{

"cells": [

{

"cell\_type": "code",

"execution\_count": null,

"id": "7c4b4d5c",

"metadata": {},

"outputs": [],

"source": [

"from keras.preprocessing.image import ImageDataGenerator"

]

},

{

"cell\_type": "code",

"execution\_count": 65,

"id": "6ad3fa30",

"metadata": {},

"outputs": [],

"source": [

"train\_datagen = ImageDataGenerator(rescale=1./255,shear\_range=0.2,zoom\_range=0.2,horizontal\_flip=True)\n",

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

]

},

{

"cell\_type": "code",

"execution\_count": 66,

"id": "ed4d4293",

"metadata": {},

"outputs": [

{

"name": "stdout",

"output\_type": "stream",

"text": [

"Found 4118 images belonging to 5 classes.\n",

"Found 929 images belonging to 5 classes.\n"

]

}

],

"source": [

"x\_train = train\_datagen.flow\_from\_directory(\n",

" r'C:\\Users\\HP\\Documents\\DataSet\\TRAIN\_SET-20221021T095226Z-001\\TRAIN\_SET',target\_size=(64, 64),batch\_size=5,color\_mode='rgb',class\_mode='sparse')\n",

"x\_test = test\_datagen.flow\_from\_directory(\n",

" r'C:\\Users\\HP\\Documents\\DataSet\\TEST\_SET-20221021T095157Z-001\\TEST\_SET' ,target\_size=(64, 64),batch\_size=5,color\_mode='rgb',class\_mode='sparse')\n"

]

},

{

"cell\_type": "code",

"execution\_count": 67,

"id": "a70e2030",

"metadata": {},

"outputs": [

{

"name": "stdout",

"output\_type": "stream",

"text": [

"{'APPLES': 0, 'BANANA': 1, 'ORANGE': 2, 'PINEAPPLE': 3, 'WATERMELON': 4}\n"

]

}

],

"source": [

"print (x\_train.class\_indices)"

]

},

{

"cell\_type": "code",

"execution\_count": 68,

"id": "e78e7713",

"metadata": {},

"outputs": [

{

"name": "stdout",

"output\_type": "stream",

"text": [

"{'APPLES': 0, 'BANANA': 1, 'ORANGE': 2, 'PINEAPPLE': 3, 'WATERMELON': 4}\n"

]

}

],

"source": [

"print(x\_test.class\_indices)"

]

},

{

"cell\_type": "code",

"execution\_count": 69,

"id": "a1c2cb32",

"metadata": {

"scrolled": true

},

"outputs": [

{

"data": {

"text/plain": [

"Counter({0: 995, 1: 1354, 2: 1019, 3: 275, 4: 475})"

]

},

"execution\_count": 69,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"from collections import Counter as c\n",

"c(x\_train.labels)\n",

"\n"

]

},

{

"cell\_type": "code",

"execution\_count": 70,

"id": "dc6912bd",

"metadata": {},

"outputs": [],

"source": [

"from keras.models import Sequential\n",

"from keras.layers import Dense\n",

"from keras.layers import Conv2D\n",

"from keras.layers import MaxPooling2D\n",

"from keras.layers import Flatten\n"

]

},

{

"cell\_type": "code",

"execution\_count": 71,

"id": "36f1d672",

"metadata": {

"scrolled": false

},

"outputs": [],

"source": [

"import numpy as np"

]

},

{

"cell\_type": "code",

"execution\_count": 72,

"id": "f1021b6b",

"metadata": {},

"outputs": [],

"source": [

"import tensorflow"

]

},

{

"cell\_type": "code",

"execution\_count": 73,

"id": "d809b19d",

"metadata": {

"scrolled": true

},

"outputs": [],

"source": [

"from tensorflow.keras.models import Sequential"

]

},

{

"cell\_type": "code",

"execution\_count": 74,

"id": "7239e628",

"metadata": {

"scrolled": true

},

"outputs": [],

"source": [

"from tensorflow.keras import layers"

]

},

{

"cell\_type": "code",

"execution\_count": 75,

"id": "eba820ec",

"metadata": {},

"outputs": [],

"source": [

"from tensorflow.keras.layers import Dense, Flatten"

]

},

{

"cell\_type": "code",

"execution\_count": 76,

"id": "73bd47f6",

"metadata": {},

"outputs": [],

"source": [

"from tensorflow.keras.layers import Conv2D,MaxPooling2D,Dropout"

]

},

{

"cell\_type": "code",

"execution\_count": 77,

"id": "7798ac53",

"metadata": {},

"outputs": [],

"source": [

"from keras.preprocessing.image import ImageDataGenerator"

]

},

{

"cell\_type": "code",

"execution\_count": 78,

"id": "6769dbb1",

"metadata": {},

"outputs": [],

"source": [

"model=Sequential()"

]

},

{

"cell\_type": "code",

"execution\_count": 79,

"id": "455498d3",

"metadata": {},

"outputs": [],

"source": [

"model.add(Conv2D(32,3,3,input\_shape=(64,64,3),activation='relu'))"

