CS 6301 CNN Project 2 - Networks

Yi-Hui (Lily) Lee
yi-hui.lee@utdallas.edu

Oct 25, 2020

Section 1 - Design

- The network structure

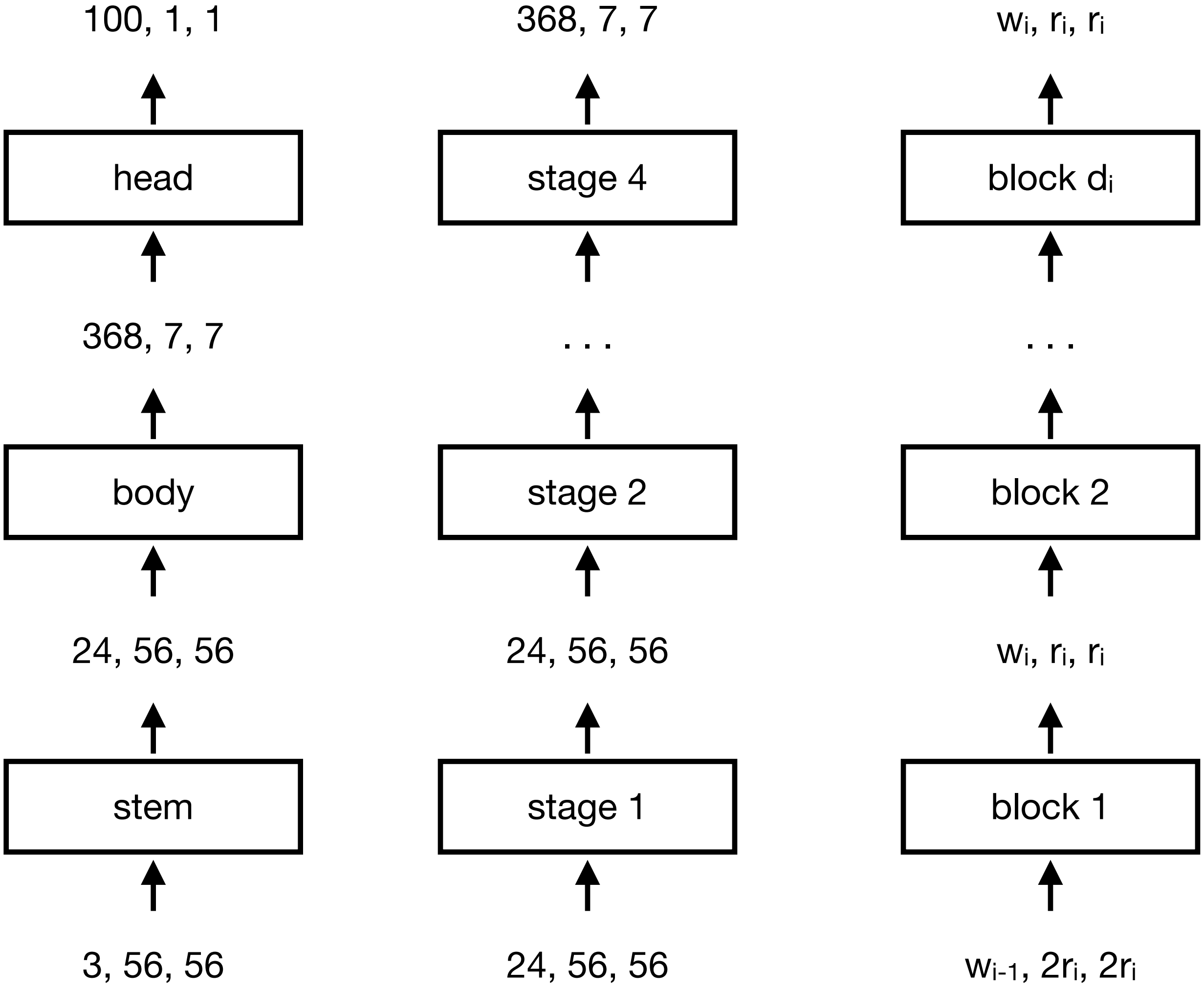


Figure 1: RegNet Network Structure

(a) network **(b) body** **(c) stage i**

- The network standard building block

