NANYANG TECHNOLOGICAL UNIVERSITY

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

AI6126 Advanced Computer Vision

Project 1:

CelebAMask Face Parsing

Zheng Weixiang
wzheng014@e.ntu.edu.sg
G2103278G

Document	ACV Project 2 Report
Name	Zheng Weixiang
Matriculation Number	G2103278G

Introduction

This is the report for AI6126 Advanced Computer Vision Project 1 CelebAMask Face Parsing. In this report, I will briefly describe the dataset, the task I am working on and the result I achieved.

Dataset

CelebAMask-HQ dataset is a large-scale face image dataset that has 30,000 hgh-resolution face images selected from the CelebA dataset. The mask of CelebAMask were manually annotated with the size of 512x512 and 19 classes including all facial components and accessories such as skin, nose, eyes, eyebrows, ears, mouth, lip, hair, hat, eyeglass, earring, necklace, neck, and cloth.[1] In this project, we will only work on a subset of the dataset will consist of only 7000 images. We will further divide the dataset into 3 parts: training set which consists of 5000 images with 5000 corresponding masks, validation set which consists of 1000 images with 1000 corresponding masks, and a testing set which consists of 1000 images without masks. Our task is to correctly label every pixel in the image. The classes available are from 0 to 18, in total 19 classes. They are "background", "skin", "nose", "eye_glasses", "left_eye", "right_eye", "left_brow", "right_brow", "left_ear", "right_ear", "mouth", "upper_lip", "lower_lip", "hair","hat", "earing", "necklace", "neck", and "cloth".

Specs of your training machine

I used the school provided environment. The specification is as follows:
Environment: SCSEGPU
Number of GPUs: 1
GPU: RTX 2080 Ti
CPU: Intel(R) Xeon(R) Gold 6240 CPU @ 2.60GHz

Models

I've tried several models including:

1. DeepLabV3+ (CVPR'2018) [2]
   - R-50 20000 iterations
   - R-50 40000 iterations
   - R-101 20000 iterations
   - R-101 40000 iterations
2. PSANet (ECCV'2018) [2]
   - R-50 80000 iterations
   - R-101 80000 iterations
3. Swin Transformer (ICCV'2021) [2] [3]
   - UperNet Swin-T ImageNet-1K 224x224 160k iterations
   - UperNet Swin-B ImageNet-1K 224x224 160k iterations

For all of DeepLabV3+ models and PSANet models, I have mIoU around 70-73. With fine-tune of the model, I noticed the mIoU increases particularly when the learning rate is relatively low, and then the necklace IoU shows a sign of increasing therefore pumps up the overall mIoU. The best performance I can get using this method is 73.5972618437 on Coda leaderboard & 73.87 on SCSE GPU Server. Among all of them, the best empirical result is from Swin Transformer (ICCV'2021) [3], which has mIoU of 76.7753074062(17th entry) and that is the resut from Swin-T model [3]. The overall best result on leaderboard is 77.5776134513 (19th entry) and that is the result from Swin-B model [3].

