

## Classifying Images with Convolutional Neural Networks (CNNs)

Team members - 200501P  
- 200677H

Q1.

Model	Network Architecture	Results																																																																								
1	<table> <tr> <th>Layer (type)</th><th>Output Shape</th><th>Param #</th></tr> <tr><td>Conv2d-1</td><td>[32, 64, 198, 198]</td><td>4,864</td></tr> <tr><td>BatchNorm2d-2</td><td>[32, 64, 198, 198]</td><td>128</td></tr> <tr><td>ReLU-3</td><td>[32, 64, 198, 198]</td><td>0</td></tr> <tr><td>MaxPool2d-4</td><td>[32, 64, 99, 99]</td><td>0</td></tr> <tr><td>Conv2d-5</td><td>[32, 128, 97, 97]</td><td>73,856</td></tr> <tr><td>BatchNorm2d-6</td><td>[32, 128, 97, 97]</td><td>256</td></tr> <tr><td>ReLU-7</td><td>[32, 128, 97, 97]</td><td>0</td></tr> <tr><td>MaxPool2d-8</td><td>[32, 128, 48, 48]</td><td>0</td></tr> <tr><td>Conv2d-9</td><td>[32, 256, 44, 44]</td><td>819,456</td></tr> <tr><td>BatchNorm2d-10</td><td>[32, 256, 44, 44]</td><td>512</td></tr> <tr><td>ReLU-11</td><td>[32, 256, 44, 44]</td><td>0</td></tr> <tr><td>MaxPool2d-12</td><td>[32, 256, 22, 22]</td><td>0</td></tr> <tr><td>Conv2d-13</td><td>[32, 500, 20, 20]</td><td>1,152,500</td></tr> <tr><td>BatchNorm2d-14</td><td>[32, 500, 20, 20]</td><td>1,000</td></tr> <tr><td>ReLU-15</td><td>[32, 500, 20, 20]</td><td>0</td></tr> <tr><td>MaxPool2d-16</td><td>[32, 500, 10, 10]</td><td>0</td></tr> <tr><td>Flatten-17</td><td>[32, 50000]</td><td>0</td></tr> <tr><td>Dropout-18</td><td>[32, 50000]</td><td>0</td></tr> <tr><td>Linear-19</td><td>[32, 1024]</td><td>51,201,024</td></tr> <tr><td>ReLU-20</td><td>[32, 1024]</td><td>0</td></tr> <tr><td>Dropout-21</td><td>[32, 1024]</td><td>0</td></tr> <tr><td>Linear-22</td><td>[32, 256]</td><td>262,400</td></tr> <tr><td>...</td><td></td><td></td></tr> </table> <p>Forward/backward pass size (MB): 3522.18 Params size (MB): 204.21 Estimated Total Size (MB): 3741.04</p>	Layer (type)	Output Shape	Param #	Conv2d-1	[32, 64, 198, 198]	4,864	BatchNorm2d-2	[32, 64, 198, 198]	128	ReLU-3	[32, 64, 198, 198]	0	MaxPool2d-4	[32, 64, 99, 99]	0	Conv2d-5	[32, 128, 97, 97]	73,856	BatchNorm2d-6	[32, 128, 97, 97]	256	ReLU-7	[32, 128, 97, 97]	0	MaxPool2d-8	[32, 128, 48, 48]	0	Conv2d-9	[32, 256, 44, 44]	819,456	BatchNorm2d-10	[32, 256, 44, 44]	512	ReLU-11	[32, 256, 44, 44]	0	MaxPool2d-12	[32, 256, 22, 22]	0	Conv2d-13	[32, 500, 20, 20]	1,152,500	BatchNorm2d-14	[32, 500, 20, 20]	1,000	ReLU-15	[32, 500, 20, 20]	0	MaxPool2d-16	[32, 500, 10, 10]	0	Flatten-17	[32, 50000]	0	Dropout-18	[32, 50000]	0	Linear-19	[32, 1024]	51,201,024	ReLU-20	[32, 1024]	0	Dropout-21	[32, 1024]	0	Linear-22	[32, 256]	262,400	...			<p>Test Accuracy: tensor(0.7267) Test Precision: tensor(0.7083) Test Recall: tensor(0.8395) Test F1: tensor(0.7684)</p>
Layer (type)	Output Shape	Param #																																																																								
Conv2d-1	[32, 64, 198, 198]	4,864																																																																								
BatchNorm2d-2	[32, 64, 198, 198]	128																																																																								
ReLU-3	[32, 64, 198, 198]	0																																																																								
MaxPool2d-4	[32, 64, 99, 99]	0																																																																								
Conv2d-5	[32, 128, 97, 97]	73,856																																																																								
BatchNorm2d-6	[32, 128, 97, 97]	256																																																																								
ReLU-7	[32, 128, 97, 97]	0																																																																								
MaxPool2d-8	[32, 128, 48, 48]	0																																																																								
Conv2d-9	[32, 256, 44, 44]	819,456																																																																								
BatchNorm2d-10	[32, 256, 44, 44]	512																																																																								
ReLU-11	[32, 256, 44, 44]	0																																																																								
MaxPool2d-12	[32, 256, 22, 22]	0																																																																								
Conv2d-13	[32, 500, 20, 20]	1,152,500																																																																								
BatchNorm2d-14	[32, 500, 20, 20]	1,000																																																																								
ReLU-15	[32, 500, 20, 20]	0																																																																								
MaxPool2d-16	[32, 500, 10, 10]	0																																																																								
Flatten-17	[32, 50000]	0																																																																								
Dropout-18	[32, 50000]	0																																																																								
Linear-19	[32, 1024]	51,201,024																																																																								
ReLU-20	[32, 1024]	0																																																																								
Dropout-21	[32, 1024]	0																																																																								
Linear-22	[32, 256]	262,400																																																																								
...																																																																										
2	<table> <tr> <th>Layer (type)</th><th>Output Shape</th><th>Param #</th></tr> <tr><td>Conv2d-1</td><td>[32, 128, 198, 198]</td><td>3,584</td></tr> <tr><td>BatchNorm2d-2</td><td>[32, 128, 198, 198]</td><td>256</td></tr> <tr><td>ReLU-3</td><td>[32, 128, 198, 198]</td><td>0</td></tr> <tr><td>MaxPool2d-4</td><td>[32, 128, 99, 99]</td><td>0</td></tr> <tr><td>Conv2d-5</td><td>[32, 256, 97, 97]</td><td>295,168</td></tr> <tr><td>BatchNorm2d-6</td><td>[32, 256, 97, 97]</td><td>512</td></tr> <tr><td>ReLU-7</td><td>[32, 256, 97, 97]</td><td>0</td></tr> <tr><td>MaxPool2d-8</td><td>[32, 256, 48, 48]</td><td>0</td></tr> <tr><td>Conv2d-9</td><td>[32, 512, 46, 46]</td><td>1,180,160</td></tr> <tr><td>BatchNorm2d-10</td><td>[32, 512, 46, 46]</td><td>1,024</td></tr> <tr><td>ReLU-11</td><td>[32, 512, 46, 46]</td><td>0</td></tr> <tr><td>MaxPool2d-12</td><td>[32, 512, 23, 23]</td><td>0</td></tr> <tr><td>Conv2d-13</td><td>[32, 1024, 21, 21]</td><td>4,719,616</td></tr> <tr><td>BatchNorm2d-14</td><td>[32, 1024, 21, 21]</td><td>2,048</td></tr> <tr><td>ReLU-15</td><td>[32, 1024, 21, 21]</td><td>0</td></tr> <tr><td>MaxPool2d-16</td><td>[32, 1024, 10, 10]</td><td>0</td></tr> <tr><td>Conv2d-17</td><td>[32, 2048, 8, 8]</td><td>18,876,416</td></tr> <tr><td>BatchNorm2d-18</td><td>[32, 2048, 8, 8]</td><td>4,096</td></tr> <tr><td>ReLU-19</td><td>[32, 2048, 8, 8]</td><td>0</td></tr> <tr><td>MaxPool2d-20</td><td>[32, 2048, 4, 4]</td><td>0</td></tr> <tr><td>Flatten-21</td><td>[32, 32768]</td><td>0</td></tr> <tr><td>Dropout-22</td><td>[32, 32768]</td><td>0</td></tr> <tr><td>...</td><td></td><td></td></tr> </table> <p>Forward/backward pass size (MB): 7227.03 Params size (MB): 355.94 Estimated Total Size (MB): 7597.62</p>	Layer (type)	Output Shape	Param #	Conv2d-1	[32, 128, 198, 198]	3,584	BatchNorm2d-2	[32, 128, 198, 198]	256	ReLU-3	[32, 128, 198, 198]	0	MaxPool2d-4	[32, 128, 99, 99]	0	Conv2d-5	[32, 256, 97, 97]	295,168	BatchNorm2d-6	[32, 256, 97, 97]	512	ReLU-7	[32, 256, 97, 97]	0	MaxPool2d-8	[32, 256, 48, 48]	0	Conv2d-9	[32, 512, 46, 46]	1,180,160	BatchNorm2d-10	[32, 512, 46, 46]	1,024	ReLU-11	[32, 512, 46, 46]	0	MaxPool2d-12	[32, 512, 23, 23]	0	Conv2d-13	[32, 1024, 21, 21]	4,719,616	BatchNorm2d-14	[32, 1024, 21, 21]	2,048	ReLU-15	[32, 1024, 21, 21]	0	MaxPool2d-16	[32, 1024, 10, 10]	0	Conv2d-17	[32, 2048, 8, 8]	18,876,416	BatchNorm2d-18	[32, 2048, 8, 8]	4,096	ReLU-19	[32, 2048, 8, 8]	0	MaxPool2d-20	[32, 2048, 4, 4]	0	Flatten-21	[32, 32768]	0	Dropout-22	[32, 32768]	0	...			
Layer (type)	Output Shape	Param #																																																																								
Conv2d-1	[32, 128, 198, 198]	3,584																																																																								
BatchNorm2d-2	[32, 128, 198, 198]	256																																																																								
ReLU-3	[32, 128, 198, 198]	0																																																																								
MaxPool2d-4	[32, 128, 99, 99]	0																																																																								
Conv2d-5	[32, 256, 97, 97]	295,168																																																																								
BatchNorm2d-6	[32, 256, 97, 97]	512																																																																								
ReLU-7	[32, 256, 97, 97]	0																																																																								
MaxPool2d-8	[32, 256, 48, 48]	0																																																																								
Conv2d-9	[32, 512, 46, 46]	1,180,160																																																																								
BatchNorm2d-10	[32, 512, 46, 46]	1,024																																																																								
ReLU-11	[32, 512, 46, 46]	0																																																																								
MaxPool2d-12	[32, 512, 23, 23]	0																																																																								
Conv2d-13	[32, 1024, 21, 21]	4,719,616																																																																								
BatchNorm2d-14	[32, 1024, 21, 21]	2,048																																																																								
ReLU-15	[32, 1024, 21, 21]	0																																																																								
MaxPool2d-16	[32, 1024, 10, 10]	0																																																																								
Conv2d-17	[32, 2048, 8, 8]	18,876,416																																																																								
BatchNorm2d-18	[32, 2048, 8, 8]	4,096																																																																								
ReLU-19	[32, 2048, 8, 8]	0																																																																								
MaxPool2d-20	[32, 2048, 4, 4]	0																																																																								
Flatten-21	[32, 32768]	0																																																																								
Dropout-22	[32, 32768]	0																																																																								
...																																																																										