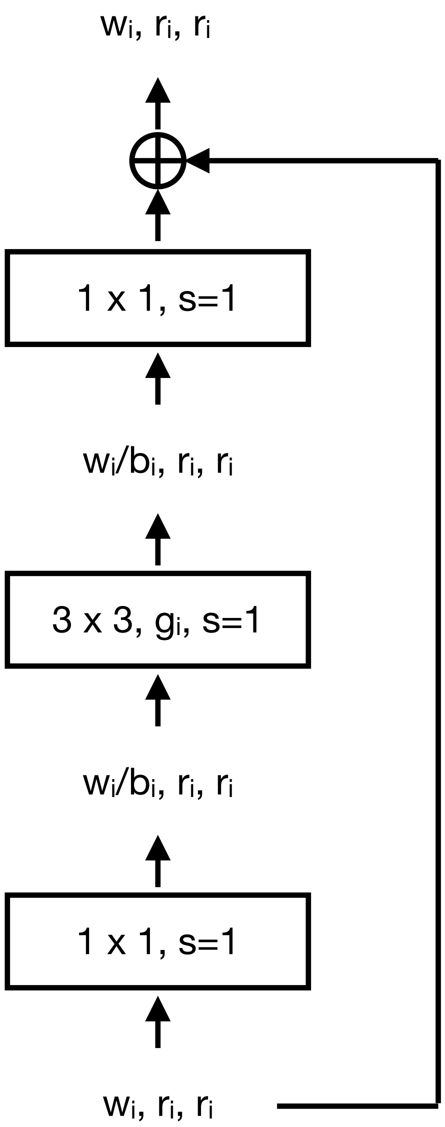


Figure 2: RegNet Standard Building Block

- The network down sampling building block

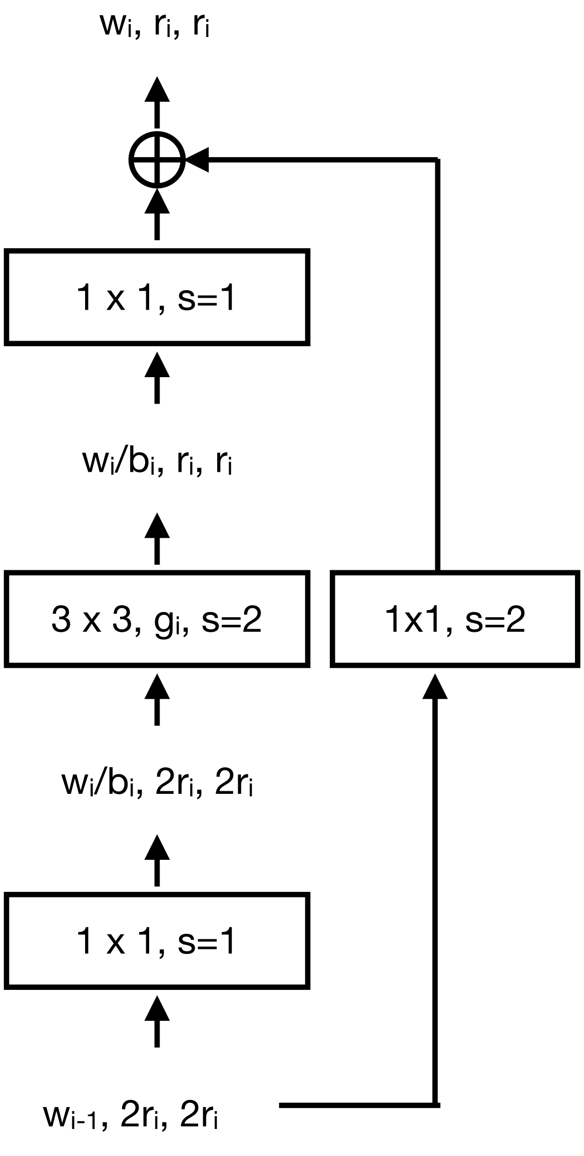


Figure 3: RegNet Down Sampling Building Block

- Table describes the parameters (eg., channels, groups, repeats, …)

