Picture classification project-Team 02

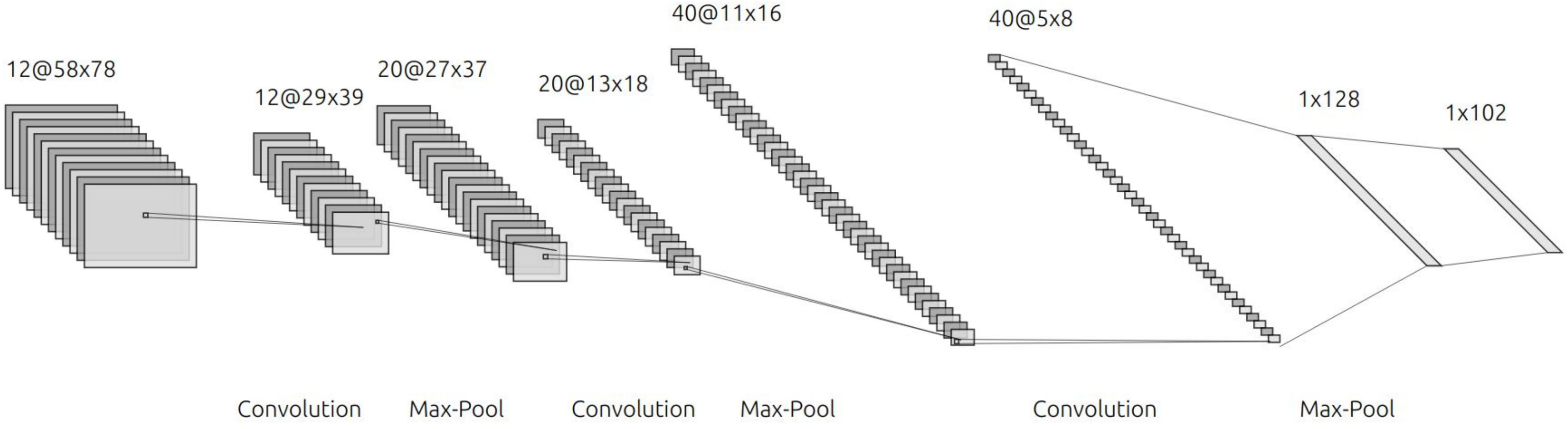
Members:

1)SRAVANI DHARA
2)DHEERAJ

Choice of classification technique: Convolutional Neural Networks

2.Neural Network

- The simple network increments convolutions with each layer followed by a max-pooling layer to control the number of trainable parameters.
- 30% Drop out after each convolution layer to control overfitting.
- Number of trainable parameters = 227,842



Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 58, 78, 12)	336
max_pooling2d (MaxPooling2D)	(None, 29, 39, 12)	0
dropout (Dropout)	(None, 29, 39, 12)	0
conv2d_1 (Conv2D)	(None, 27, 37, 20)	2180
max_pooling2d_1 (MaxPooling2	(None, 13, 18, 20)	0
dropout_1 (Dropout)	(None, 13, 18, 20)	0
conv2d_2 (Conv2D)	(None, 11, 16, 40)	7240
max_pooling2d_2 (MaxPooling2	(None, 5, 8, 40)	0
dropout_2 (Dropout)	(None, 5, 8, 40)	0
flatten (Flatten)	(None, 1600)	0
dense (Dense)	(None, 128)	204928
dense_1 (Dense)	(None, 102)	13158

