**from** tensorflow.keras.preprocessing.image **import** ImageDataGenerator

In [3]:

train\_datagen**=**ImageDataGenerator(rescale**=**1.**/**255,zoom\_range**=**0.2,horizontal\_flip**=True**,vertical\_flip**=False**)

In [4]:

test\_datagen**=**ImageDataGenerator(rescale**=**1.**/**255)

In [5]:

x\_train**=**train\_datagen**.**flow\_from\_directory(r"C:\Users\maris\_q3mm6nk\Desktop\FILES\data\_for\_ibm\Fertilizers\_Recommendation\_ System\_For\_Disease\_ Prediction\Dataset Plant Disease\Veg-dataset\Veg-dataset\train\_set",target\_size**=**(128,128),

class\_mode**=**'categorical',batch\_size**=**24)

Found 11385 images belonging to 9 classes.

In [6]:

x\_test**=**test\_datagen**.**flow\_from\_directory(r'C:\Users\maris\_q3mm6nk\Desktop\FILES\data\_for\_ibm\Fertilizers\_Recommendation\_ System\_For\_Disease\_ Prediction\Dataset Plant Disease\Veg-dataset\Veg-dataset\test\_set',target\_size**=**(128,128),

class\_mode**=**'categorical',batch\_size**=**24)

Found 3416 images belonging to 9 classes.

In [7]:

**from** tensorflow.keras.models **import** Sequential

**from** tensorflow.keras.layers **import** Dense,Convolution2D,MaxPooling2D,Flatten

In [8]:

model**=**Sequential()

In [9]:

model**.**add(Convolution2D(32,(3,3),input\_shape**=**(128,128,3),activation**=**'relu'))

In [10]:

model**.**add(MaxPooling2D(pool\_size**=**(2,2)))

In [11]:

model**.**add(Flatten())

In [12]:

model**.**summary()

Model: "sequential"

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Layer (type) Output Shape Param #

=================================================================

conv2d (Conv2D) (None, 126, 126, 32) 896

max\_pooling2d (MaxPooling2D (None, 63, 63, 32) 0

)

flatten (Flatten) (None, 127008) 0

=================================================================

Total params: 896

Trainable params: 896

Non-trainable params: 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

In [13]:

model**.**add(Dense(300,activation**=**'relu'))

model**.**add(Dense(150,activation**=**'relu'))

In [20]:

model**.**add(Dense(9,activation**=**'softmax'))

In [21]:

model**.**compile(loss**=**'categorical\_crossentropy',optimizer**=**'adam',metrics**=**['accuracy'])

In [22]:

len(x\_train)

Out[22]:

475

In [23]:

1238**/**24

Out[23]:

51.583333333333336

In [24]:

model**.**fit(x\_train,steps\_per\_epoch**=**len(x\_train),validation\_data**=**x\_test,validation\_steps**=**len(x\_test),epochs**=**10)

Epoch 1/10

475/475 [==============================] - 237s 498ms/step - loss: 2.1787 - accuracy: 0.1331 - val\_loss: 2.1362 - val\_accuracy: 0.1953

Epoch 2/10

475/475 [==============================] - 224s 470ms/step - loss: 2.1077 - accuracy: 0.1868 - val\_loss: 2.1022 - val\_accuracy: 0.1953

Epoch 3/10

475/475 [==============================] - 242s 509ms/step - loss: 2.0872 - accuracy: 0.1868 - val\_loss: 2.0911 - val\_accuracy: 0.1953

Epoch 4/10

475/475 [==============================] - 244s 514ms/step - loss: 2.0795 - accuracy: 0.1868 - val\_loss: 2.0859 - val\_accuracy: 0.1953

Epoch 5/10

475/475 [==============================] - 249s 525ms/step - loss: 2.0761 - accuracy: 0.1868 - val\_loss: 2.0846 - val\_accuracy: 0.1953

Epoch 6/10

475/475 [==============================] - 249s 525ms/step - loss: 2.0745 - accuracy: 0.1868 - val\_loss: 2.0837 - val\_accuracy: 0.1953

Epoch 7/10

475/475 [==============================] - 250s 526ms/step - loss: 2.0738 - accuracy: 0.1868 - val\_loss: 2.0830 - val\_accuracy: 0.1953

Epoch 8/10

475/475 [==============================] - 248s 521ms/step - loss: 2.0735 - accuracy: 0.1868 - val\_loss: 2.0842 - val\_accuracy: 0.1953

Epoch 9/10

475/475 [==============================] - 221s 466ms/step - loss: 2.0734 - accuracy: 0.1868 - val\_loss: 2.0844 - val\_accuracy: 0.1953

Epoch 10/10

475/475 [==============================] - 221s 465ms/step - loss: 2.0734 - accuracy: 0.1868 - val\_loss: 2.0836 - val\_accuracy: 0.1953

Out[24]:

In [25]:

model**.**save('vegetabledata.h5')

In [26]:

**import** numpy **as** np

**from** tensorflow.keras.models **import** load\_model

**from** tensorflow.keras.preprocessing **import** image

In [27]:

model**=**load\_model('vegetabledata.h5')

In [33]:

img**=**image**.**load\_img(r"C:\Users\maris\_q3mm6nk\Desktop\FILES\data\_for\_ibm\Fertilizers\_Recommendation\_ System\_For\_Disease\_ Prediction\Dataset Plant Disease\Veg-dataset\Veg-dataset\test\_set\Potato\_\_\_Early\_blight/b817817e-a6b1-4123-88e7-db98b453ce17\_\_\_RS\_Early.B 6880.jpg")