Table 1: RegNet parameters

Layer name	Output Size	RegNetX-200MF
enc_stem	56 x 56	3 x 3, 24, stride 1
enc_1	56 x 56	$\left[\begin{array}{l}\text{24, 1 x 1, 24}\\\text{24, 3 x 3, 24, C=8}\\\text{24, 1 x 1, 24}\end{array}\right]\text{x1}$
enc_2	28 x 28	$\left[\begin{array}{l}\text{24, 1 x 1, 56}\\\text{56, 3 x 3, 56, C=8}\\\text{56, 1 x 1, 56}\end{array}\right]\text{x1}$
enc_3	14 x 14	$\left[\begin{array}{l}\text{56, 1 x 1, 152}\\\text{152, 3 x 3, 152, C=8}\\\text{152, 1 x 1, 152}\end{array}\right]\text{x1}$ $\left[\begin{array}{l}\text{152, 1 x 1, 152}\\\text{152, 3 x 3, 152, C=8}\\\text{152, 1 x 1, 152}\end{array}\right]\text{x 3}$
enc_4	7 x 7	$\left[\begin{array}{l}\text{152, 1 x 1, 368}\\\text{368, 3 x 3, 368, C=8}\\\text{368, 1 x 1, 368}\end{array}\right]\text{x 1}$ $\left[\begin{array}{l}\text{368, 1 x 1, 368}\\\text{368, 3 x 3, 368, C=8}\\\text{368, 1 x 1, 368}\end{array}\right]\text{x 6}$
head	1 x 1	global average pool 100-d fc, softmax

Section 2 - Training

- Table that have all hyper parameters and associated values

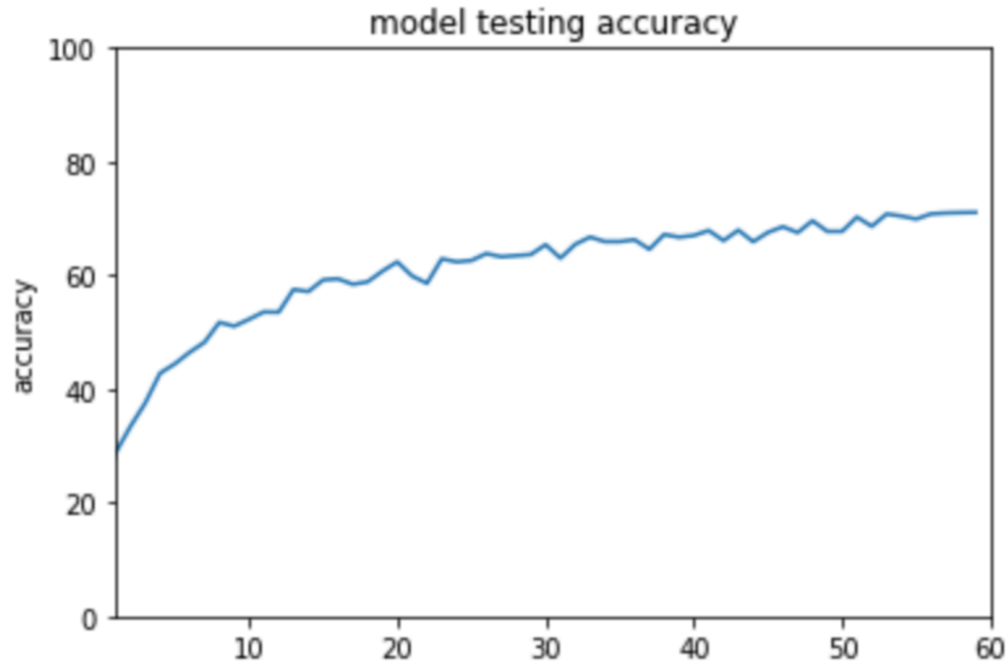
Table 2: RegNet hyper parameters
(Right: detailed learning rate(linear warmup followed by cosine decay))

epoch	60
batch size	512
data crop (train and test)	56
stage 1 blocks	1
stage 2 blocks	1
stage 3 blocks	4
stage 4 blocks	7
stem output channels	24
stage 1 channels	24
stage 2 channels	56
stage 3 channels	152
stage 4 channels	368
bottleneck ratio	1
groups	8
learning rate max	0.1
learning initial scale	0.5
learning initial epochs	5
learning final scale	0.01
learning initial epochs	55
Initial learning rate	0.05
final learning rate	0.001

Epoch	**Learning rate**
1	0.05
2	0.06
3	0.07
4	0.08
5	0.09
6	0.1
7	0.099958
8	0.099833
9	0.099623
10	0.099331
11…50	0.098955 ,…, 0.029394
51	0.026623
52	0.023831
53	0.021020
54	0.018191
55	0.015348
56	0.012493
57	0.009628
58	0.006756
59	0.003879
60	0.001000

