{

"nbformat": 4,

"nbformat\_minor": 0,

"metadata": {

"colab": {

"provenance": []

},

"kernelspec": {

"name": "python3",

"display\_name": "Python 3"

},

"language\_info": {

"name": "python"

}

},

"cells": [

{

"cell\_type": "code",

"execution\_count": 1,

"metadata": {

"colab": {

"base\_uri": "https://localhost:8080/"

},

"id": "OIPBS3CsTuZs",

"outputId": "ed1c6bf8-edc5-42dd-a0f7-3c64b96b57f8"

},

"outputs": [

{

"output\_type": "stream",

"name": "stdout",

"text": [

"/usr/local/lib/python3.7/dist-packages/gdown/cli.py:131: FutureWarning: Option `--id` was deprecated in version 4.3.1 and will be removed in 5.0. You don't need to pass it anymore to use a file ID.\n",

" category=FutureWarning,\n",

"Downloading...\n",

"From: https://drive.google.com/uc?id=1npY\_sDIDyQWjm2ZH4cCCuDhZA9liaNUm\n",

"To: /content/dataset.zip\n",

"100% 523M/523M [00:05<00:00, 98.0MB/s]\n"

]

}

],

"source": [

"! gdown --id 1npY\_sDIDyQWjm2ZH4cCCuDhZA9liaNUm"

]

},

{

"cell\_type": "code",

"source": [

"! unzip '/content/dataset.zip'"

],

"metadata": {

"id": "CZxNutu-UfPE"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

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

],

"metadata": {

"id": "-7AiozwqUnVK"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"#Data Augmentation"

],

"metadata": {

"id": "SSp\_SLOGUsoP"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"train\_datagen = ImageDataGenerator(rescale=1./225,shear\_range=0.2,zoom\_range=0.2,horizontal\_flip=True)"

],

"metadata": {

"id": "jswWUyIPUwod"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"test\_datagen = ImageDataGenerator(rescale=1./225)"

],

"metadata": {

"id": "rljamDTGU2lz"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"x\_train = train\_datagen.flow\_from\_directory('/content/dataset/train\_set' , target\_size=(64,64),batch\_size=5,color\_mode='rgb',class\_mode='categorical')"

],

"metadata": {

"id": "GUUbqoNJVFSh"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "markdown",

"source": [

"Found 742 images belonging to 4 classes.\n"

],

"metadata": {

"id": "usXQ\_eiIVObh"

}

},

{

"cell\_type": "code",

"source": [

"x\_test = test\_datagen.flow\_from\_directory('/content/dataset/test\_set' , target\_size=(64,64),batch\_size=5,color\_mode='rgb',class\_mode='categorical')\n"

],

"metadata": {

"id": "mEuVwe2dVFO\_"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "markdown",

"source": [

"Found 198 images belonging to 4 classes."

],

"metadata": {

"id": "shAaI4w7VkLc"

}

},

{

"cell\_type": "code",

"source": [

"#importing Required Packages"

],

"metadata": {

"id": "XetNtP4KVlWx"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"from tensorflow.keras.models import Sequential\n",

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

],

"metadata": {

"id": "CoNzTEu3VrrR"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"#Building CNN block"

],

"metadata": {

"id": "MgELD1IDV2QY"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"model = Sequential()\n",

"model.add(Convolution2D(32,(3,3), input\_shape=(64,64,3),activation='relu'))\n",

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

"model.add(Convolution2D(32 ,(3,3) , activation='relu'))\n",

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

"model.add(Flatten())\n",

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

"model.add(Dense(units =4 , activation='softmax'))"

],

"metadata": {

"id": "SEhl4DA0V3Zp"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"#Compiling the model"

],

"metadata": {

"id": "ida-RJQ3WDZE"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

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

],

"metadata": {

"id": "v05lIC-oWHQj"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"model.fit\_generator(generator=x\_train ,steps\_per\_epoch= len(x\_train),epochs=20, validation\_data=x\_test,validation\_steps=len(x\_test))\n"

],

"metadata": {

"id": "1oVrLmduXTyC"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "markdown",