In [34]:

img

Out[34]:



In [35]:

x**=**image**.**img\_to\_array(img)

In [44]:

img**=**image**.**load\_img(r"C:\Users\maris\_q3mm6nk\Desktop\FILES\data\_for\_ibm\Fertilizers\_Recommendation\_ System\_For\_Disease\_ Prediction\Dataset Plant Disease\Veg-dataset\Veg-dataset\test\_set\Potato\_\_\_Early\_blight/b817817e-a6b1-4123-88e7-db98b453ce17\_\_\_RS\_Early.B 6880.jpg",target\_size**=**(128,128))

img

Out[44]:



In [45]:

x**=**image**.**img\_to\_array(img)

In [46]:

x

Out[46]:

array([[[135., 131., 145.],

[134., 130., 144.],

[133., 129., 143.],

...,

[166., 164., 178.],

[188., 186., 200.],

[213., 211., 225.]],

[[141., 137., 151.],

[139., 135., 149.],

[128., 124., 138.],

...,

[201., 199., 213.],

[157., 155., 169.],

[172., 170., 184.]],

[[136., 132., 146.],

[135., 131., 145.],

[141., 137., 151.],

...,

[166., 164., 178.],

[169., 167., 181.],

[166., 164., 178.]],

...,

[[163., 161., 175.],

[154., 152., 166.],

[160., 158., 172.],

...,

[203., 201., 214.],

[221., 219., 232.],

[207., 205., 218.]],

[[148., 146., 160.],

[165., 163., 177.],

[152., 150., 164.],

...,

[176., 174., 187.],

[192., 190., 203.],

[189., 187., 200.]],

[[162., 160., 174.],

[155., 153., 167.],

[141., 139., 153.],

...,

[180., 178., 191.],

[190., 188., 201.],

[191., 189., 202.]]], dtype=float32)

In [47]:

x**=**np**.**expand\_dims(x,axis**=**0)

In [48]:

x

Out[48]:

array([[[[135., 131., 145.],

[134., 130., 144.],

[133., 129., 143.],

...,

[166., 164., 178.],

[188., 186., 200.],

[213., 211., 225.]],

[[141., 137., 151.],

[139., 135., 149.],

[128., 124., 138.],

...,

[201., 199., 213.],

[157., 155., 169.],

[172., 170., 184.]],

[[136., 132., 146.],

[135., 131., 145.],

[141., 137., 151.],

...,

[166., 164., 178.],

[169., 167., 181.],

[166., 164., 178.]],

...,

[[163., 161., 175.],

[154., 152., 166.],

[160., 158., 172.],

...,

[203., 201., 214.],

[221., 219., 232.],

[207., 205., 218.]],

[[148., 146., 160.],

[165., 163., 177.],

[152., 150., 164.],

...,

[176., 174., 187.],

[192., 190., 203.],

[189., 187., 200.]],

[[162., 160., 174.],

[155., 153., 167.],

[141., 139., 153.],

...,

[180., 178., 191.],

[190., 188., 201.],

[191., 189., 202.]]]], dtype=float32)

In [49]:

y**=**np**.**argmax(model**.**predict(x),axis**=**1)

1/1 [==============================] - 0s 89ms/step

In [50]:

x\_train**.**class\_indices

Out[50]:

{'Pepper,\_bell\_\_\_Bacterial\_spot': 0,

'Pepper,\_bell\_\_\_healthy': 1,

'Potato\_\_\_Early\_blight': 2,

'Potato\_\_\_Late\_blight': 3,

'Potato\_\_\_healthy': 4,

'Tomato\_\_\_Bacterial\_spot': 5,

'Tomato\_\_\_Late\_blight': 6,

'Tomato\_\_\_Leaf\_Mold': 7,

'Tomato\_\_\_Septoria\_leaf\_spot': 8}

In [51]:

index**=**['Pepper,\_bell\_\_\_Bacterial\_spot','Pepper,\_bell\_\_\_healthy','Potato\_\_\_Early\_blight','Potato\_\_\_Late\_blight','Potato\_\_\_healthy','Tomato\_\_\_Bacterial\_spot','Tomato\_\_\_Late\_blight','Tomato\_\_\_Leaf\_Mold','Tomato\_\_\_Septoria\_leaf\_spot']

In [52]:

index[y[0]]

Out[52]:

'Tomato\_\_\_Bacterial\_spot'

In [53]:

img**=**image**.**load\_img(r"C:\Users\maris\_q3mm6nk\Desktop\FILES\data\_for\_ibm\Fertilizers\_Recommendation\_ System\_For\_Disease\_ Prediction\Dataset Plant Disease\Veg-dataset\Veg-dataset\test\_set\Potato\_\_\_Early\_blight/b817817e-a6b1-4123-88e7-db98b453ce17\_\_\_RS\_Early.B 6880.jpg",target\_size**=**(128,128))

x**=**image**.**img\_to\_array(img)

x**=**np**.**expand\_dims(x,axis**=**0)

y**=**np**.**argmax(model**.**predict(x),axis**=**1)

index**=**['Pepper,\_bell\_\_\_Bacterial\_spot','Pepper,\_bell\_\_\_healthy','Potato\_\_\_Early\_blight','Potato\_\_\_Late\_blight','Potato\_\_\_healthy','Tomato\_\_\_Bacterial\_spot','Tomato\_\_\_Leaf\_Mold','Tomato\_\_\_Septoria\_leaf\_spot']

index[y[0]]

1/1 [==============================] - 0s 38ms/step

Out[53]:

'Tomato\_\_\_Bacterial\_spot'

In [ ]: