Cmpt 412

Project 2

Hao Ran Wei

301354701

Team name: hrwei_sfu
Team members: Justin Lin, Beatrice Chan

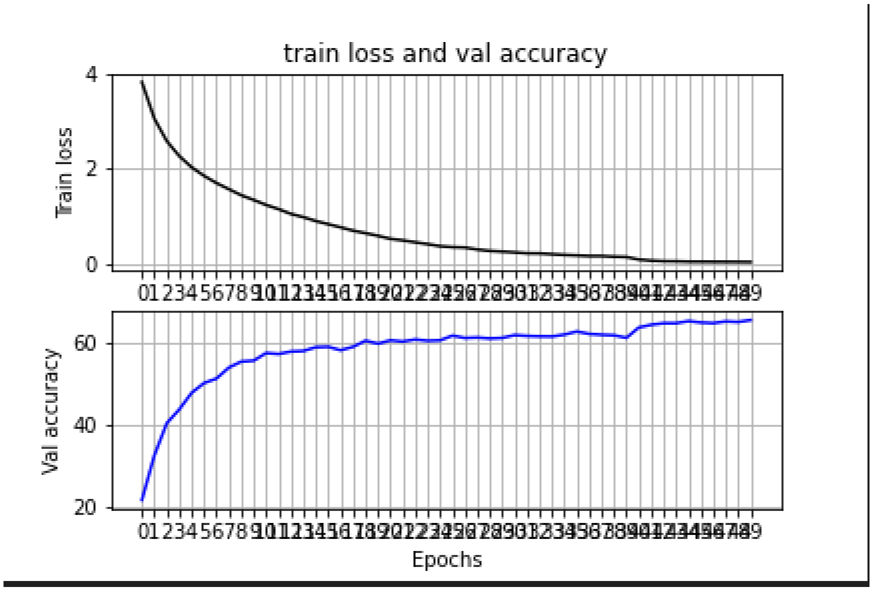
Part 1

ConvN is all conv2d where N is a number from 1-6

Layer No.	Layer type	Kernel size	input/output dimensions	input/output Channels For conv layers
1	conv1	5	32|32	3|20
2	norm	-	32|32	-
3	relu	-	32|32	-
4	conv2	3	32|32	20|64
5	norm	-	32|32	-
6	relu	-	32|32	-
7	pool	-	32|16	-
8	conv3	5	16|16	64|128
9	norm	-	16|16	-
10	relu	-	16|16	-
11	conv4	3	16|16	128|256
12	norm	-	16|16	-
13	relu	-	16|16	-
14	pool	-	16|8	-
15	conv5	5	8|8	256|512
16	norm	-	8|8	-
17	relu	-	8|8	-
18	conv6	3	8|8	512|1024

19	norm	-	8|8	-
20	relu	-	8|8	-
21	pool	-	8|4	-
22	linear	-	16384|4096	-
23	norm	-	4096|4096	-
24	relu	-	4096|4096	-
25	linear	-	4096|2048	-
26	norm	-	2048|2048	-
27	relu	-	2048|2048	-
28	linear	-	2048|50	-
29	norm	-	50|50	-
30	relu	-	50|50	-
31	linear	-	50|100	-

Best performance: 0.655 as shown on Kaggle



Ablation study

The initial image augmentations were:
RandomCrop(30)
resize(32)
RandomRotation(22)
RandomHorizontalFip()
RandomVerticalFlip

First was adding a normalization layer, which improved accuracy from 25% to 42%
Adding deeper layers at first didn't do anything.
Then we increased channel size from 50 to 512 and epochs to 50, improving accuracy from 42 to 52%. finally, we added two more convolutional layers to get to 57%.

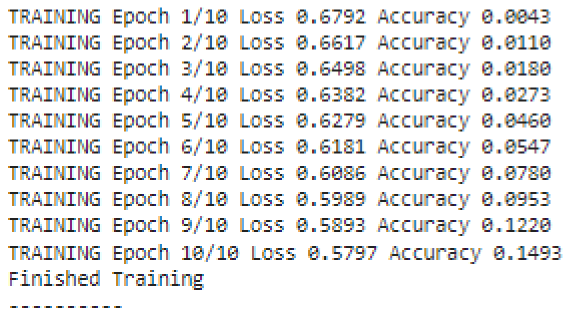
Change the cropping from 30 to 28; we increase from 57% to 58%.

Getting rid of random Rotations, Random HorizontalFip and ColourJitters increased by 63%, as shown in Kaggle. This also includes adjusting the learning rate to 0.005.

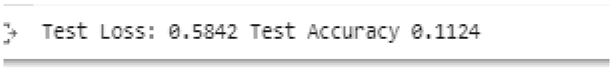
Then decreasing the learning rate of a factor of 10 at 40 epochs increased the percentage to 65%

Part 2

Original:



```
TRAINING Epoch 1/10 Loss 0.6792 Accuracy 0.0043
TRAINING Epoch 2/10 Loss 0.6617 Accuracy 0.0110
TRAINING Epoch 3/10 Loss 0.6498 Accuracy 0.0180
TRAINING Epoch 4/10 Loss 0.6382 Accuracy 0.0273
TRAINING Epoch 5/10 Loss 0.6279 Accuracy 0.0460
TRAINING Epoch 6/10 Loss 0.6181 Accuracy 0.0547
TRAINING Epoch 7/10 Loss 0.6086 Accuracy 0.0780
TRAINING Epoch 8/10 Loss 0.5989 Accuracy 0.0953
TRAINING Epoch 9/10 Loss 0.5893 Accuracy 0.1220
TRAINING Epoch 10/10 Loss 0.5797 Accuracy 0.1493
Finished Training
----------
```



```
Test Loss: 0.5842 Test Accuracy 0.1124
```

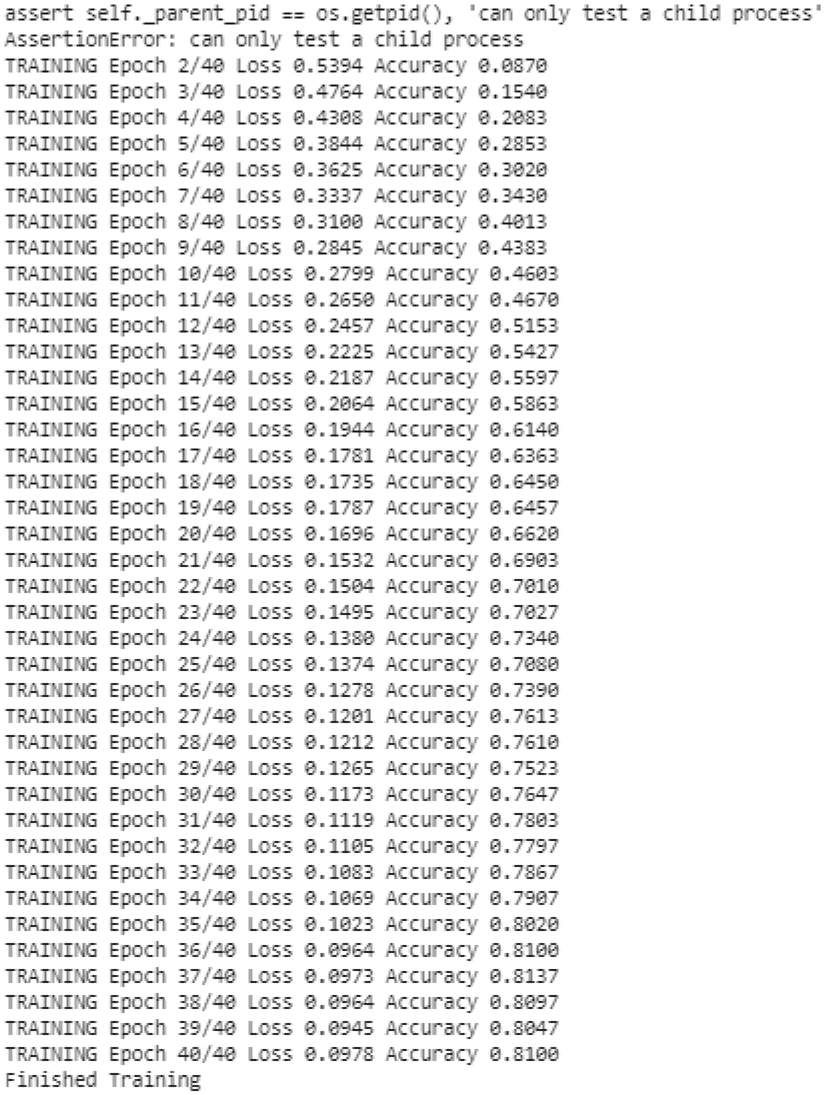
First fine-tune trial:

It seems like larger resizing sizes mean lower testing acracy

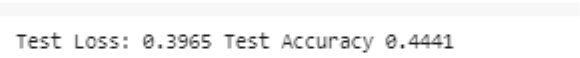
Other parameters:

Batch_size: 8
Learning_rate: 0.006
Resnet_last_only: False
Num_epochs:40

Image augmentations:
RandomResizedCrop to 230,



```
assert self._parent_pid == os.getpid(), 'can only test a child process'
AssertionError: can only test a child process
TRAINING Epoch 2/40 Loss 0.5394 Accuracy 0.0870
TRAINING Epoch 3/40 Loss 0.4764 Accuracy 0.1540
TRAINING Epoch 4/40 Loss 0.4308 Accuracy 0.2083
TRAINING Epoch 5/40 Loss 0.3844 Accuracy 0.2853
TRAINING Epoch 6/40 Loss 0.3625 Accuracy 0.3020
TRAINING Epoch 7/40 Loss 0.3337 Accuracy 0.3430
TRAINING Epoch 8/40 Loss 0.3100 Accuracy 0.4013
TRAINING Epoch 9/40 Loss 0.2845 Accuracy 0.4383
TRAINING Epoch 10/40 Loss 0.2799 Accuracy 0.4603
TRAINING Epoch 11/40 Loss 0.2650 Accuracy 0.4670
TRAINING Epoch 12/40 Loss 0.2457 Accuracy 0.5153
TRAINING Epoch 13/40 Loss 0.2225 Accuracy 0.5427
TRAINING Epoch 14/40 Loss 0.2187 Accuracy 0.5597
TRAINING Epoch 15/40 Loss 0.2064 Accuracy 0.5863
TRAINING Epoch 16/40 Loss 0.1944 Accuracy 0.6140
TRAINING Epoch 17/40 Loss 0.1781 Accuracy 0.6363
TRAINING Epoch 18/40 Loss 0.1735 Accuracy 0.6450
TRAINING Epoch 19/40 Loss 0.1787 Accuracy 0.6457
TRAINING Epoch 20/40 Loss 0.1696 Accuracy 0.6620
TRAINING Epoch 21/40 Loss 0.1532 Accuracy 0.6903
TRAINING Epoch 22/40 Loss 0.1504 Accuracy 0.7010
TRAINING Epoch 23/40 Loss 0.1495 Accuracy 0.7027
TRAINING Epoch 24/40 Loss 0.1380 Accuracy 0.7340
TRAINING Epoch 25/40 Loss 0.1374 Accuracy 0.7080
TRAINING Epoch 26/40 Loss 0.1278 Accuracy 0.7390
TRAINING Epoch 27/40 Loss 0.1201 Accuracy 0.7613
TRAINING Epoch 28/40 Loss 0.1212 Accuracy 0.7610
TRAINING Epoch 29/40 Loss 0.1265 Accuracy 0.7523
TRAINING Epoch 30/40 Loss 0.1173 Accuracy 0.7647
TRAINING Epoch 31/40 Loss 0.1119 Accuracy 0.7803
TRAINING Epoch 32/40 Loss 0.1105 Accuracy 0.7797
TRAINING Epoch 33/40 Loss 0.1083 Accuracy 0.7867
TRAINING Epoch 34/40 Loss 0.1069 Accuracy 0.7907
TRAINING Epoch 35/40 Loss 0.1023 Accuracy 0.8020
TRAINING Epoch 36/40 Loss 0.0964 Accuracy 0.8100
TRAINING Epoch 37/40 Loss 0.0973 Accuracy 0.8137
TRAINING Epoch 38/40 Loss 0.0964 Accuracy 0.8097
TRAINING Epoch 39/40 Loss 0.0945 Accuracy 0.8047
TRAINING Epoch 40/40 Loss 0.0978 Accuracy 0.8100
Finished Training
```



```
Test Loss: 0.3965 Test Accuracy 0.4441
```

Second fine-tune trial:

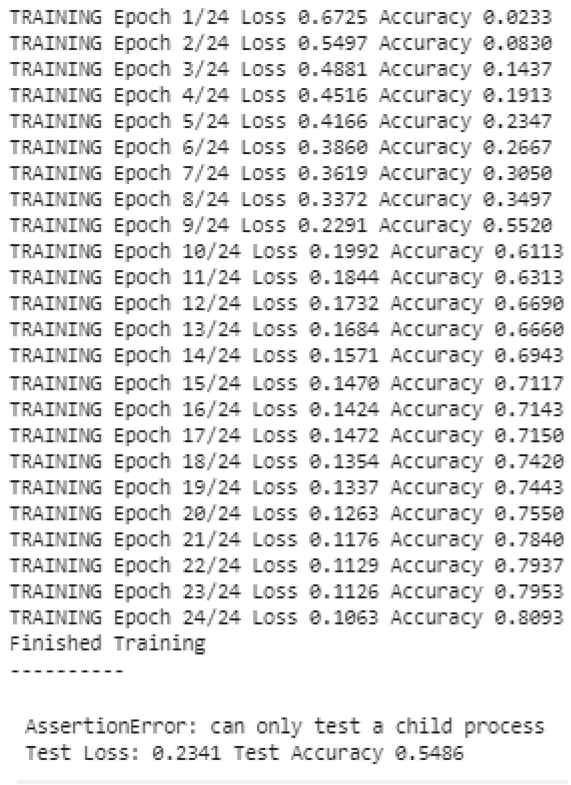
From the previous screenshot, I notice that the increase in accuracy significantly slows down around epoch 10. Try to reduce the lr by a factor of 5 at increments of 10, which results in a better epoch increase rate and a testing rate of around 55%. (the main idea is similar to drawing; a rough sketch at first then needs a more significant learning rate. Reducing the learning rate at increments is like filling in the details) source: https://www.jeremyjordan.me/nn-learning-rate/

Other parameters:

Batch_size: 8
Learning_rate: 0.006
Resnet_last_only: False
num_epochs:24

Image augmentations:

Center crop at 224
RandomResizedCrop: 224,
Random horizontal flip



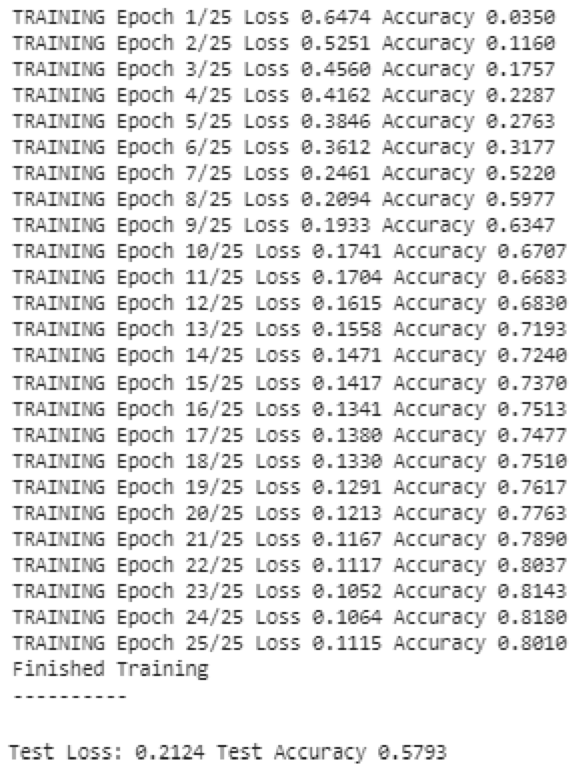
```
TRAINING Epoch 1/24 Loss 0.6725 Accuracy 0.0233
TRAINING Epoch 2/24 Loss 0.5497 Accuracy 0.0830
TRAINING Epoch 3/24 Loss 0.4881 Accuracy 0.1437
TRAINING Epoch 4/24 Loss 0.4516 Accuracy 0.1913
TRAINING Epoch 5/24 Loss 0.4166 Accuracy 0.2347
TRAINING Epoch 6/24 Loss 0.3860 Accuracy 0.2667
TRAINING Epoch 7/24 Loss 0.3619 Accuracy 0.3050
TRAINING Epoch 8/24 Loss 0.3372 Accuracy 0.3497
TRAINING Epoch 9/24 Loss 0.2291 Accuracy 0.5520
TRAINING Epoch 10/24 Loss 0.1992 Accuracy 0.6113
TRAINING Epoch 11/24 Loss 0.1844 Accuracy 0.6313
TRAINING Epoch 12/24 Loss 0.1732 Accuracy 0.6690
TRAINING Epoch 13/24 Loss 0.1684 Accuracy 0.6660
TRAINING Epoch 14/24 Loss 0.1571 Accuracy 0.6943
TRAINING Epoch 15/24 Loss 0.1470 Accuracy 0.7117
TRAINING Epoch 16/24 Loss 0.1424 Accuracy 0.7143
TRAINING Epoch 17/24 Loss 0.1472 Accuracy 0.7150
TRAINING Epoch 18/24 Loss 0.1354 Accuracy 0.7420
TRAINING Epoch 19/24 Loss 0.1337 Accuracy 0.7443
TRAINING Epoch 20/24 Loss 0.1263 Accuracy 0.7550
TRAINING Epoch 21/24 Loss 0.1176 Accuracy 0.7840
TRAINING Epoch 22/24 Loss 0.1129 Accuracy 0.7937
TRAINING Epoch 23/24 Loss 0.1126 Accuracy 0.7953
TRAINING Epoch 24/24 Loss 0.1063 Accuracy 0.8093
Finished Training
----------

 AssertionError: can only test a child process
 Test Loss: 0.2341 Test Accuracy 0.5486
```