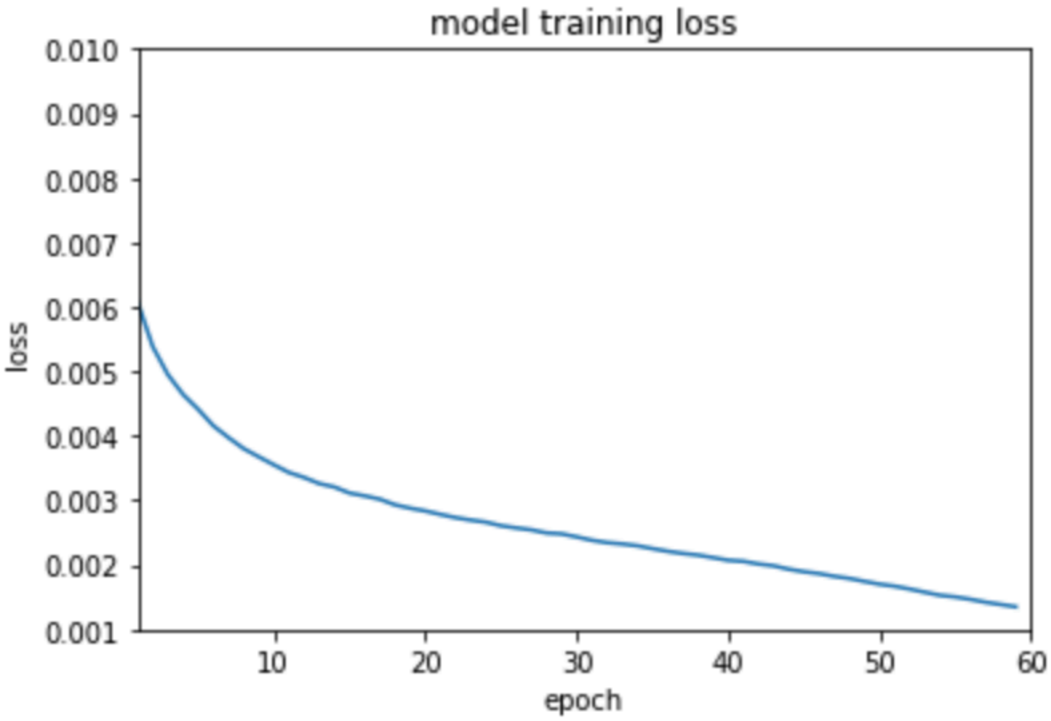
- Training data loss vs epoch

Figure 4: RegNetX-200MF Training Data Loss per Epoch



- Testing data accuracy vs epoch

Figure 5: RegNetX-200MF Testing Data Accuracy per Epoch



- Final accuracy = 71.68

Section 3 - Implementation

- Table include a list of each CNN style 2D convolution operator and associated the number of MACs and parameters; at the bottom of the table put a sum for the whole network

Table 3: RegNetX-200MF Each CNN style 2D Convolution Operator and Associated MACs and Parameters

#	Type	MACs	Parameters
Stem	Conv	2030.4	648
	Norm	150.4	48
Body, Stage1, Block 1	Conv	1804.8	576
	Norm	150.4	48
	Conv	2030.4	648
	Norm	150.4	48
	Conv	1804.8	576
	Norm	1052.8	48
Body, Stage 2, Block 1	Conv	150.4	1,344
	Norm	4211.2	112
	Conv	350.15	3,528
	Norm	2763.6	112
	Conv	86.95	3,136
(Residual)	Conv	2455.75	1,344
	Norm	1666.15	112
Body, Stage 3, Block 1	Conv	86.95	8,512

#	Type	MACs	Parameters
	Norm	6666.95	304
	Conv	237.35	25,992
	Norm	5090.1	304
	Conv	58.75	23,104
Residual	Conv	4523.75	8,512
	Norm	58.75	304
Body, Stage 3, Block 2	Conv	4523.75	23,104
	Norm	58.75	304
	Conv	5090.1	25,992
	Norm	58.75	304
	Conv	4523.75	23,104
	Norm	58.75	304
Body, Stage 3, Block 3	Conv	4523.75	23,104
	Norm	58.75	304
	Conv	5090.1	25,992
	Norm	58.75	304
	Conv	4523.75	23,104
	Norm	58.75	304
Body, Stage 3, Block 4	Conv	4523.75	23,104
	Norm	58.75	304
	Conv	5090.1	25,992
	Norm	58.75	304
	Conv	4523.75	23,104
	Norm	2737.75	304
Body, Stage 4, Block 1	Conv	58.75	55,936
	Norm	10953.35	736
	Conv	143.35	152,352

#	Type	MACs	Parameters
	Norm	7458.9	736
	Conv	35.25	135,424
(Residual)	Conv	6629.35	55,936
	Norm	35.25	736
Body, Stage 4, Block 2	Conv	6629.35	135,424
	Norm	35.25	736
	Conv	7458.9	152,352
	Norm	35.25	736
	Conv	6629.35	135,424
	Norm	35.25	736
Body, Stage 4, Block 3	Conv	6629.35	135,424
	Norm	35.25	736
	Conv	7458.9	152,352
	Norm	35.25	736
	Conv	6629.35	135,424
	Norm	35.25	736
Body, Stage 4, Block 4	Conv	6629.35	135,424
	Norm	35.25	736
	Conv	7458.9	152,352
	Norm	35.25	736
	Conv	6629.35	135,424
	Norm	35.25	736
Body, Stage 4, Block 5	Conv	6629.35	135,424
	Norm	35.25	736
	Conv	7458.9	152,352
	Norm	35.25	736
	Conv	6629.35	135,424

#	Type	MACs	Parameters
	Norm	35.25	736
Body, Stage 4, Block 6	Conv	6629.35	135,424
	Norm	35.25	736
	Conv	7458.9	152,352
	Norm	35.25	736
	Conv	6629.35	135,424
	Norm	35.25	736
Body, Stage 4, Block 7	Conv	6629.35	135,424
	Norm	35.25	736
	Conv	7458.9	152,352
	Norm	35.25	736
	Conv	6629.35	135,424
	Norm	35.25	736
Head	Full	37.6	36,900
Sum		**0.235 G (235,000)**	**3,289,900**

Section 4 - Extra

• Extra1: More epochs

I train on more epochs. And the results in Table 4 and Figure 6 show that the model is getting over fitting. Since my training loss is keep going down but my testing accuracy is stable.

Table 4: Train on More Epochs' Training Loss and Testing Accuracy

epochs	Training loss	Testing accuracy
60	0.001352	71.68
100	0.000965	71.14
150	0.000805	70.81