"source": [

"Epoch 1/20\n",

"149/149 [==============================] - 37s 188ms/step - loss: 1.2177 - accuracy: 0.4367 - val\_loss: 1.5993 - val\_accuracy: 0.4646\n",

"Epoch 2/20\n",

"149/149 [==============================] - 28s 187ms/step - loss: 0.9553 - accuracy: 0.5943 - val\_loss: 0.9020 - val\_accuracy: 0.6515\n",

"Epoch 3/20\n",

"149/149 [==============================] - 28s 186ms/step - loss: 0.7419 - accuracy: 0.7116 - val\_loss: 0.7201 - val\_accuracy: 0.7424\n",

"Epoch 4/20\n",

"149/149 [==============================] - 28s 187ms/step - loss: 0.6761 - accuracy: 0.7345 - val\_loss: 0.7405 - val\_accuracy: 0.7576\n",

"Epoch 5/20\n",

"149/149 [==============================] - 28s 188ms/step - loss: 0.6356 - accuracy: 0.7480 - val\_loss: 0.8016 - val\_accuracy: 0.7424\n",

"Epoch 6/20\n",

"149/149 [==============================] - 30s 202ms/step - loss: 0.5759 - accuracy: 0.7749 - val\_loss: 0.9617 - val\_accuracy: 0.6869\n",

"Epoch 7/20\n",

"149/149 [==============================] - 28s 185ms/step - loss: 0.5246 - accuracy: 0.8181 - val\_loss: 0.7854 - val\_accuracy: 0.7071\n",

"Epoch 8/20\n",

"149/149 [==============================] - 28s 188ms/step - loss: 0.4662 - accuracy: 0.8248 - val\_loss: 0.6588 - val\_accuracy: 0.7273\n",

"Epoch 9/20\n",

"149/149 [==============================] - 28s 188ms/step - loss: 0.4304 - accuracy: 0.8302 - val\_loss: 0.6534 - val\_accuracy: 0.7727\n",

"Epoch 10/20\n",

"149/149 [==============================] - 28s 187ms/step - loss: 0.3771 - accuracy: 0.8544 - val\_loss: 0.8804 - val\_accuracy: 0.7222\n",

"Epoch 11/20\n",

"149/149 [==============================] - 28s 188ms/step - loss: 0.3379 - accuracy: 0.8733 - val\_loss: 0.9850 - val\_accuracy: 0.7222\n",

"Epoch 12/20\n",

"149/149 [==============================] - 28s 185ms/step - loss: 0.3635 - accuracy: 0.8464 - val\_loss: 0.7546 - val\_accuracy: 0.7727\n",

"Epoch 13/20\n",

"149/149 [==============================] - 28s 190ms/step - loss: 0.3426 - accuracy: 0.8733 - val\_loss: 0.8590 - val\_accuracy: 0.7222\n",

"Epoch 14/20\n",

"149/149 [==============================] - 28s 188ms/step - loss: 0.2759 - accuracy: 0.8949 - val\_loss: 0.9976 - val\_accuracy: 0.7374\n",

"Epoch 15/20\n",

"149/149 [==============================] - 28s 187ms/step - loss: 0.3028 - accuracy: 0.8854 - val\_loss: 1.4439 - val\_accuracy: 0.6313\n",

"Epoch 16/20\n",

"149/149 [==============================] - 28s 185ms/step - loss: 0.2939 - accuracy: 0.8949 - val\_loss: 0.7897 - val\_accuracy: 0.7576\n",

"Epoch 17/20\n",

"149/149 [==============================] - 29s 197ms/step - loss: 0.2254 - accuracy: 0.9191 - val\_loss: 1.0229 - val\_accuracy: 0.7677\n",

"Epoch 18/20\n",

"149/149 [==============================] - 28s 187ms/step - loss: 0.2084 - accuracy: 0.9218 - val\_loss: 1.0623 - val\_accuracy: 0.7323\n",

"Epoch 19/20\n",

"149/149 [==============================] - 28s 186ms/step - loss: 0.1692 - accuracy: 0.9394 - val\_loss: 1.0719 - val\_accuracy: 0.7576\n",

