DLCV HW1 Report

R13921052 劉祥暉

Part 1

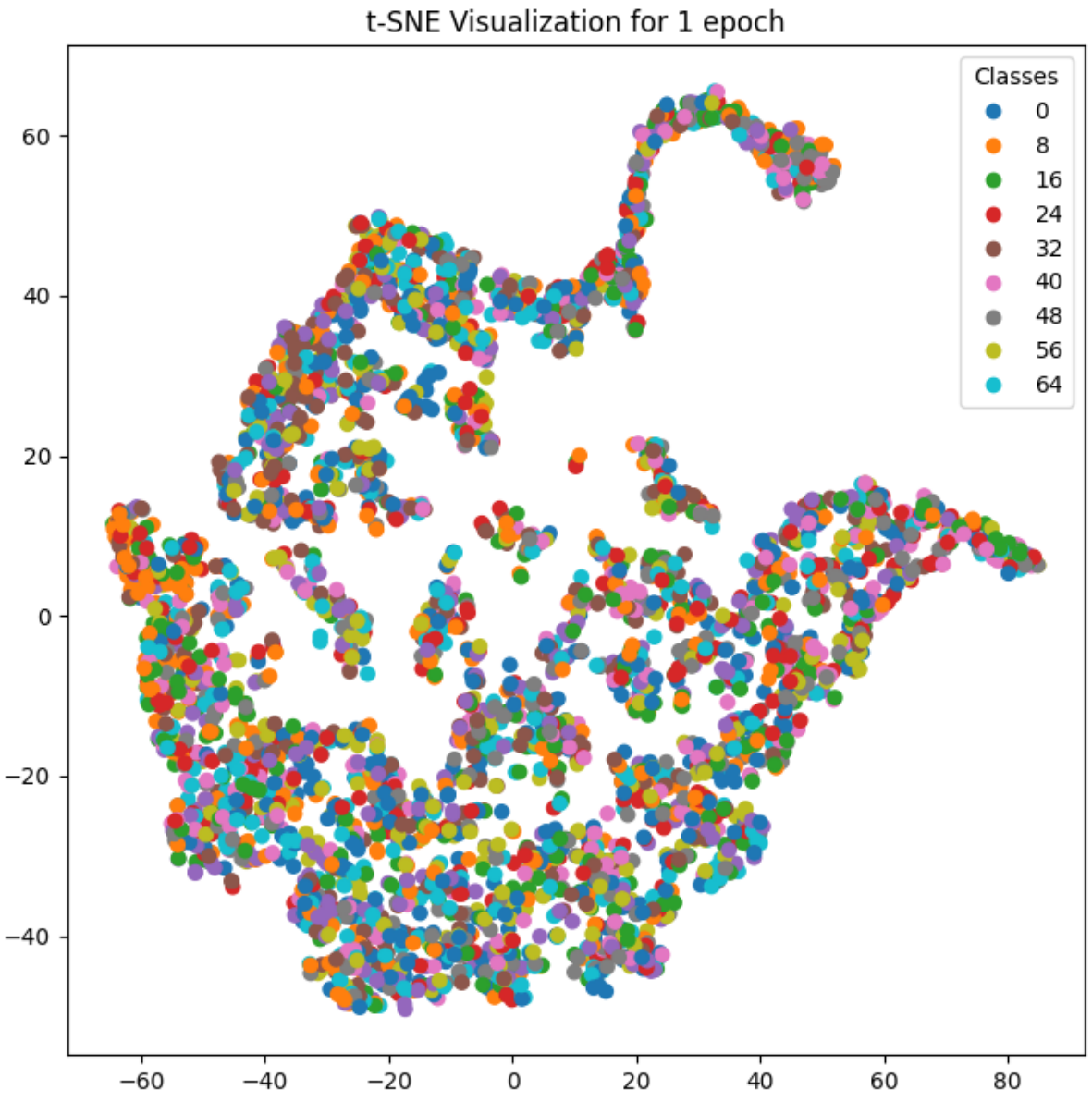
1. Implementation details for pre-training

- Use BYOL to pre-train ResNet50 backbone
- Augmentation: Trivial Augmentation (https://arxiv.org/abs/2103.10158)
- Optimizer: Adam
- Scheduler: OneCycleLR with max learning rate = 0.01 (https://pytorch.org/docs/stable/generated/torch.optim.lr_scheduler.OneCycleLR.html)
- batch size: 64
- early stopping patience = 15 epochs

