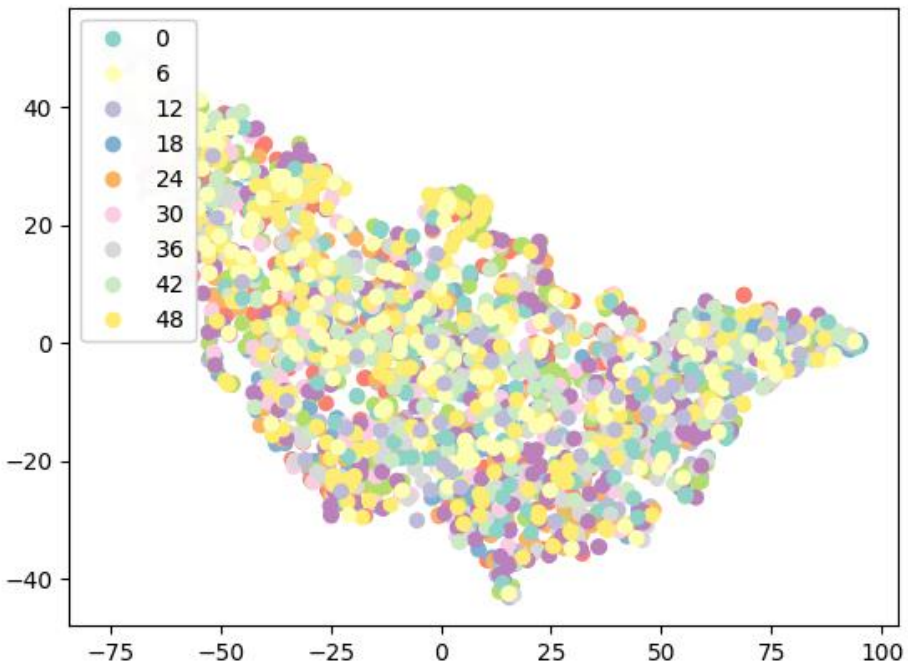
Secondly, considering that ImageNet data comprises images with dimensions of 224x224, we have adjusted the random resize crop size from 32 to 224 in order to fully leverage the model's feature extraction capabilities.

5. Visualization of learned representations by PCA

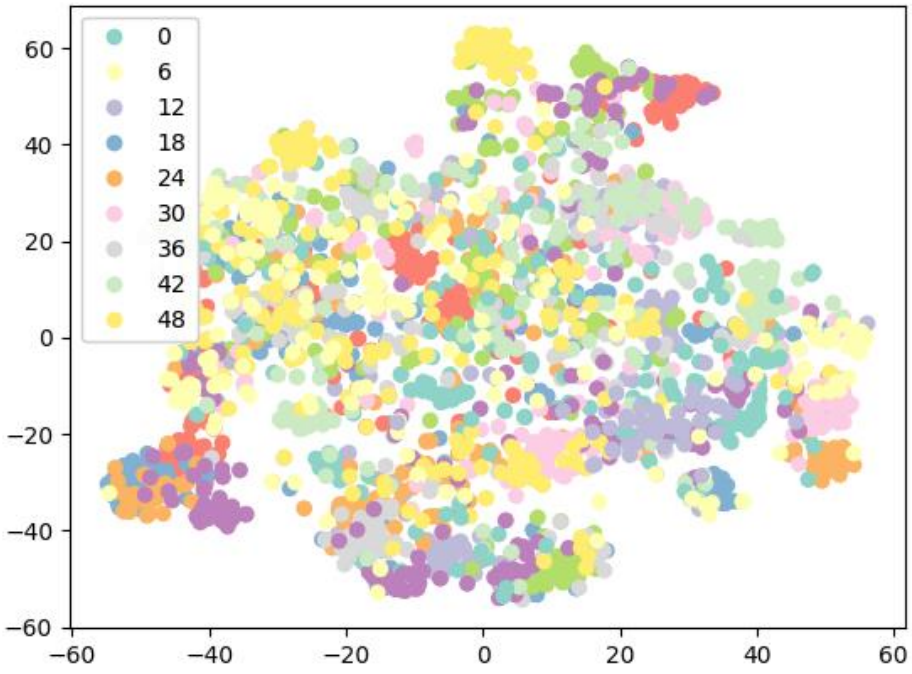


The data points are densely clustered, making it challenging to partition the representations into 50 distinct clusters, owing to the inherent crowding issue arising from the curse of dimensionality.