Total params: 227,842
Trainable params: 227,842
Non-trainable params: 0

3.Parameters

3.1- Layers -

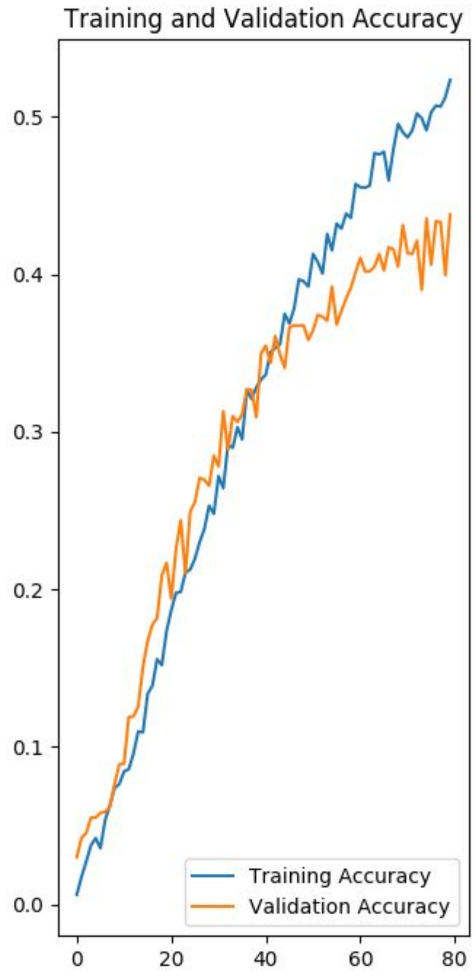
- **Network with too many convolution layers will require more processing time, for it has to train large number of parameters. Too few layers do not generate effective features.**
- **parameters like stride, kernel size, max pool kernel size etc will impact processing time and performance.**

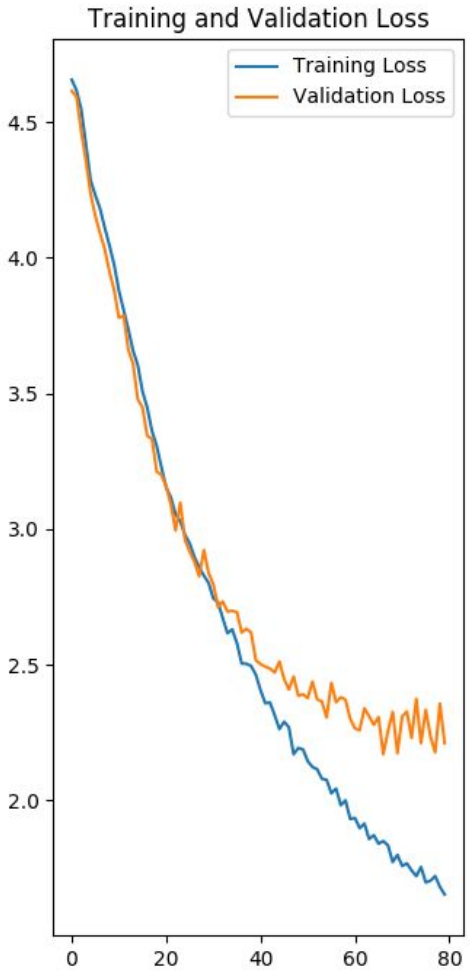
3.2 - Drop out - Being solution to over fitting,Drop out parameter is to be set precise enough to ensure that major learning of neurons is not lost. At the same time initial layers should have less drop out.

3.3 - Epoch - If the increment of epoch does not contribute to improvement of accuracy the epoch count should be reduced.This will save computation cost. We chose to run for 80 epochs based on accuracy and loss data of training and validation data sets.