]

},

{

"cell\_type": "code",

"execution\_count": 80,

"id": "fae88e4b",

"metadata": {},

"outputs": [],

"source": [

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

]

},

{

"cell\_type": "code",

"execution\_count": 81,

"id": "2d090a28",

"metadata": {},

"outputs": [],

"source": [

"model.add(Flatten())"

]

},

{

"cell\_type": "code",

"execution\_count": 82,

"id": "354f4509",

"metadata": {},

"outputs": [],

"source": [

"#model.add(Dense(output\_dim=128,activation='relu',init='random\_uniform'))"

]

},

{

"cell\_type": "code",

"execution\_count": 83,

"id": "874c0651",

"metadata": {},

"outputs": [],

"source": [

"#model.add(Dense(output\_dim=1,activation='sigmoid',init='random\_uniform'))"

]

},

{

"cell\_type": "code",

"execution\_count": 84,

"id": "8704fcd5",

"metadata": {},

"outputs": [],

"source": [

"classifier = Sequential()"

]

},

{

"cell\_type": "code",

"execution\_count": 85,

"id": "a926bbb5",

"metadata": {},

"outputs": [],

"source": [

"classifier.add(Conv2D(32, (3, 3), input\_shape=(64, 64, 3), activation='relu'))\n",

"classifier.add(MaxPooling2D(pool\_size=(2, 2)))"

]

},

{

"cell\_type": "code",

"execution\_count": 86,

"id": "eb0b8bad",

"metadata": {},

"outputs": [],

"source": [

"classifier.add(Conv2D(32, (3, 3), activation='relu'))\n",

"\n",

" \n",

"\n",

"\n"

]

},

{

"cell\_type": "code",

"execution\_count": 87,

"id": "b1ca8e74",

"metadata": {},

"outputs": [],

"source": [

"classifier.add(MaxPooling2D(pool\_size=(2, 2)))\n"

]

},

{

"cell\_type": "code",

"execution\_count": 88,

"id": "b84fa096",

"metadata": {},

"outputs": [],

"source": [

"classifier.add(Flatten())"

]

},

{

"cell\_type": "code",

"execution\_count": 90,

"id": "9acf57b9",

"metadata": {},

"outputs": [],

"source": [

"classifier.add(Dense(units=128, activation='relu'))\n",

"classifier.add(Dense(units=5, activation='softmax'))"

]

},

{

"cell\_type": "code",

"execution\_count": 91,

"id": "23c1f8c5",

"metadata": {},

"outputs": [

{

"name": "stdout",

"output\_type": "stream",

"text": [

"Model: \"sequential\_4\"\n",

"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n",

" Layer (type) Output Shape Param # \n",

"=================================================================\n",

" conv2d\_6 (Conv2D) (None, 62, 62, 32) 896 \n",

" \n",

" max\_pooling2d\_8 (MaxPooling (None, 31, 31, 32) 0 \n",

" 2D) \n",

" \n",

" conv2d\_7 (Conv2D) (None, 29, 29, 32) 9248 \n",

" \n",

" max\_pooling2d\_9 (MaxPooling (None, 14, 14, 32) 0 \n",

" 2D) \n",

" \n",

" flatten\_6 (Flatten) (None, 6272) 0 \n",

" \n",

" dense\_4 (Dense) (None, 128) 802944 \n",

" \n",

" dense\_5 (Dense) (None, 5) 645 \n",

" \n",

"=================================================================\n",

"Total params: 813,733\n",

"Trainable params: 813,733\n",

"Non-trainable params: 0\n",

"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n"

]

}

],

"source": [

"classifier.summary()\n"

]

},

{

"cell\_type": "code",

"execution\_count": 92,

"id": "b5aeb89e",

"metadata": {},

"outputs": [],

"source": [

"classifier.compile(optimizer='adam', loss='sparse\_categorical\_crossentropy', metrics=['accuracy'])"

]

},

{

"cell\_type": "code",

"execution\_count": 93,

"id": "cdefcc8a",

"metadata": {},

"outputs": [

{

"name": "stderr",

"output\_type": "stream",

"text": [

"C:\\Users\\HP\\AppData\\Local\\Temp\\ipykernel\_13540\\1881611701.py:1: UserWarning: `Model.fit\_generator` is deprecated and will be removed in a future version. Please use `Model.fit`, which supports generators.\n",

" classifier.fit\_generator(\n"

]

},

{

"name": "stdout",

"output\_type": "stream",

"text": [

"Epoch 1/10\n",

"824/824 [==============================] - 41s 48ms/step - loss: 0.6049 - accuracy: 0.7739 - val\_loss: 0.5004 - val\_accuracy: 0.8116\n",

"Epoch 2/10\n",

"824/824 [==============================] - 41s 50ms/step - loss: 0.4198 - accuracy: 0.8473 - val\_loss: 0.4984 - val\_accuracy: 0.8235\n",