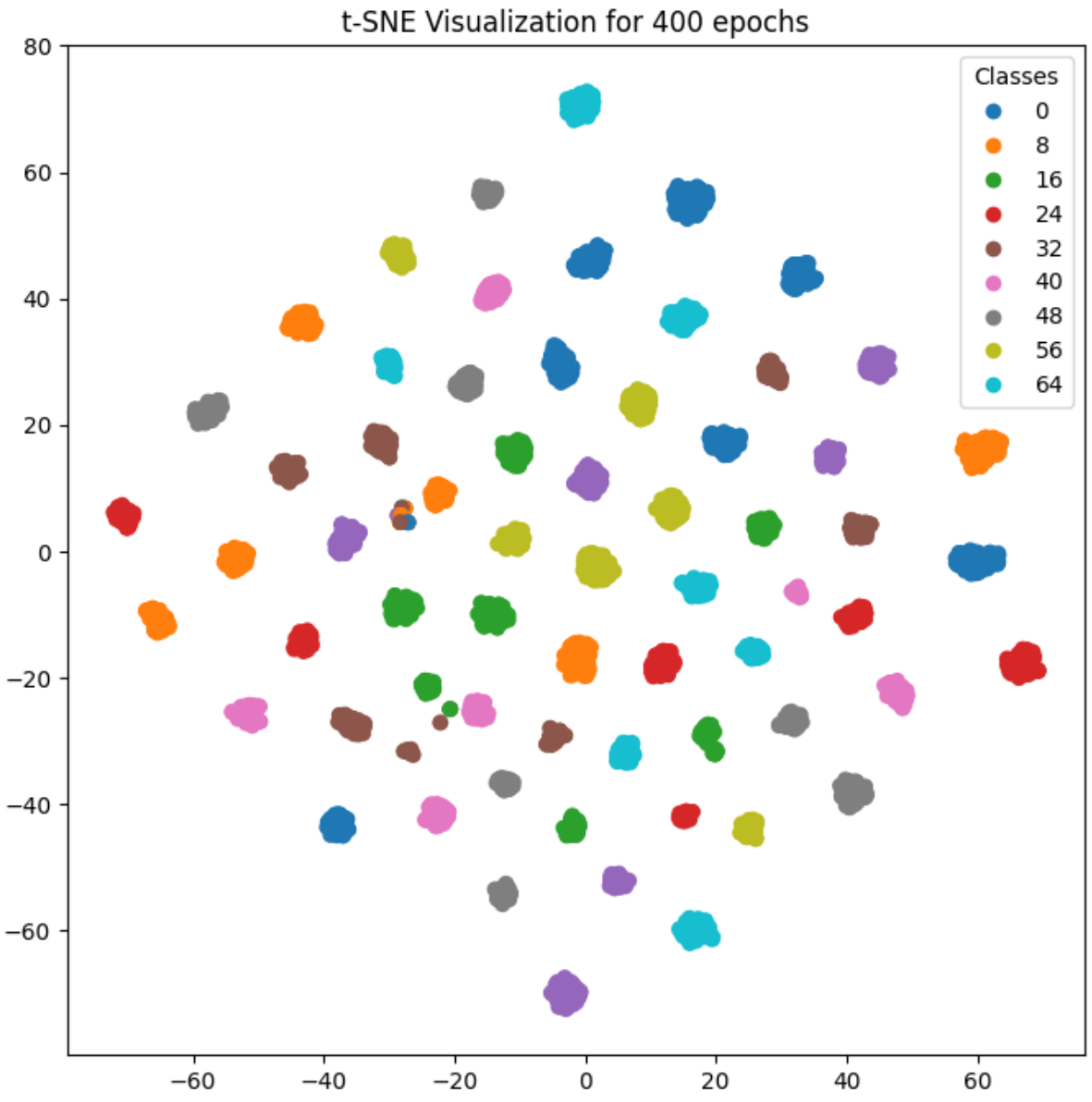
2. Office-Home dataset performance

Setting	Validation Accuracy
A	49%
B	55%
C	50%
D	33%
E	9%

3. t-SNE



t-SNE plot after the first epoch



t-SNE plot after the last (400th) epoch

Brief explanation:

t-SNE圖代表維度壓縮之後的資料分群情況，如可以看到同一個class的data points明顯可被分群，代表模型可根據特徵做出有依據的分群並且有好的表現。上圖可看見在第一個epoch過後不同顏色的data points都混雜在一起，代表此時模型效果欠佳，有可能是因為classifier尚未發展成熟或是pre-train模型表現不好；而在400個epochs之後，每種顏色的data points都非常緊密地聚集，且不同群之間界線明顯，代表模型可以很好地做出正確分類。