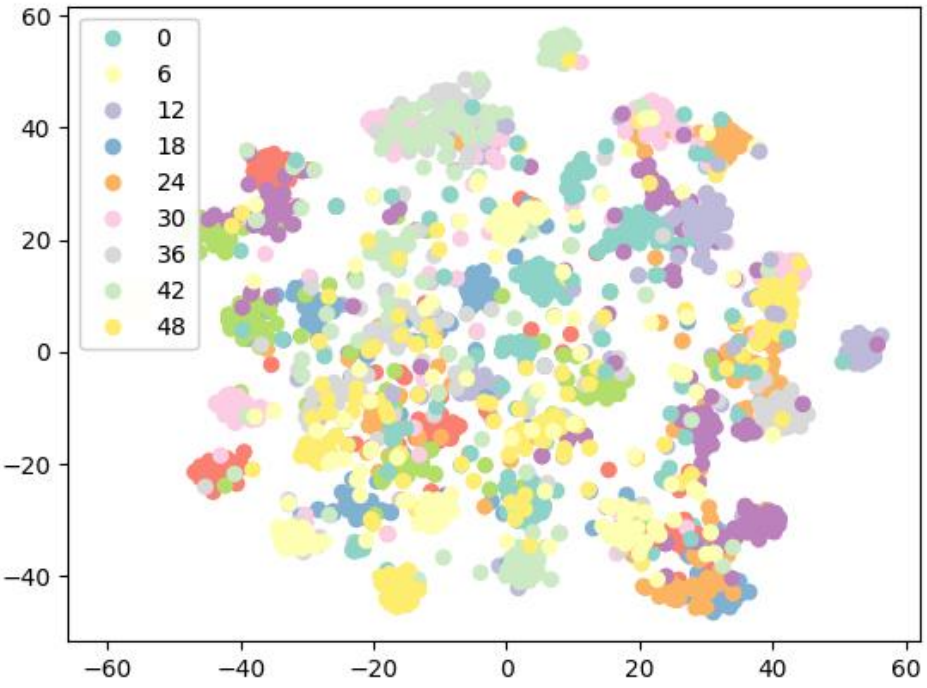
6. Visualization of learned representations by t-SNE



Epoch 0



Epoch 15



Epoch 80

The distribution of the clusters appears to be less dense compared to the result of PCA, indicating that t-SNE mitigate the issue caused by the curse of dimensionality. Additionally, as we continue to train the model, it becomes increasingly feasible to separate clusters belonging to different classes.

Problem 2

1. Implementation details of SSL

The self-supervised pre-training method employed is BYOL. Throughout the training process, a batch size of 160 is utilized, spanning a total of 40 training epochs. The initial learning rate is set at 0.0003 and would decreases to 0.00003 at epoch 20. Also, we use the Adam optimizer to update the model.

2. Result of setting A to E

Setting	Pre-training	Fine-tuning	Validation accuracy
A	-	Train full model	0.3818
B	w/ label	Train full model	0.4335
C	w/o label	Train full model	0.4138
D	w/ label	Train classifier only	0.3300
E	w/o label	Train classifier only	0.0640