3	Layer (type)	Output Shape	Param #
	Conv2d-1	[200, 128, 198, 198]	9,728
	BatchNorm2d-2	[200, 128, 198, 198]	256
	ReLU-3	[200, 128, 198, 198]	0
	MaxPool2d-4	[200, 128, 99, 99]	0
	Conv2d-5	[200, 256, 97, 97]	295,168
	BatchNorm2d-6	[200, 256, 97, 97]	512
	ReLU-7	[200, 256, 97, 97]	0
	MaxPool2d-8	[200, 256, 48, 48]	0
	Conv2d-9	[200, 512, 44, 44]	3,277,312
	BatchNorm2d-10	[200, 512, 44, 44]	1,024
	ReLU-11	[200, 512, 44, 44]	0
	MaxPool2d-12	[200, 512, 22, 22]	0
	Conv2d-13	[200, 1024, 20, 20]	4,719,616
	BatchNorm2d-14	[200, 1024, 20, 20]	2,048
	ReLU-15	[200, 1024, 20, 20]	0
	MaxPool2d-16	[200, 1024, 10, 10]	0
	Conv2d-17	[200, 2048, 8, 8]	52,430,848
	BatchNorm2d-18	[200, 2048, 8, 8]	4,096
	ReLU-19	[200, 2048, 8, 8]	0
	MaxPool2d-20	[200, 2048, 4, 4]	0
	Flatten-21	[200, 32768]	0
	Dropout-22	[200, 32768]	0
	...		
	Forward/backward pass size (MB): 11746.06		
	Params size (MB): 360.77		
	Estimated Total Size (MB): 12198.39		

## Q2. Transfer Learning

<pre> vgg19  VGG(   (features): Sequential(     (0): Conv2d(3, 64, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))     (1): ReLU(inplace=True)     (2): Conv2d(64, 64, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))     (3): ReLU(inplace=True)     (4): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)     (5): Conv2d(64, 128, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))     (6): ReLU(inplace=True)     (7): Conv2d(128, 128, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))     (8): ReLU(inplace=True)     (9): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)     (10): Conv2d(128, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))     (11): ReLU(inplace=True)     (12): Conv2d(256, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))     (13): ReLU(inplace=True)     (14): Conv2d(256, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))     (15): ReLU(inplace=True)     (16): Conv2d(256, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))     (17): ReLU(inplace=True)     (18): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)     (19): Conv2d(256, 512, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))     (20): ReLU(inplace=True)     (21): Conv2d(512, 512, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))     (22): ReLU(inplace=True)   )   ...   (4): ReLU(inplace=True)   (5): Dropout(p=0.5, inplace=False)   (6): Linear(in_features=4096, out_features=1000, bias=True) ) </pre>	<p>Test Accuracy: tensor(0.6867)</p> <p>Test Precision: tensor(0.8269)</p> <p>Test Recall: tensor(0.5309)</p> <p>Test F1: tensor(0.6466)</p>
--	--