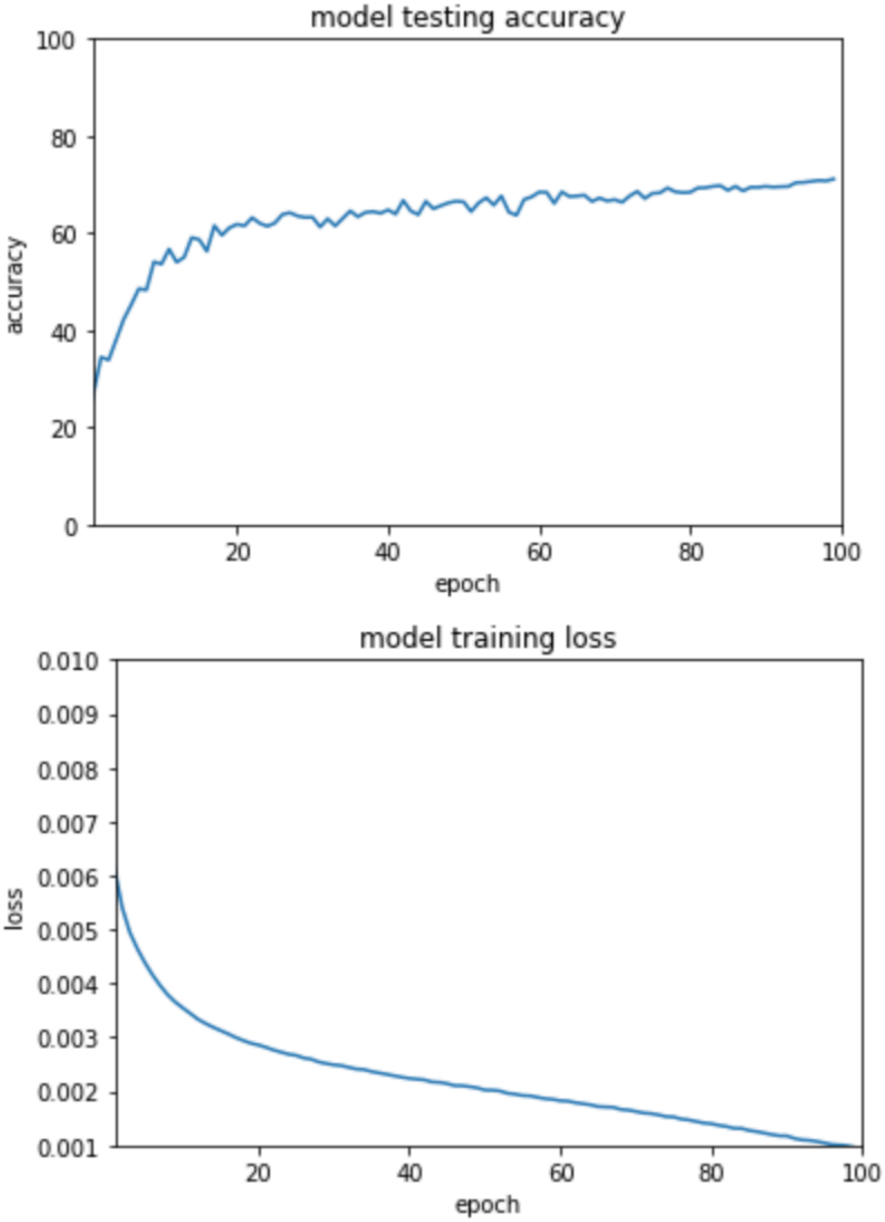


Figure 6: RegNetX-200MF Testing Accuracy and Training Data Loss per Epoch (Total Epochs = 100)

- Extra 2: Larger test crops result in better accuracy

Reading the paper "Fixing the train-test resolution discrepancy" (https://arxiv.org/pdf/1906.06423.pdf). On sec3.3 they point out that larger test crops result in better accuracy. We believe it's worth to try this idea on our RegNetX-200MF version.
The following is the table of the testing accuracy through the different test crops from 40 to 91.

Table 5: Testing Accuracy through Test Crops from 40 to 91.

Test crops	Testing accuracy
40	62.78
41	57.53
42	62.61
43	63.41
44	65.62
45	62.46
46	66.23
47	65.95
48	67.73
49	64.43
50	66.91
51	67.51
52	69.29
53	67.56
54	69.88
55	70.55
56	71.68
57	67.64
58	70.25
59	69.77
60	71.22
61	69.68
62	71.88
63	72.44
64	72.85
65	68.14

Test crops	Testing accuracy
66	68.19
67	67.80
68	69.16
69	68.10
70	68.47
71	68.95
72	70.31
73	64.11
74	66.34
75	65.28
76	66.88
77	64.82
78	66.19
79	66.84
80	67.14
81	61.50
82	63.30
83	64.08
84	65.34
85	63.09
86	63.30
87	63.11
88	63.80
89	57.34
90	54.77
91	56.92

When test crops = 56 which is the same crop size as train crops. Has 71.68 testing accuracy.

When test crops = 62 is the accuracy is 71.88 > **71.68.**
When test crops = 63 is the accuracy is 72.44 > **71.68.**
When test crops = 64 is the accuracy is **72.58** > **71.68.** And also is the best result.

Some how our results show's that larger test crops do result in better accuracy.

But when test crop is larger than 89 the testing accuracy becomes very low that the testing accuracy even smaller than 60%.

• Extra 3: Group size and bottleneck ratio.

We reproduce the RegNet 200MF based on the parameters on the paper. But in the paper every group size is 8 and the bottleneck ratio is set to 1.

Based on the idea of lower the parameters and time of the model but still achieve the good testing accuracy. I designed the experiment that set the group size to 4, 2, 1 and in the mean while set the bottleneck ratio to 2, 4, 8.

The idea of designing this experiment is to find out will lower the parameters influence the quality of the model or not.

Table 6: Testing Accuracy Based on Smaller Parameters and Training Time

Group size	Bottleneck ratio	Parameters	Accuracy	Training Time per epoch
8	1	3,289,900	71.68	171.09
4	2	1,699,908	70.03	114.47
2	4	904,912	68.47	82.26
1	8	507,414	64.52	73.93

Based on the results in Table 6. We can find out that lower the parameters didn't influence the quality of the model and even save more time on training process. This give me an idea to design the model in small group size with high bottleneck ratio but more complicated design on different aspect. Hopefully, after design the more complicate model but in small group size with high bottleneck ratio, the model's parameters and training time will stay as the same size as RegNet but works even better. In the future, we may try to design different group size on each four different stage. Or even think about concat the the different stage output like GoogLeNet.