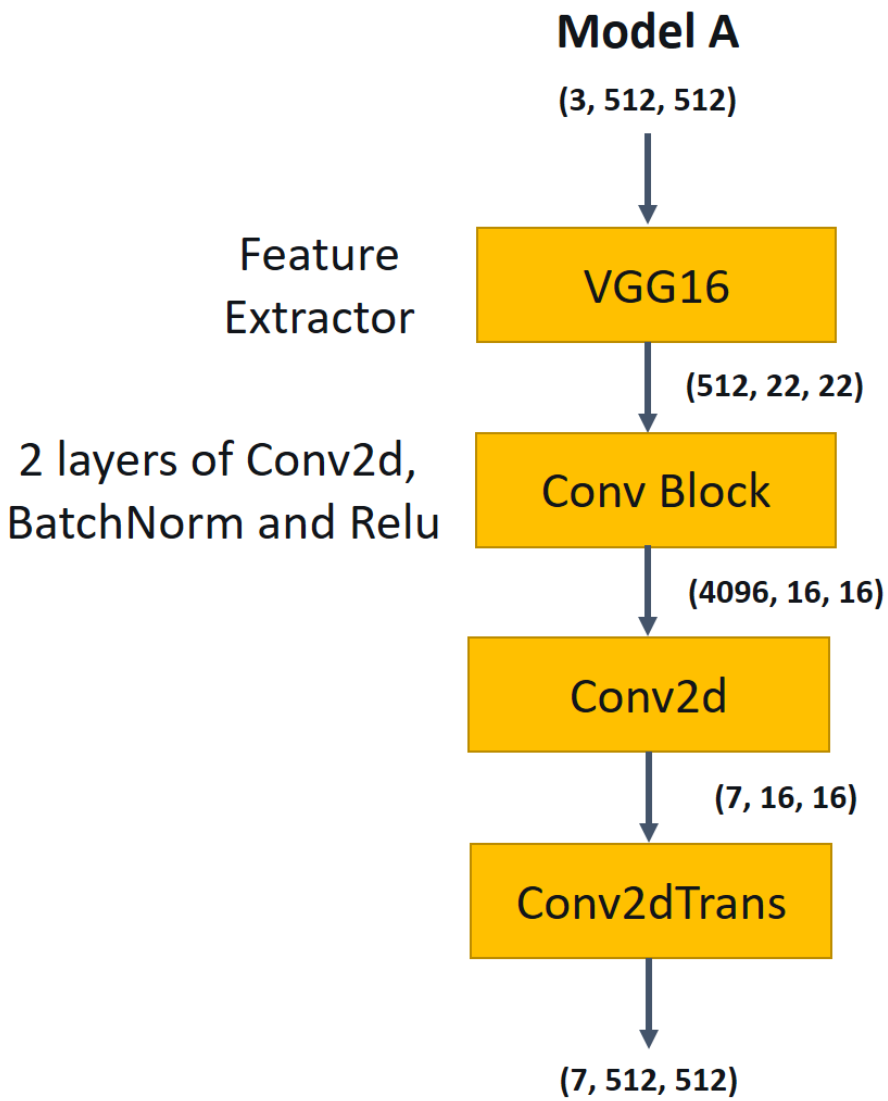
The result of setting A shows that when the model is trained directly using the office dataset, it can attain an accuracy of 38.18% upon completion of the training.

Furthermore, the outcomes of setting B and C demonstrates that, using the same fine-tuning parameters, pre-training our model before training leads to increased accuracy. Specifically, the accuracy reaches 43.35% with labels and 41.38% without labels. These results indicate that self-supervised pre-training performs only slightly less effectively than the supervised pre-training.