Part 2

1. Network architecture of model A

```
VGGFCN(
  (features): Sequential(
    (0-4): [Conv2d(3, 64), ReLU, Conv2d(64, 64), ReLU, MaxPool2d]
    (5-9): [Conv2d(64, 128), ReLU, Conv2d(128, 128), ReLU, MaxPool2d]
    (10-16): [Conv2d(128, 256), ReLU, Conv2d(256, 256), ReLU, Conv2d(2
    (17-23): [Conv2d(256, 512), ReLU, Conv2d(512, 512), ReLU, Conv2d(5
    (24-30): [Conv2d(512, 512), ReLU, Conv2d(512, 512), ReLU, Conv2d(5
  )
  (conv6): Conv2d(512, 4096, kernel_size=(7, 7))
  (conv7): Conv2d(4096, 4096, kernel_size=(1, 1))
  (score_fr): Conv2d(4096, 7, kernel_size=(1, 1))
  (upsample): ConvTranspose2d(7, 7, kernel_size=(32, 32), stride=(32,
  (relu): ReLU(inplace=True)
)
```

2. Network architecture of model B

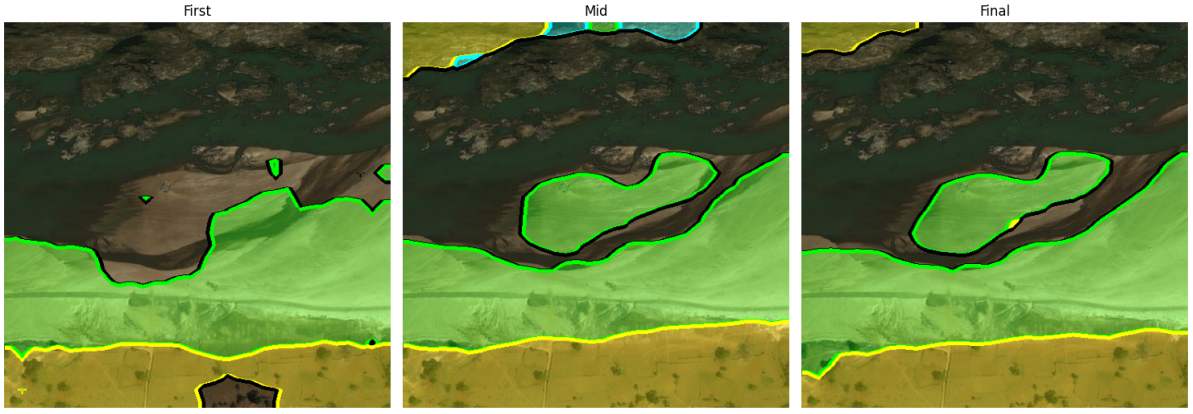
```
Deeplabv3_Mobilenet_Model(
  (model): DeepLabV3(
    (backbone): IntermediateLayerGetter(
      (0): Conv2d + BatchNorm + Hardswish (input: 3 -> 16)
      (1-3): InvertedResidual (Conv layers, groups, SE blocks)
      (4-7): InvertedResidual (Strided convolutions, SE blocks)
      (8-15): InvertedResidual (Deeper convolutions, more SE blocks)
    )
    (classifier): ASPP + Conv2d + BatchNorm
  )
)
```

Backbone的部分，model A使用了pretrained resnet50，而model B則使用pretrained mobilenet_v3。model B的激活函數也和單純使用ReLU的model A不同，model B使用 Hardswish 和 ReLU6 激活函數，可減少計算負擔。