The best result

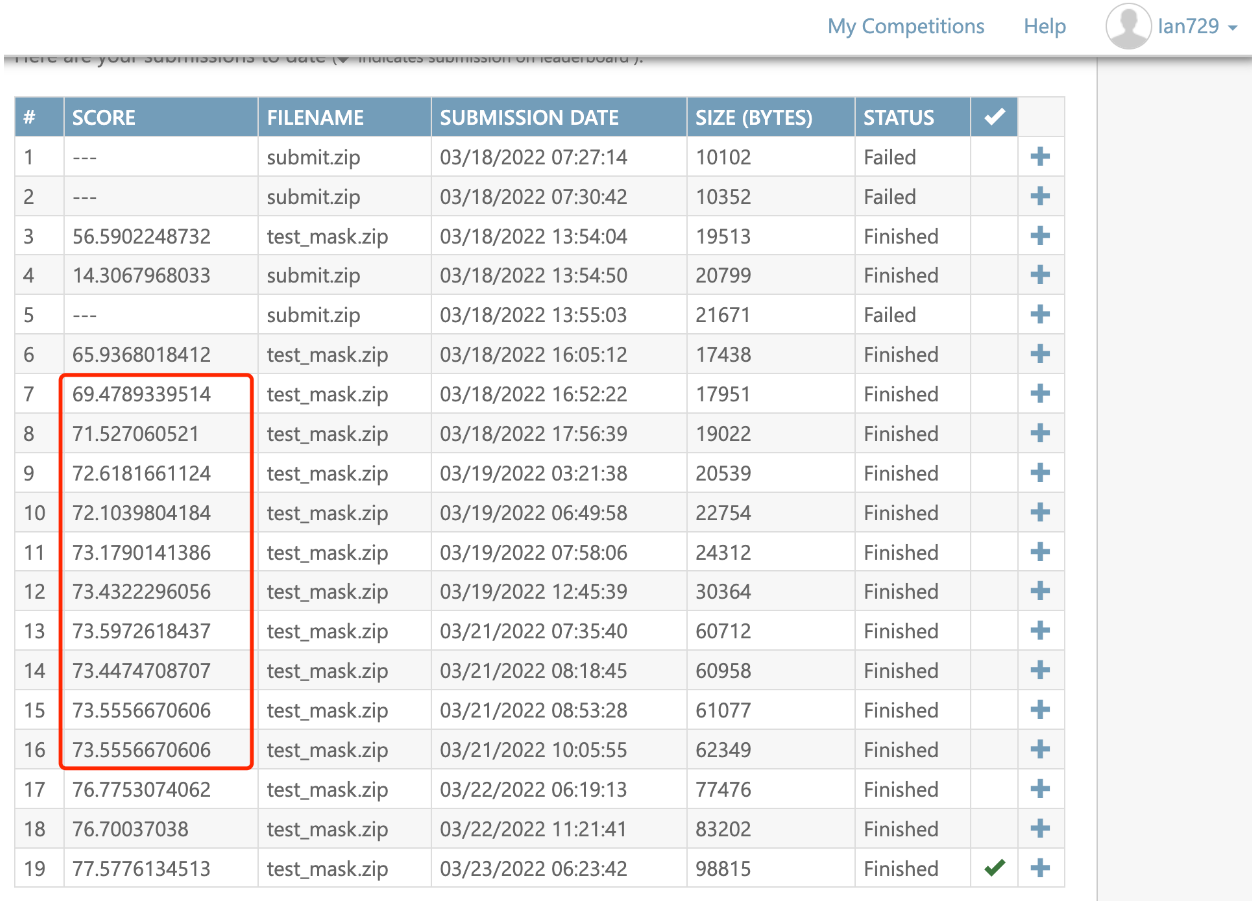
Score of 78.76 on validation set and score of 77.57 on Coda testing set.

```
2022-03-23 01:22:31,684 - mmseg - INFO - 
+-------------+-------+-------+
|    Class    |  IoU  |  Acc  |
+-------------+-------+-------+
|  background | 92.92 | 96.44 |
|     skin    | 93.17 | 96.66 |
|     nose    | 88.54 | 93.45 |
| eye glasses |  84.6 | 94.16 |
|   left eye  | 82.29 | 89.85 |
|  right eye  |  82.4 | 90.43 |
|  left brow  | 76.12 | 86.03 |
|  right brow | 74.84 | 84.45 |
|   left ear  | 77.97 | 86.79 |
|  right ear  |  77.4 | 87.01 |
|    mouth    | 85.62 | 92.95 |
|  upper lip  | 81.31 | 89.79 |
|  lower lip  |  83.3 | 90.19 |
|     hair    | 91.42 | 95.71 |
|     hat     | 76.51 |  84.8 |
|   earring   |  56.3 | 69.07 |
|   necklace  | 29.57 | 35.31 |
|     neck    | 83.47 | 91.18 |
|    cloth    | 78.76 | 85.57 |
+-------------+-------+-------+
2022-03-23 01:22:31,685 - mmseg - INFO - Summary:
2022-03-23 01:22:31,685 - mmseg - INFO - 
+-------+-------+-------+
|  aAcc |  mIoU |  mAcc |
+-------+-------+-------+
| 95.13 | 78.76 | 86.31 |
+-------+-------+-------+
```

Number of parameters

```
>>> print(sum(p.numel() for p in model.parameters() if p.requires_grad))
121178846
```

Screenshots of Coda submitted results



Here are your submissions to date (✔ indicates submission on leaderboard).