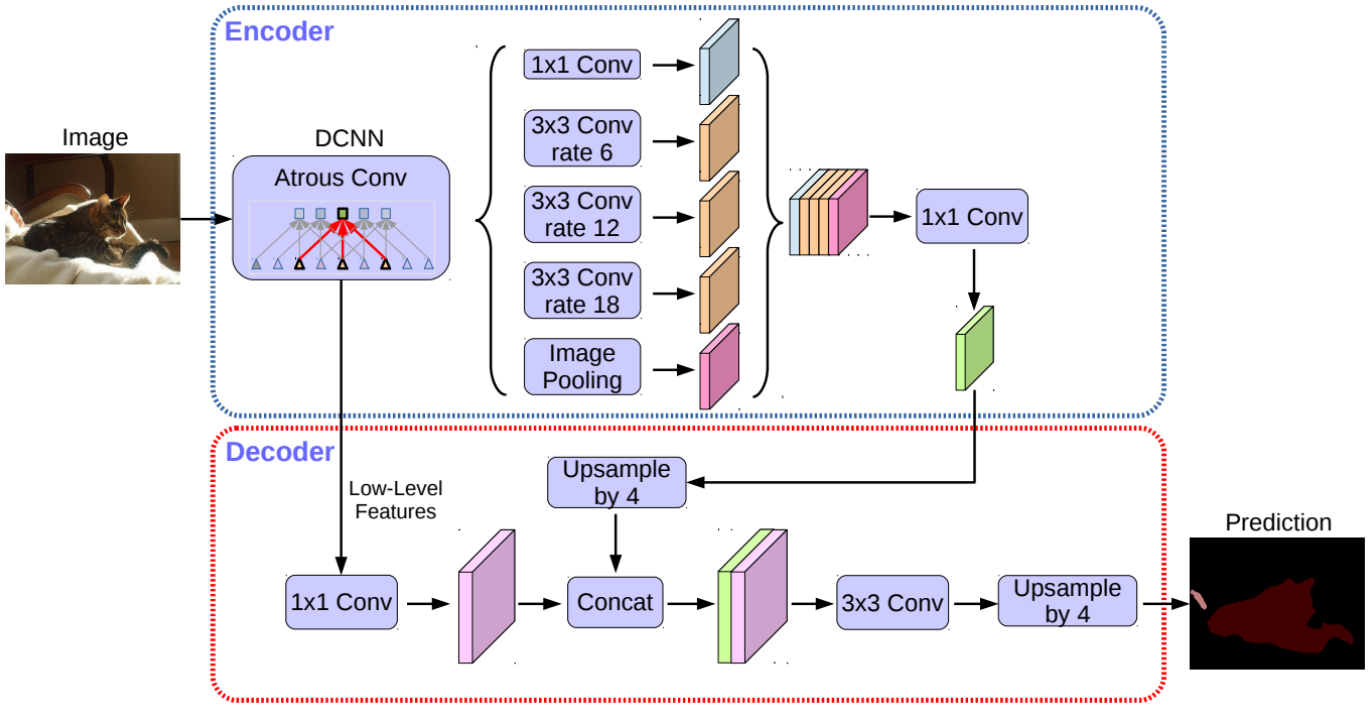
Moreover, the results from settings D and E reveal that when the feature extractor remains frozen during fine-tuning, the model pre-trained using self-supervised methods struggles to train effectively, whereas the model pre-trained with labels continues to perform well. This outcome may be attributed to the different in task nature, as our downstream task involves classification, while the self-supervised training does not.

Problem 3

1. Architecture of my model A (VGG16-FCN32s)



2. Architecture of my model B (DeepLabv3+)



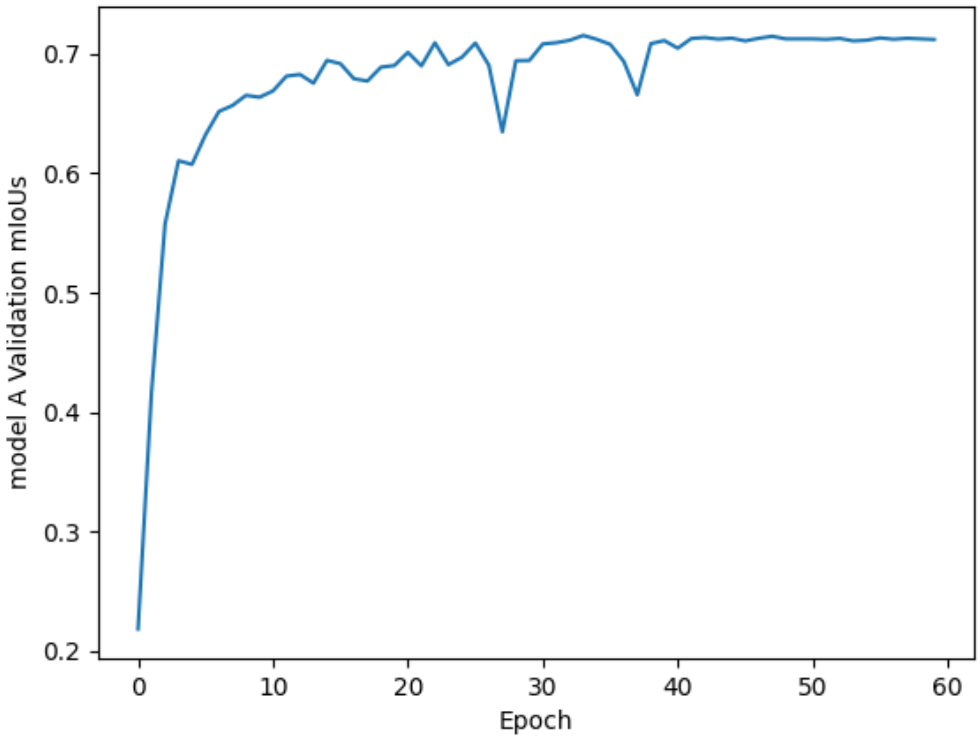
Chen, Liang-Chieh, et al. "Encoder-decoder with atrous separable convolution for semantic image segmentation. " *Proceedings of the European conference on computer vision (ECCV)*. 2018.