Accuracy and Loss plots for Training and Validation datasets

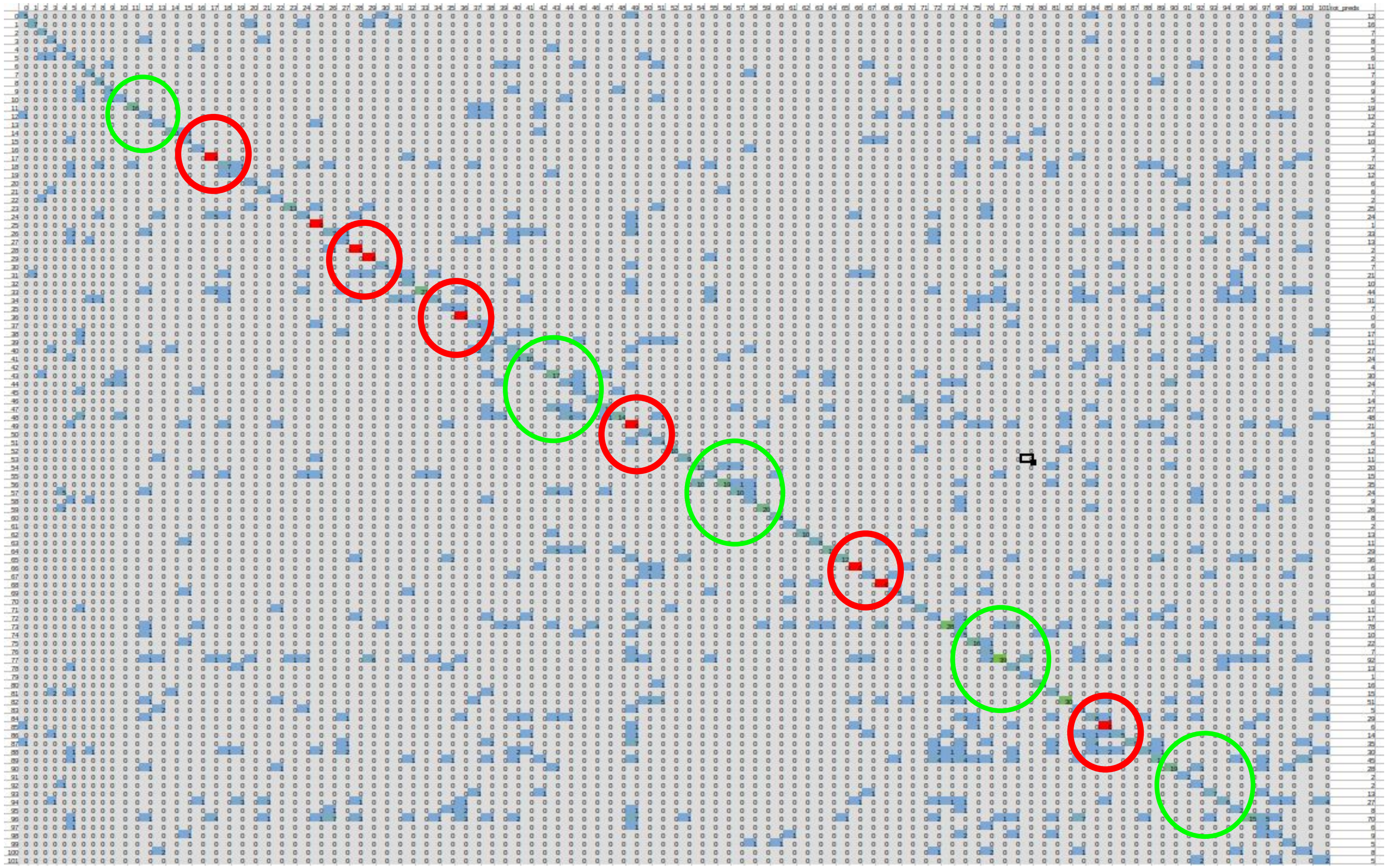
- Training:
  - Accuracy - 50.56%
  - Loss- 1.732
- Validation:
  - Accuracy - 41.3%
  - Loss- 2.18
- Testing:
  - Accuracy - 40%
  - Loss- 2.43





4.Results

4.1 Confusion Matrix:



- Areas highlighted in red circles show classes with zero ‘True Positive’ instances.
- Green highlighted classes had more than 10 ‘True Positives’ instances (correct - predictions)

4.2 Performance stats:

- Classification report from sklearn library shows the following
  - accuracy = 40.42%
  - Precision = 42.7%
  - Recall = 40.42%
  - F1 score = 37.7%