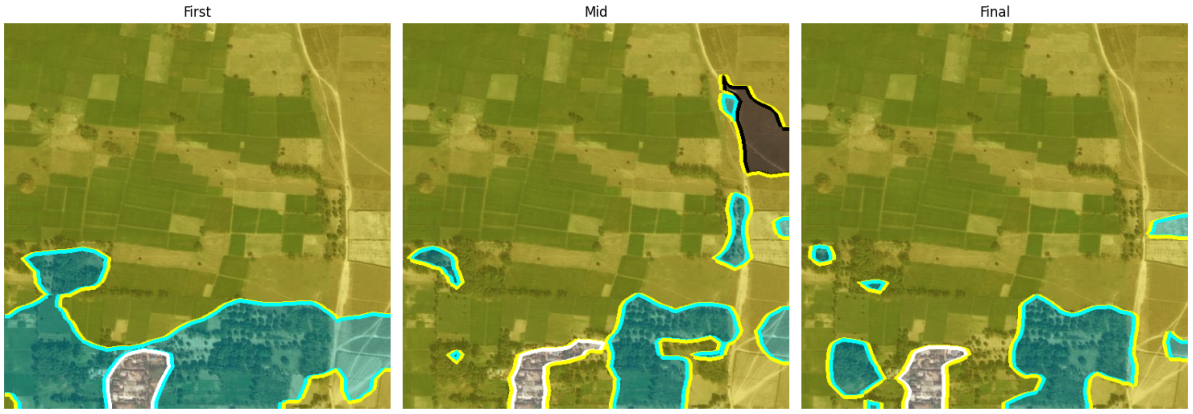
3. Performance

Model	**mIoU**
VGG16-FCN32s	0.3579
Deeplabv3_Mobilenet	0.7048

4. Masks during different stages



validation/0013_sat.jpg



validation/0062_sat.jpg

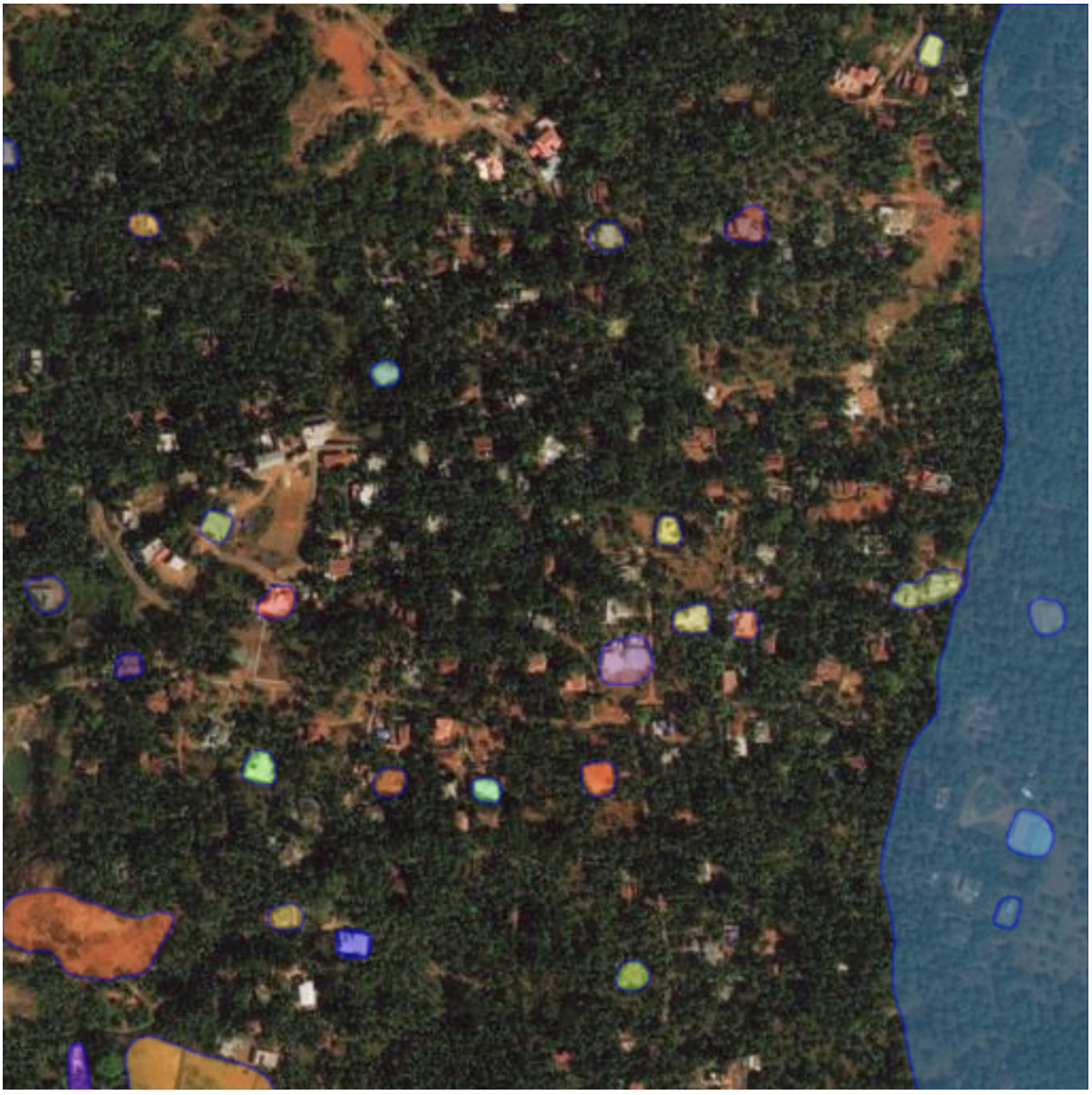
validation/0104_sat.jpg

5. SAM

Method:

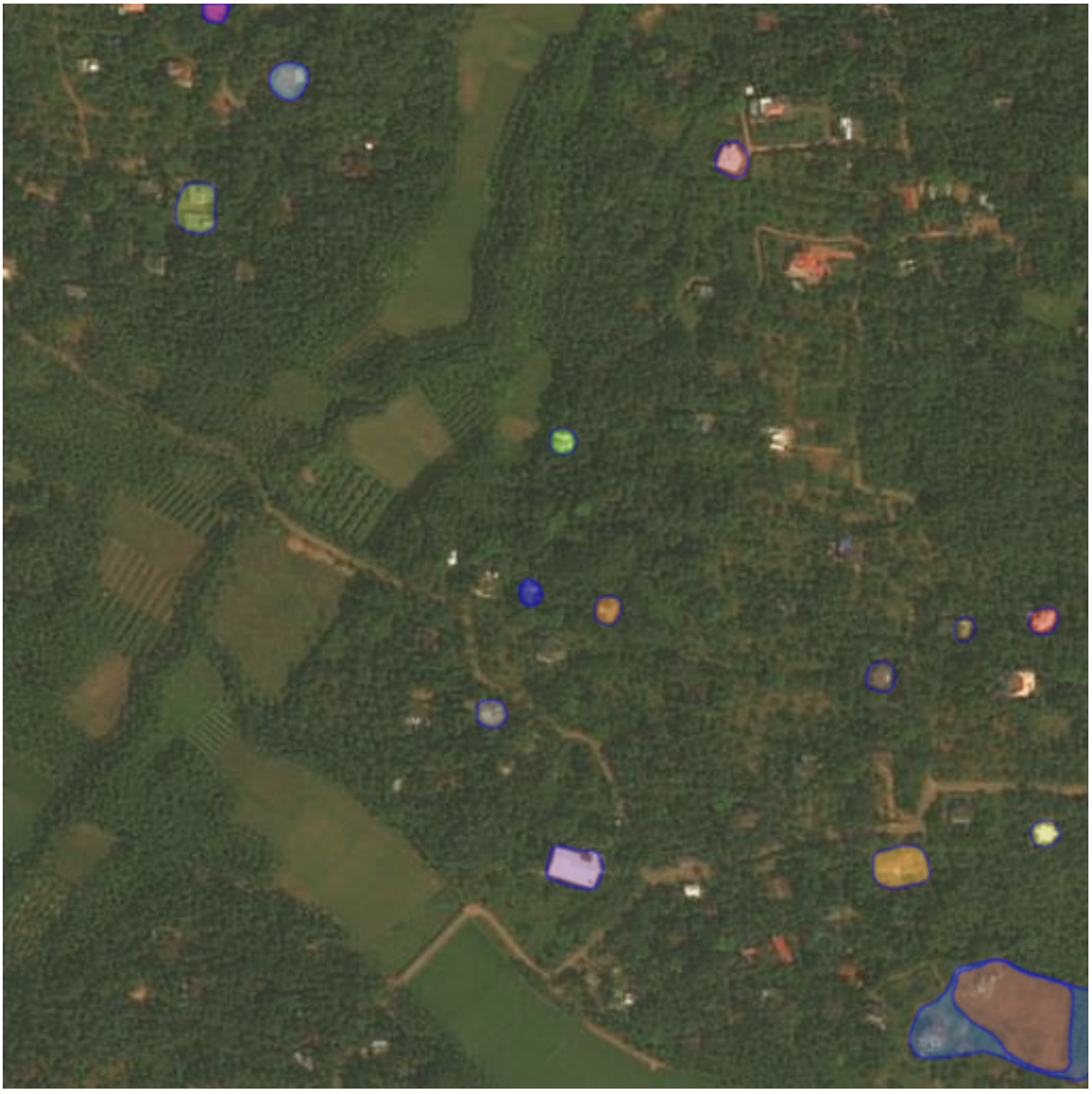
use the “everything” prompt on the SAM demo page (https://segment-anything.com/demo)

Result images:



validation/0011_sat.jpg

validation/0162_sat.jpg



validation/0013_sat.jpg