Compared to FCN-32s, DeepLabv3+ has a more sophisticated and context-aware architecture. Firstly, it utilizes atrous (dilated) convolutions to capture multi-scale contextual information effectively. Secondly, it has a spatial pyramid module that extracts feature at multiple scales. Thirdly, it includes a decoder module that refines the segmentation map and fuses information from different scales.

In summary, DeepLabv3+ incorporates various architectural enhancements to capture contextual information and handle multi-scale features, especially for challenging scenes and objects.

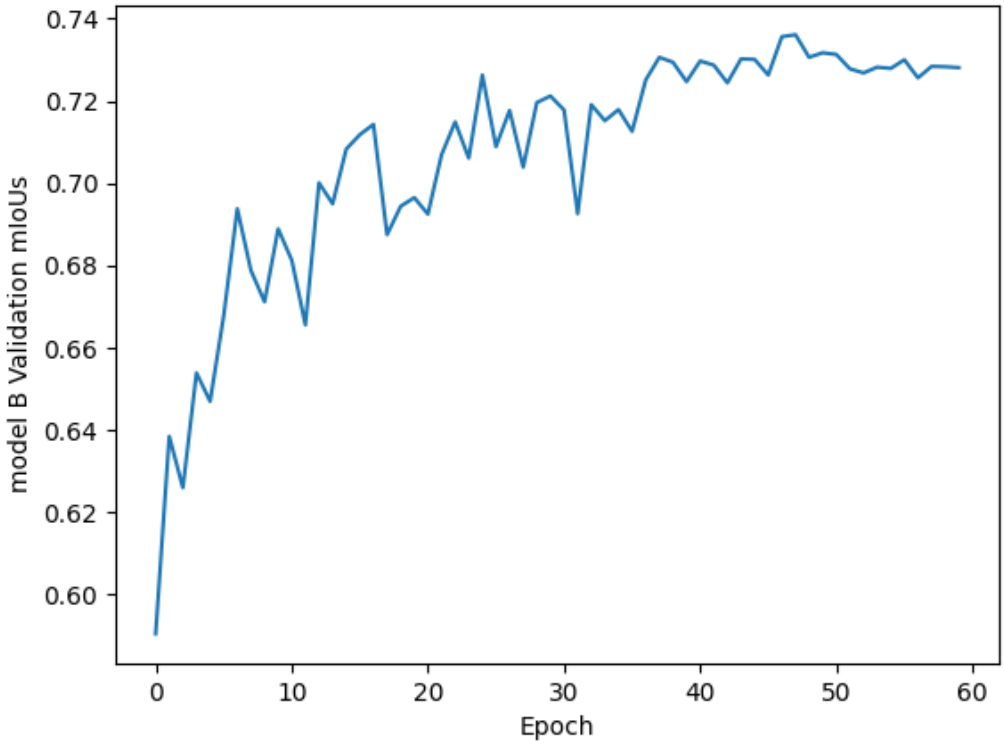
3. mIoUs of two models on the validation set

