ASSIGNMENT 4

Description of the four CNN architectures:

- **Architecture 1:** Image size of 150x150 is selected. Next the building of CNN is done with Rescaling layer as first layer, followed by Convolution layer with number of filters 32 and filter size 3x3, max pooling layer of number of filters 32 , convolution layer of filter number of filters 64, max pooling layer number of filters 64, convolution layer number of filters 128, max pooling layer number of filters 128, convolution layer number of filters 256 ,max pooling layer, then flattened, added dense layer of 128 filers, dropout filter and output layer. The summary of the model can be found below:

Layer (type)	Output Shape	Param #
rescaling (Rescaling)	(None, 150, 150, 3)	0
conv2d (Conv2D)	(None, 148, 148, 32)	896
max_pooling2d (MaxPooling2D)	(None, 74, 74, 32)	0
conv2d_1 (Conv2D)	(None, 72, 72, 64)	18496
max_pooling2d_1 (MaxPooling2D)	(None, 36, 36, 64)	0
conv2d_2 (Conv2D)	(None, 34, 34, 128)	73856
max_pooling2d_2 (MaxPooling2D)	(None, 17, 17, 128)	0
conv2d_3 (Conv2D)	(None, 15, 15, 256)	295168
max_pooling2d_3 (MaxPooling2D)	(None, 7, 7, 256)	0
flatten (Flatten)	(None, 12544)	0
dense (Dense)	(None, 128)	1605760
dropout (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 5)	645

- **Architecture 2:** Architecture 2 is same as Architecture 1 except for dropout layer. The dropout layer has been excluded in this architecture. The summary of the model can be found below:

Layer (type)	Output Shape	Param #
rescaling (Rescaling)	(None, 150, 150, 3)	0
conv2d (Conv2D)	(None, 148, 148, 32)	896
max_pooling2d (MaxPooling2D)	(None, 74, 74, 32)	0
conv2d_1 (Conv2D)	(None, 72, 72, 64)	18496
max_pooling2d_1 (MaxPooling2D)	(None, 36, 36, 64)	0
conv2d_2 (Conv2D)	(None, 34, 34, 128)	73856
max_pooling2d_2 (MaxPooling2D)	(None, 17, 17, 128)	0
conv2d_3 (Conv2D)	(None, 15, 15, 256)	295168
max_pooling2d_3 (MaxPooling2D)	(None, 7, 7, 256)	0
flatten (Flatten)	(None, 12544)	0
dense (Dense)	(None, 128)	1605760
dense_1 (Dense)	(None, 5)	645

- **Architecture 3:** In architecture 3, in contrast to architecture 2, I removed all the max-pooling layers and added the dropout layer. The convolution layers filters have been changed to 12 and 24. The summary of the model can be found below:

Layer (type)	Output Shape	Param #
rescaling (Rescaling)	(None, 150, 150, 3)	0
conv2d (Conv2D)	(None, 148, 148, 32)	896
conv2d_1 (Conv2D)	(None, 146, 146, 12)	3468
conv2d_2 (Conv2D)	(None, 144, 144, 24)	2616
flatten (Flatten)	(None, 497664)	0
dense (Dense)	(None, 128)	63701120
dropout (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 5)	645

- **Architecture 4:** Architecture 4 is a bit similar to architecture 1 but various convolution layers of filters 16,32,64 and 128 filters are added along with max-pool layers and additional dropout layer has been added as per the below image. The summary of this architecture can be found below.

Layer (type)	Output Shape	Param #
rescaling (Rescaling)	(None, 150, 150, 3)	0
conv2d (Conv2D)	(None, 148, 148, 32)	896
max_pooling2d (MaxPooling2D)	(None, 74, 74, 32)	0
conv2d_1 (Conv2D)	(None, 72, 72, 16)	4624
max_pooling2d_1 (MaxPooling2D)	(None, 36, 36, 16)	0
conv2d_2 (Conv2D)	(None, 34, 34, 32)	4640
max_pooling2d_2 (MaxPooling2D)	(None, 17, 17, 32)	0
conv2d_3 (Conv2D)	(None, 15, 15, 64)	18496
max_pooling2d_3 (MaxPooling2D)	(None, 7, 7, 64)	0
conv2d_4 (Conv2D)	(None, 5, 5, 128)	73856
max_pooling2d_4 (MaxPooling2D)	(None, 2, 2, 128)	0
max_pooling2d_5 (MaxPooling2D)	(None, 1, 1, 128)	0
flatten (Flatten)	(None, 128)	0
dense (Dense)	(None, 128)	16512
dropout (Dropout)	(None, 128)	0
dropout_1 (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 5)	645

- In all the above architectures, the image size is 150x150, activation function is relu for all the layers except for output layer which is softmax.

Results:

CNN Architecture	Accuracy
First Architecture	0.7211528680801392
Second Architecture	0.7137891054153442
Third Architecture	0.5469292998313904
Fourth Architecture	**0.7311703562736511**

The above table shows that the fourth architecture has the highest accuracy of 73.11%, i.e. the architecture with several convolution layers, max-pool layers, and two drop-out layers. In comparison, the third architecture has the least accuracy of 54.69%, i.e., the architecture with no max-pooling layers. The first and second architectures performed almost similar and nearly identical to the fourth architecture. Also, the third architecture is computationally demanding and comparatively took a lot of time than the rest of the architectures.

Predictions of fourth architecture on my dataset:

Image 1:

Predictions of the model: [[9.9578637e-01 8.9159532e-04 2.0221502e-03 9.3594589e-04 3.6393324e-04]]

It is an image of daisy. My model has predicted it as 99.5% daisy. It has predicted accurately.

Image 2:

Predictions of the model: [[9.9990118e-01 9.7702534e-05 8.6441673e-07 2.6122689e-07 1.0078970e-08]]

It is an image of daisy. My model has predicted it as 99.99% daisy. It has predicted accurately though it looks similar to sunflower.

Image 3:

Predictions of the model: [[0.21502729 0.12040379 0.34984863 0.06049548 0.25422484]]

It is an image of dandelion. But my model has predicted it as 12% dandelion and 34.9% rose. It has predicted inaccurately.

Image 4:

Predictions of the model: [[1.6138285e-04 9.9749434e-01 2.6618052e-05 2.2803852e-03 3.7324036e-05]]

It is an image of dandelion. My model has predicted it as 99.74% dandelion. It has predicted accurately.

Image 5:

Predictions of the model: [[0.03595607 0.00166742 0.7122948 0.00169317 0.24838857]]

It is an image of roses. My model has predicted it as 71.22% as rose. It has predicted correctly.

Image 6:

Predictions of the model: [[1.5942901e-04 4.0262475e-07 9.7161901e-01 3.5649848e-08 2.8221097e-02]]

It is an image of rose with a honey bee. My model has predicted it as 97.16% rose. It has predicted accurately.

Image 7:

Predictions of the model: [[6.779737e-04 3.094685e-03 7.905317e-05 9.948519e-01 1.296488e-03]]

It is an image of field of sunflowers. My model has predicted it as 99.48% as sunflower. It has predicted correctly.

Image 8:

Predictions of the model: [[0.03052221 0.00767096 0.01431909 0.892027 0.05546066]]

It is an image of sunflowers along with a person. My model has predicted it as 89.2% sunflower. It has done a great job though there exists a person and blurred background.

Image 9:

Predictions of the model: [[0.0857577 0.03161678 0.36905658 0.05374446 0.45982447]]

It is an image of blue color tulips. My model has predicted it as 45.9% as tulip. It has predicted accurately.

Image 10:

Predictions of the model: [[4.3962379e-03 5.2079465e-04 6.2916839e-01 7.7428886e-05 3.6583713e-01]]

It is an image of pink color tulips field along with a person. My model has predicted as 36.5% tulip and 62.91% as roses. It has predicted inaccurately.

To conclude, the fourth architecture has predicted 8/10 images correctly. The prediction accuracy can be improved by more training of the architecture. Overall, the architecture has done a great job in identifying the flowers of my dataset.