"Epoch 3/10\n",

"824/824 [==============================] - 43s 52ms/step - loss: 0.3806 - accuracy: 0.8555 - val\_loss: 0.4393 - val\_accuracy: 0.8332\n",

"Epoch 4/10\n",

"824/824 [==============================] - 38s 45ms/step - loss: 0.3472 - accuracy: 0.8681 - val\_loss: 0.4119 - val\_accuracy: 0.8504\n",

"Epoch 5/10\n",

"824/824 [==============================] - 37s 45ms/step - loss: 0.3323 - accuracy: 0.8779 - val\_loss: 0.4166 - val\_accuracy: 0.8428\n",

"Epoch 6/10\n",

"824/824 [==============================] - 37s 44ms/step - loss: 0.3072 - accuracy: 0.8834 - val\_loss: 0.4251 - val\_accuracy: 0.8428\n",

"Epoch 7/10\n",

"824/824 [==============================] - 39s 47ms/step - loss: 0.2990 - accuracy: 0.8905 - val\_loss: 0.5612 - val\_accuracy: 0.7783\n",

"Epoch 8/10\n",

"824/824 [==============================] - 48s 58ms/step - loss: 0.2626 - accuracy: 0.8975 - val\_loss: 0.3379 - val\_accuracy: 0.8859\n",

"Epoch 9/10\n",

"824/824 [==============================] - 49s 59ms/step - loss: 0.2587 - accuracy: 0.9004 - val\_loss: 0.4058 - val\_accuracy: 0.8568\n",

"Epoch 10/10\n",

"824/824 [==============================] - 46s 56ms/step - loss: 0.2350 - accuracy: 0.9053 - val\_loss: 0.3717 - val\_accuracy: 0.8794\n"

]

},

{

"data": {

"text/plain": [

"<keras.callbacks.History at 0x1dea14127c0>"

]

},

"execution\_count": 93,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"classifier.fit\_generator(\n",

" generator=x\_train,steps\_per\_epoch = len(x\_train),\n",

" epochs=10, validation\_data=x\_test,validation\_steps = len(x\_test))"

]

},

{

"cell\_type": "code",

"execution\_count": 94,

"id": "27dfd665",

"metadata": {},

"outputs": [],

"source": [

"classifier.save('nutrition.h5')"

]

},

{

"cell\_type": "code",

"execution\_count": 95,

"id": "3dbf69b2",

"metadata": {},

"outputs": [],

"source": [

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

]

},

{

"cell\_type": "code",

"execution\_count": 96,

"id": "86cd3d3d",

"metadata": {},

"outputs": [],

"source": [

"from keras.preprocessing import image\n",

"from tensorflow.keras.preprocessing import image \n",

"model = load\_model(\"nutrition.h5\")"

]

},

{

"cell\_type": "code",

"execution\_count": 97,

"id": "5da73d02",

"metadata": {},

"outputs": [

{

"data": {

"image/png": "\n",

"text/plain": [

"<PIL.Image.Image image mode=RGB size=64x64 at 0x1DE9F9C7AC0>"

]

},

"execution\_count": 97,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"img = image.load\_img(r\"E:\\Flask\\Sample\_Images-20221104T061454Z-001\\Sample\_Images\\Test\_Image1.jpg\", grayscale=False,target\_size= (64,64))\n",

"img\n"

]

},

{

"cell\_type": "code",

"execution\_count": 98,

"id": "8261cae3",

"metadata": {},

"outputs": [

{

"name": "stdout",

"output\_type": "stream",

"text": [

"1/1 [==============================] - 0s 241ms/step\n"

]

},

{

"data": {

"text/plain": [

"0"

]

},

"execution\_count": 98,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"x = image.img\_to\_array(img)\n",

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

"pred = np.argmax(model.predict(x)) \n",

"pred"

]

},

{

"cell\_type": "code",

"execution\_count": 99,

"id": "3bd399b5",

"metadata": {},

"outputs": [

{

"data": {

"text/plain": [

"'APPLES'"

]

},

"execution\_count": 99,

"metadata": {},

"output\_type": "execute\_result"

}

],

"source": [

"labels=['APPLES', 'BANANA', 'ORANGE','PINEAPPLE','WATERMELON']\n",

"labels[np.argmax(pred)]"

]

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "d2d06946",

"metadata": {},

"outputs": [],

"source": []

},

{

"cell\_type": "code",

"execution\_count": null,

"id": "43b2d4b6",

"metadata": {},

"outputs": [],

"source": []

}

],

"metadata": {

"kernelspec": {

"display\_name": "Python 3 (ipykernel)",

"language": "python",

"name": "python3"

},

"language\_info": {

"codemirror\_mode": {

"name": "ipython",

"version": 3

},

"file\_extension": ".py",

"mimetype": "text/x-python",

"name": "python",

"nbconvert\_exporter": "python",

"pygments\_lexer": "ipython3",

"version": "3.9.12"

}

},

"nbformat": 4,

"nbformat\_minor": 5

}