Model A



Best Validation mIoU: 0.7156

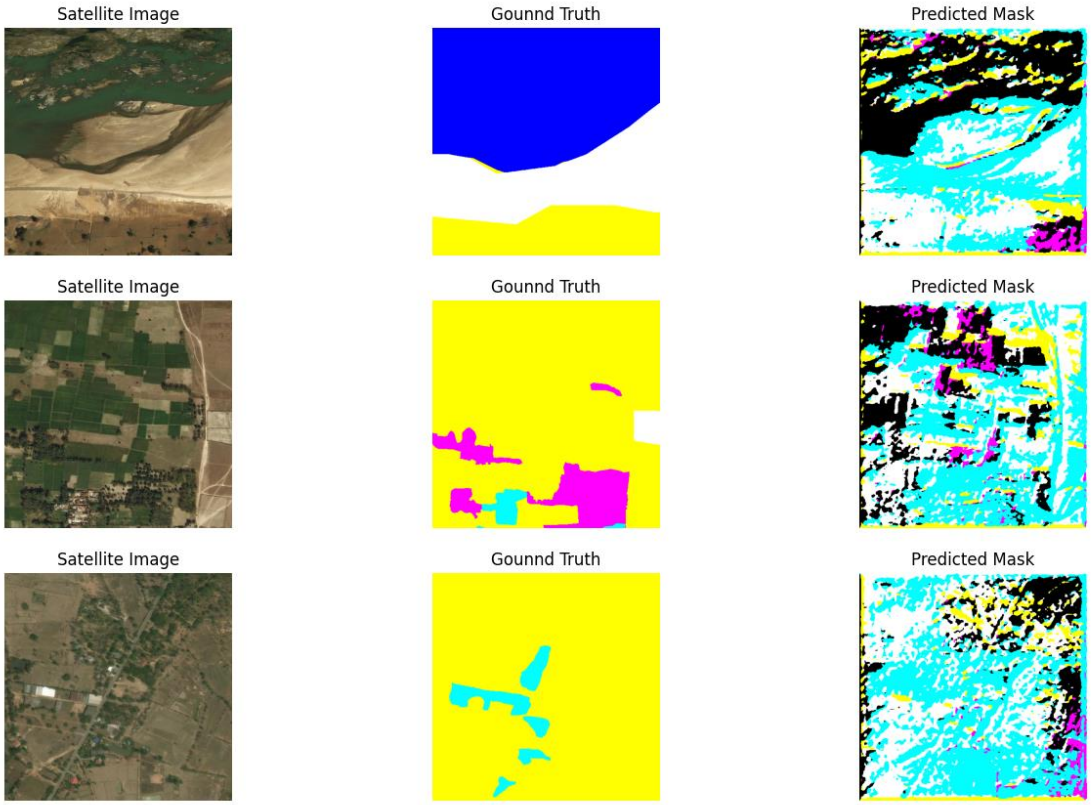
Model B



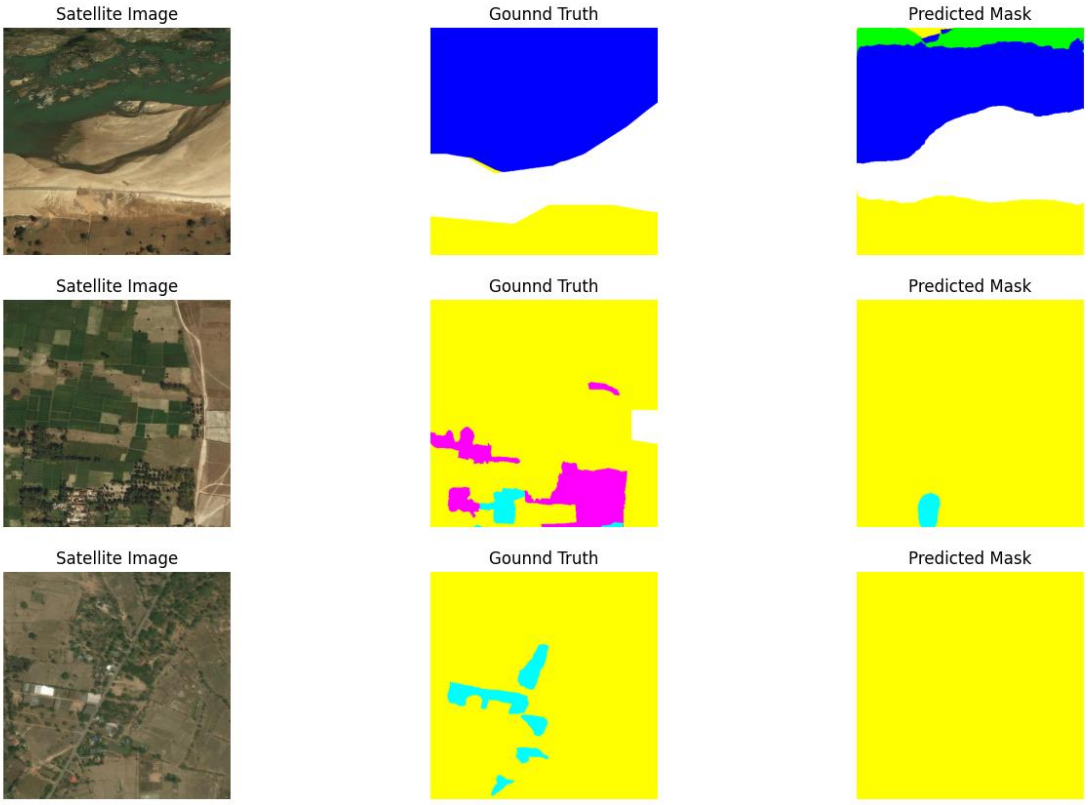
Best Validation mIoU: 0.7361

4. Visualization of predicted segmentation mask

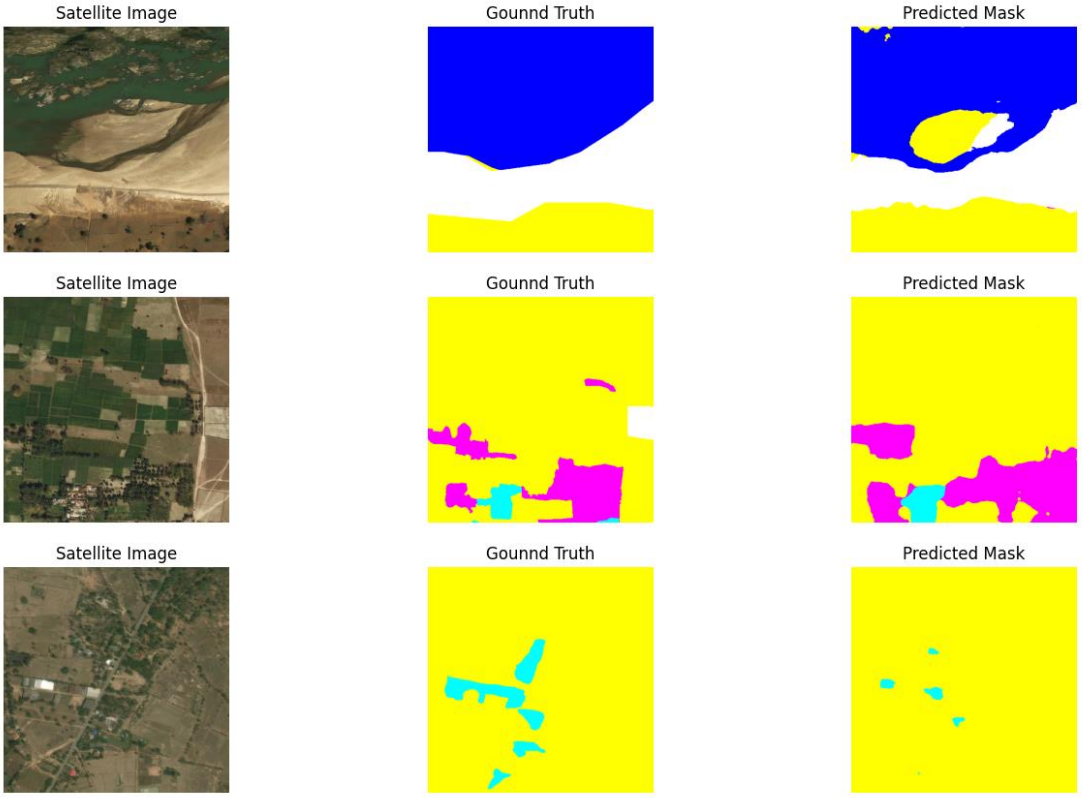
Early Stage



Middle Stage



Final Stage


Satellite Image
Gounnd Truth
Predicted Mask
Satellite Image
Gounnd Truth
Predicted Mask
Satellite Image
Gounnd Truth
Predicted Mask