	precision	recall	f1-score		precision	recall	f1-score		precision	recall	f1-score
0	00.417	00.625	00.500	36	00.000	00.000	00.000	72	00.059	00.050	00.054
1	00.438	00.778	00.560	37	00.500	00.214	00.300	73	00.359	00.718	00.479
2	01.000	00.700	00.824	38	00.235	00.154	00.186	74	00.500	00.143	00.222
3	00.500	00.333	00.400	39	00.182	00.167	00.174	75	00.727	00.667	00.696
4	00.400	00.200	00.267	40	00.259	00.269	00.264	76	00.714	00.227	00.345
5	00.333	00.111	00.167	41	00.417	00.526	00.465	77	00.424	00.765	00.545
6	00.273	00.167	00.207	42	00.250	00.125	00.167	78	00.615	00.286	00.390
7	00.857	00.600	00.706	43	00.567	00.425	00.486	79	01.000	00.111	00.200
8	00.667	00.600	00.632	44	00.125	00.214	00.158	80	00.875	00.824	00.848
9	00.556	00.500	00.526	45	00.429	00.200	00.273	81	00.467	00.333	00.389
10	00.200	00.111	00.143	46	00.429	00.600	00.500	82	00.588	00.882	00.706
11	00.842	00.941	00.889	47	00.370	00.769	00.500	83	00.200	00.043	00.071
12	00.250	00.176	00.207	48	00.292	00.737	00.418	84	00.276	00.296	00.286
13	00.500	00.100	00.167	49	00.000	00.000	00.000	85	00.000	00.000	00.000
14	00.769	00.833	00.800	50	00.800	00.235	00.364	86	00.357	00.385	00.370
15	00.400	00.333	00.364	51	00.667	00.211	00.320	87	00.257	00.750	00.383
16	00.667	00.250	00.364	52	00.833	00.769	00.800	88	00.233	00.538	00.326
17	00.000	00.000	00.000	53	00.818	00.600	00.692	89	00.289	00.419	00.342
18	00.219	00.368	00.275	54	00.600	00.545	00.571	90	00.679	00.514	00.585
19	00.250	00.333	00.286	55	00.267	00.286	00.276	91	01.000	00.200	00.333
20	00.500	00.333	00.400	56	00.543	00.826	00.655	92	00.500	00.059	00.105
21	00.667	00.444	00.533	57	00.417	00.714	00.526	93	00.538	00.438	00.483
22	00.500	00.125	00.200	58	00.333	00.333	00.333	94	00.222	00.429	00.293
23	00.520	00.929	00.667	59	00.769	00.909	00.833	95	00.250	00.200	00.222
24	00.167	00.250	00.200	60	01.000	00.800	00.889	96	00.214	00.455	00.291
25	00.000	00.000	00.000	61	01.000	00.182	00.308	97	00.500	00.115	00.188
26	00.212	00.412	00.280	62	00.769	00.909	00.833	98	00.222	00.105	00.143
27	00.154	00.182	00.167	63	00.727	00.727	00.727	99	00.600	00.214	00.316
28	00.000	00.000	00.000	64	00.517	00.714	00.600	100	00.125	00.059	00.080
29	00.000	00.000	00.000	65	00.333	00.923	00.490	101	00.400	00.154	00.222
30	00.714	00.625	00.667	66	00.000	00.000	00.000				
31	00.143	00.333	00.200	67	00.231	00.273	00.250	accuracy	0.4042806	0.404281	0.40428
32	00.700	00.467	00.560	68	00.000	00.000	00.000	macro avg	0.4340789	0.390856	0.37236
33	00.477	00.955	00.636	69	00.500	00.625	00.556	weighted avg	0.4272337	0.404281	0.37717
34	00.194	00.500	00.279	70	00.500	00.231	00.316				
35	00.286	00.222	00.250	71	00.636	00.438	00.519				

Inferences from results:

- Many of the classes with least performance are the classes with least number of samples. The classes with zero performance stats were 18, 26, 28, 29, 36, 49, 66, 68, 85.
- Most of the wrong predictions are inexplicable. Number of samples, flower’s appearance, inter-class similarity etc, are the factors from dataset.
- The CNN architecture, convolutions, generated feature vectors etc can be equally responsible, which can all be improved.

class 18: dissimilar flowers

class 67: dissimilar flowers & smaller dataset

class 29 & 27: mispredictions due to similar features

5.Problem Faced

- **Processing time**: There are around 15 hyper parameters which should be tweaked when a desired result is not evident. While the network took around 1-2 hours ,at least, per configuration of these hyper parameters, this makes it important to trial with extreme caution.

- **overfitting**: There is a lot of room left for data augmentation to increase the samples, so that much deeper networks can learn without overfitting.

6.Self Assessment

6.1 QUALITY OF RESULTS:

1. More performance metrics could be extracted to better assess the problems and misclassifications.

6.2 METHODS APPLIED

- CNN has better performance in image classification.
- In CNN ,networks with numerous layers like VGG16 or Mobilenet have greater performance only when the sample size is high because these networks have high computation power and parameters.
- Hence a small neural network is good enough to classify the given set.

Questions?