"Epoch 20/20\n",

"149/149 [==============================] - 28s 186ms/step - loss: 0.1843 - accuracy: 0.9340 - val\_loss: 0.9710 - val\_accuracy: 0.7525\n"

],

"metadata": {

"id": "wG5IrrtcYX8Q"

}

},

{

"cell\_type": "code",

"source": [

"#Saving Model"

],

"metadata": {

"id": "PJdAmY1eYi5U"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"model.save('disaster.h5')"

],

"metadata": {

"id": "OaVRpWlsYrrQ"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"#Testing the model"

],

"metadata": {

"id": "4tf2rx1JYvnq"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"import numpy as np\n",

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

],

"metadata": {

"id": "MUHt3LhzYzZJ"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "code",

"source": [

"img = image.load\_img('/content/dataset/test\_set/Flood/1015.jpg' , target\_size=(64,64))\n",

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

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

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

"print(pred,model.predict(x))\n",

"op=['Cyclone','Earthquake','Flood','Wildfire']\n",

"print(op[pred])"

],

"metadata": {

"id": "56rxRxZtZDSc"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "markdown",

"source": [

"1/1 [==============================] - 0s 124ms/step\n",

"1/1 [==============================] - 0s 16ms/step\n",

"2 [[0. 0. 1. 0.]]\n",

"Flood"

],

"metadata": {

"id": "n6UOHzhlZEad"

}

},

{

"cell\_type": "code",

"source": [

"img = image.load\_img('/content/dataset/test\_set/Earthquake/1327.jpg' , target\_size=(64,64))\n",

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

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

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

"print(pred,model.predict(x))\n",

"op=['Cyclone','Earthquake','Flood','Wildfire']\n",

"print(op[pred])"

],

"metadata": {

"id": "\_BwmPT5zZNPH"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "markdown",

"source": [

"1/1 [==============================] - 0s 15ms/step\n",

"1/1 [==============================] - 0s 14ms/step\n",

"1 [[0. 1. 0. 0.]]\n",

"Earthquake"

],

"metadata": {

"id": "aJiDmU3CZbHz"

}

},

{

"cell\_type": "code",

"source": [

"img = image.load\_img('/content/dataset/test\_set/Wildfire/1050.jpg' , target\_size=(64,64))\n",

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

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

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

"print(pred,model.predict(x))\n",

"op=['Cyclone','Earthquake','Flood','Wildfire']\n",

"print(op[pred])"

],

"metadata": {

"id": "n4Vw0lDNZiMh"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "markdown",

"source": [

"1/1 [==============================] - 0s 16ms/step\n",

"1/1 [==============================] - 0s 16ms/step\n",

"1 [[0. 1. 0. 0.]]\n",

"Earthquake"

],

"metadata": {

"id": "a40uvvi8ZrAH"

}

},

{

"cell\_type": "code",

"source": [

"img = image.load\_img('/content/test.jpg' , target\_size=(64,64))\n",

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

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

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

"print(pred,model.predict(x))\n",

"op=['Cyclone','Earthquake','Flood','Wildfire']\n",

"print(op[pred])"

],

"metadata": {

"id": "qZh-2NCtZt2F"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "markdown",

"source": [

"1/1 [==============================] - 0s 16ms/step\n",

"1/1 [==============================] - 0s 16ms/step\n",

"0 [[1. 0. 0. 0.]]\n",

"Cyclone"

],

"metadata": {

"id": "f7AXCKdWZ67O"

}

},

{

"cell\_type": "code",

"source": [

"img = image.load\_img('/content/images.jpeg' , target\_size=(64,64))\n",

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

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

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

"print(pred,model.predict(x))\n",

"op=['Cyclone','Earthquake','Flood','Wildfire']\n",

"print(op[pred])"

],

"metadata": {

"id": "joBtoi2AaFnX"

},

"execution\_count": null,

"outputs": []

},

{

"cell\_type": "markdown",

"source": [

"1/1 [==============================] - 0s 125ms/step\n",

"1/1 [==============================] - 0s 14ms/step\n",

"1 [[0. 1. 0. 0.]]\n",

"Earthquake"

],

"metadata": {

"id": "OWCrTO4NaI3v"

}

}

]

}