#	SCORE	FILENAME	SUBMISSION DATE	SIZE (BYTES)	STATUS	✔	
1	---	submit.zip	03/18/2022 07:27:14	10102	Failed		+
2	---	submit.zip	03/18/2022 07:30:42	10352	Failed		+
3	56.5902248732	test_mask.zip	03/18/2022 13:54:04	19513	Finished		+
4	14.3067968033	submit.zip	03/18/2022 13:54:50	20799	Finished		+
5	---	submit.zip	03/18/2022 13:55:03	21671	Failed		+
6	65.9368018412	test_mask.zip	03/18/2022 16:05:12	17438	Finished		+
7	69.4789339514	test_mask.zip	03/18/2022 16:52:22	17951	Finished		+
8	71.527060521	test_mask.zip	03/18/2022 17:56:39	19022	Finished		+
9	72.6181661124	test_mask.zip	03/19/2022 03:21:38	20539	Finished		+
10	72.1039804184	test_mask.zip	03/19/2022 06:49:58	22754	Finished		+
11	73.1790141386	test_mask.zip	03/19/2022 07:58:06	24312	Finished		+
12	73.4322296056	test_mask.zip	03/19/2022 12:45:39	30364	Finished		+
13	73.5972618437	test_mask.zip	03/21/2022 07:35:40	60712	Finished		+
14	73.4474708707	test_mask.zip	03/21/2022 08:18:45	60958	Finished		+
15	73.5556670606	test_mask.zip	03/21/2022 08:53:28	61077	Finished		+
16	73.5556670606	test_mask.zip	03/21/2022 10:05:55	62349	Finished		+
17	76.7753074062	test_mask.zip	03/22/2022 06:19:13	77476	Finished		+
18	76.70037038	test_mask.zip	03/22/2022 11:21:41	83202	Finished		+
19	77.5776134513	test_mask.zip	03/23/2022 06:23:42	98815	Finished	✔	+

The detail of best performing model

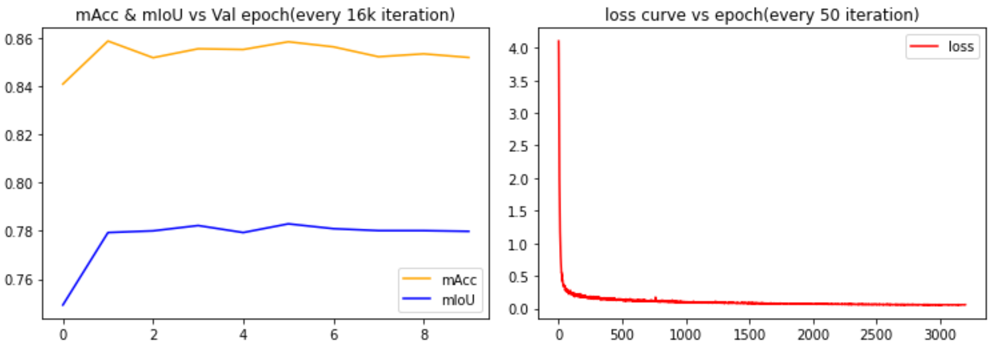
It uses a pretrained checkpoint: swin_base_patch4_window7_224.pth. It extends the config of upernet_swin_tiny_patch4_window7_512x512_160k_pretrain_224x224_1K.py. In addition, it overwrites the backbone with a more capable model.

```
    backbone=dict(
        embed_dims=128, depths=[2, 2, 18, 2], num_heads=[4, 8, 16, 32]),
    decode_head=dict(in_channels=[128, 256, 512, 1024], num_classes=19),
    auxiliary_head=dict(in_channels=512, num_classes=19))
```