Third fine-tune trial:

Reduce learning rate to 0.005 and changing increments at 6 and 20 instead of 10 and 20 results in the following:

Other parameters:
Batch_size: 8
Learning_rate: 0.005
Resnet_last_only: False
Num_epochs:25
Image augmentations:
RandomResizedCrop: 224,
Random horizontal flip



```
TRAINING Epoch 1/25 Loss 0.6474 Accuracy 0.0350
TRAINING Epoch 2/25 Loss 0.5251 Accuracy 0.1160
TRAINING Epoch 3/25 Loss 0.4560 Accuracy 0.1757
TRAINING Epoch 4/25 Loss 0.4162 Accuracy 0.2287
TRAINING Epoch 5/25 Loss 0.3846 Accuracy 0.2763
TRAINING Epoch 6/25 Loss 0.3612 Accuracy 0.3177
TRAINING Epoch 7/25 Loss 0.2461 Accuracy 0.5220
TRAINING Epoch 8/25 Loss 0.2094 Accuracy 0.5977
TRAINING Epoch 9/25 Loss 0.1933 Accuracy 0.6347
TRAINING Epoch 10/25 Loss 0.1741 Accuracy 0.6707
TRAINING Epoch 11/25 Loss 0.1704 Accuracy 0.6683
TRAINING Epoch 12/25 Loss 0.1615 Accuracy 0.6830
TRAINING Epoch 13/25 Loss 0.1558 Accuracy 0.7193
TRAINING Epoch 14/25 Loss 0.1471 Accuracy 0.7240
TRAINING Epoch 15/25 Loss 0.1417 Accuracy 0.7370
TRAINING Epoch 16/25 Loss 0.1341 Accuracy 0.7513
TRAINING Epoch 17/25 Loss 0.1380 Accuracy 0.7477
TRAINING Epoch 18/25 Loss 0.1330 Accuracy 0.7510
TRAINING Epoch 19/25 Loss 0.1291 Accuracy 0.7617
TRAINING Epoch 20/25 Loss 0.1213 Accuracy 0.7763
TRAINING Epoch 21/25 Loss 0.1167 Accuracy 0.7890
TRAINING Epoch 22/25 Loss 0.1117 Accuracy 0.8037
TRAINING Epoch 23/25 Loss 0.1052 Accuracy 0.8143
TRAINING Epoch 24/25 Loss 0.1064 Accuracy 0.8180
TRAINING Epoch 25/25 Loss 0.1115 Accuracy 0.8010
Finished Training
----------

Test Loss: 0.2124 Test Accuracy 0.5793
```

Note: adjusting when the first increments start seemed to have the most effect. I noticed that reducing the number to 6 had the best results.