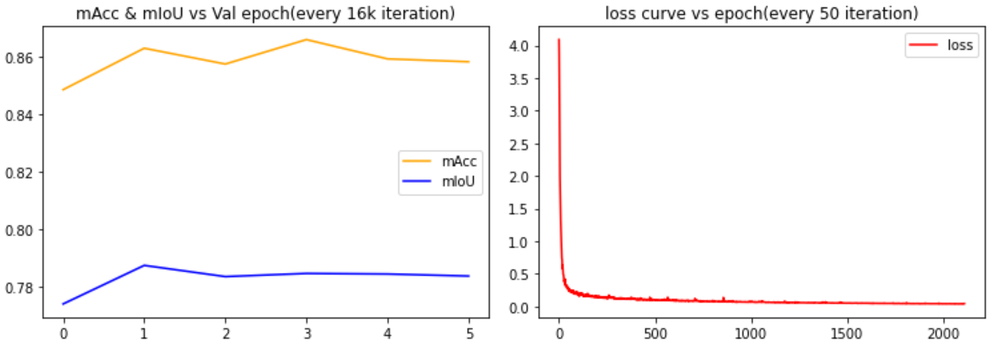
It runs 160k iterations in total, evaluate every 16k iterations, so there will be 10 evaluation epochs and 10 mIoU and mAcc in total. During the 160k iterations, log is recorded every 50 iterations. There will be 3200 entries of log recordings. Then I will show the training curves. They are plotted using plot_curve.ipynb, it is attached in the zip file as well.

Training Curves

The mIoU and mAcc & Loss plot of Swin-T model



The mIoU and mAcc & Loss plot of Swin-B model



Saved checkpoints

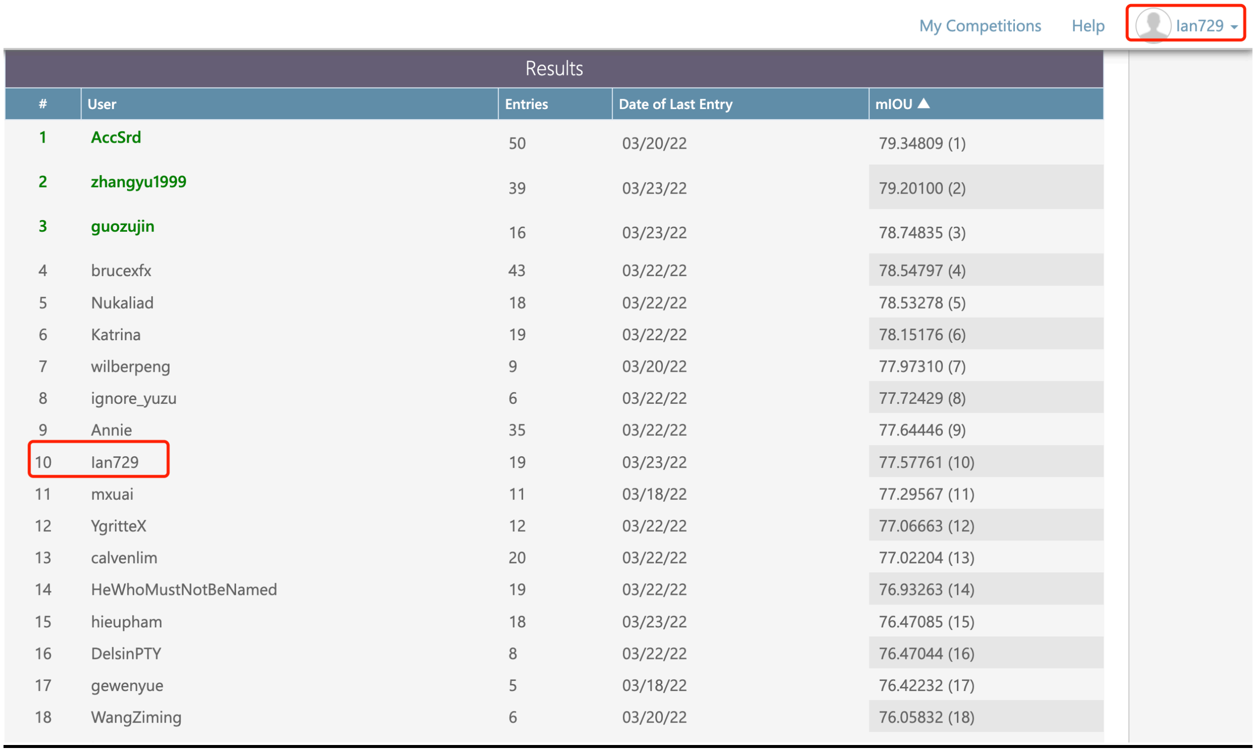
https://entuedu-my.sharepoint.com/:u:/r/personal/wzheng014_e_ntu_edu_sg/Documents/ACV/iter_12000_finetune_7387.pth?csf=1&web=1&e=hnJMgW
this is for the 15th entry 73.5556670606. It uses config file baseline.py. (attached in zip)

https://entuedu-my.sharepoint.com/:u:/r/personal/wzheng014_e_ntu_edu_sg/Documents/ACV/iter_2000_finetune_7384.pth?csf=1&web=1&e=rNUINi
this is for the 13th entry 73.5972618437, DeepLabV3+ R-50 40000 iteration model, It uses config file baseline.py (attached in zip)

https://entuedu-my.sharepoint.com/:u:/r/personal/wzheng014_e_ntu_edu_sg/Documents/ACV/iter_96000_swin_7829_palette.pth?csf=1&web=1&e=3rzNAu
this is for the 17th entry, SWIN-T model, pretrained ImageNet-1K, 160k iterations. It uses config file(attached in zip)
upernet_swin_tiny_patch4_window7_512x512_160k_pretrain_224x224_1K.py

https://entuedu-my.sharepoint.com/:u:/r/personal/wzheng014_e_ntu_edu_sg/Documents/ACV/iter_32000_swinb_7876.pth?csf=1&web=1&e=s6xghm
This is for the 19th entry, SWIN-B model, pretrained ImageNet-1K, 160k iterations. It uses config file upernet_swin_base_512x512_160k.py. (attached in zip)

The ranking on Coda
The best rank I got was on Mar 23rd 2022, which is top 10.
The screenshot is as follows:



Results

#	User	Entries	Date of Last Entry	mIOU ▲
1	AccSrd	50	03/20/22	79.34809 (1)
2	zhangyu1999	39	03/23/22	79.20100 (2)
3	guozujin	16	03/23/22	78.74835 (3)
4	brucexfx	43	03/22/22	78.54797 (4)
5	Nukaliad	18	03/22/22	78.53278 (5)
6	Katrina	19	03/22/22	78.15176 (6)
7	wilberpeng	9	03/20/22	77.97310 (7)
8	ignore_yuzu	6	03/22/22	77.72429 (8)
9	Annie	35	03/22/22	77.64446 (9)
10	Ian729	19	03/23/22	77.57761 (10)
11	mxuai	11	03/18/22	77.29567 (11)
12	YgritteX	12	03/22/22	77.06663 (12)
13	calvenlim	20	03/22/22	77.02204 (13)
14	HeWhoMustNotBeNamed	19	03/22/22	76.93263 (14)
15	hieupham	18	03/23/22	76.47085 (15)
16	DelsinPTY	8	03/22/22	76.47044 (16)
17	gewenyue	5	03/18/22	76.42232 (17)
18	WangZiming	6	03/20/22	76.05832 (18)

Reference

[1] Switchablenorms. (n.d.). *Switchablenorms/celebamask-HQ: A large-scale face dataset for face parsing, recognition, generation and editing.* GitHub. Retrieved March 23, 2022, from https://github.com/switchablenorms/CelebAMask-HQ

[2] Open-Mmlab. (n.d.). *Open-mmlab/mmsegmentation: Openmmlab semantic segmentation toolbox and benchmark.* GitHub. Retrieved March 23, 2022, from https://github.com/open-mmlab/mmsegmentation

[3] Open-Mmlab. (n.d.). *Mmsegmentation/configs/swin at master · open-mmlab/mmsegmentation*. GitHub. Retrieved March 23, 2022, from https://github.com/open-mmlab/mmsegmentation/tree/master/configs/swin