Assignment-3 Python coding:


```
{
 "nbformat": 4,
 "nbformat_minor": 0,
 "metadata": {
  "colab": {
   "provenance": [],
   "collapsed_sections": []
  },
  "kernelspec": {
   "name": "python3",
   "display_name": "Python 3"
  },
  "language_info": {
   "name": "python"
  }
 },
 "cells": [
  {
   "cell_type": "code",
   "execution_count": 1,
   "metadata": {
    "id": "FTE7A-sOr64q"
   },
   "outputs": [],
   "source": [
    "from tensorflow.keras.preprocessing.image import ImageDataGenerator"
   ]
  },
  {
   "cell_type": "code",
   "source": [
```

```
    "\n",
    "train_datagen = ImageDataGenerator(rescale=1./255,\n",
    "                                   zoom_range=0.2,\n",
    "                                   horizontal_flip=True)"
   ],
   "metadata": {
    "id": "8epn6wrksGvg"
   },
   "execution_count": 3,
   "outputs": []
  },
  {
   "cell_type": "code",
   "source": [
    "test_datagen = ImageDataGenerator(rescale=1./255)"
   ],
   "metadata": {
    "id": "EdJNJYJFsMjo"
   },
   "execution_count": 4,
   "outputs": []
  },
  {
   "cell_type": "code",
   "source": [
    "xtrain = train_datagen.flow_from_directory('/content//drive/MyDrive/flowers',\n",
    "                                           target_size=(64,64),\n",
    "                                           class_mode='categorical',\n",
    "                                           batch_size=100)"
   ],
   "metadata": {
    "colab": {
```

```
      "base_uri": "https://localhost:8080/"
    },
    "id": "QlVXYwR5sQda",
    "outputId": "8aac08ef-889c-4baa-a50c-d534865770c8"
  },
  "execution_count": 7,
  "outputs": [
    {
      "output_type": "stream",
      "name": "stdout",
      "text": [
        "Found 4317 images belonging to 5 classes.\n"
      ]
    }
  ]
},
{
  "cell_type": "markdown",
  "source": [
    "IMAGE AUGMENTATION"
  ],
  "metadata": {
    "id": "nbz4ZjW7ulQ3"
  }
},
{
  "cell_type": "code",
  "source": [
    "from tensorflow.keras.models import Sequential\n",
    "from tensorflow.keras.layers import Convolution2D,MaxPooling2D,Flatten,Dense"
  ],
  "metadata": {
```

```
      "id": "uk-UGFvysrna"
     },
     "execution_count": 8,
     "outputs": []
    },
    {
     "cell_type": "markdown",
     "source": [
      "CNN MODEL"
     ],
     "metadata": {
      "id": "jJX_UuwyvAJZ"
     }
    },
    {
     "cell_type": "code",
     "source": [
      "model = Sequential()\n",
      "model.add(Convolution2D(32,(3,3),activation='relu',input_shape=(64,64,3))) # Convolution layer\n",
      "model.add(MaxPooling2D(pool_size=(2,2))) # Max pooling layer\n",
      "model.add(Flatten()) # Flatten layer\n",
      "# Fully connected layers (ANN)\n",
      "model.add(Dense(300,activation='relu')) # Hidden layer 1\n",
      "model.add(Dense(150,activation='relu')) # Hidden layer 2\n",
      "model.add(Dense(4,activation='softmax')) # Output layer"
     ],
     "metadata": {
      "id": "-LBcIWAWsvhX"
     },
     "execution_count": 9,
     "outputs": []
    },
```

```
{
 "cell_type": "markdown",
 "source": [
  "COMPILE THE MODEL"
 ],
 "metadata": {
  "id": "m1tlC26eu46Y"
 }
},
{
 "cell_type": "code",
 "source": [
  "model.compile(optimizer='adam',loss='categorical_crossentropy',metrics=['accuracy'])"
 ],
 "metadata": {
  "id": "DpOZAIkJsy3Q"
 },
 "execution_count": 10,
 "outputs": []
},
{
 "cell_type": "markdown",
 "source": [
  "SAVE THE MODEL"
 ],
 "metadata": {
  "id": "I8308Zuvu9ZP"
 }
},
{
 "cell_type": "code",
 "source": [
```

```
    "model.save('Flower.h5')"
   ],
   "metadata": {
    "id": "FSgxTAhqtNIX"
   },
   "execution_count": 16,
   "outputs": []
  },
  {
   "cell_type": "markdown",
   "source": [
    "TEST THE MODEL"
   ],
   "metadata": {
    "id": "xlEyckOnvFef"
   }
  },
  {
   "cell_type": "code",
   "source": [
    "import numpy as np\n",
    "from tensorflow.keras.preprocessing import image"
   ],
   "metadata": {
    "id": "f8wbDr-ktexL"
   },
   "execution_count": 17,
   "outputs": []
  },
  {
   "cell_type": "code",
   "source": [
```

```
      "img = image.load_img('/content/drive/MyDrive/flowers/dandelion/10043234166_e6dd915111_n.jpg',target_size=(64,64))"
     ],
     "metadata": {
      "id": "IP1YnDHMthh4"
     },
     "execution_count": 18,
     "outputs": []
    },
    {
     "cell_type": "code",
     "source": [
      "img"
     ],
     "metadata": {
      "colab": {
       "base_uri": "https://localhost:8080/",
       "height": 81
      },
      "id": "-jDXjeZWtn0A",
      "outputId": "6987fef7-073b-4535-a49b-1a40a511ef6c"
     },
     "execution_count": 19,
     "outputs": [
      {
       "output_type": "execute_result",
       "data": {
        "text/plain": [
         "<PIL.Image.Image image mode=RGB size=64x64 at 0x7FC823134A10>"
        ],
        "image/png":
       },
```

```
      "metadata": {},
      "execution_count": 19
    }
  ]
},
{
  "cell_type": "code",
  "source": [
    "# Converting image to array\n",
    "\n",
    "x = image.img_to_array(img)\n",
    "x"
  ],
  "metadata": {
    "colab": {
      "base_uri": "https://localhost:8080/"
    },
    "id": "ZQROn0eWtuPv",
    "outputId": "22695a00-7c1c-4916-9667-b98aab50ed34"
  },
  "execution_count": 20,
  "outputs": [
    {
      "output_type": "execute_result",
      "data": {
        "text/plain": [
          "array([[[ 3.,  6.,  0.],\n",
          "        [ 8.,  9.,  1.],\n",
          "        [ 7.,  8.,  0.],\n",
          "        ...,\n",
          "        [35., 44.,  1.],\n",
          "        [35., 44.,  1.],\n",
```

```
"       [35., 43.,  2.]],\n",
"\n",
"       [[ 5.,  8.,  1.],\n",
"        [ 6.,  7.,  0.],\n",
"        [ 6.,  7.,  1.],\n",
"        ...,\n",
"        [30., 36.,  0.],\n",
"        [30., 38.,  0.],\n",
"        [31., 39.,  0.]],\n",
"\n",
"       [[ 7.,  8.,  2.],\n",
"        [ 7.,  8.,  2.],\n",
"        [ 5.,  8.,  0.],\n",
"        ...,\n",
"        [19., 22.,  1.],\n",
"        [19., 24.,  1.],\n",
"        [23., 29.,  3.]],\n",
"\n",
"       ...,\n",
"\n",
"       [[17., 20.,  1.],\n",
"        [23., 27.,  2.],\n",
"        [24., 30.,  2.],\n",
"        ...,\n",
"        [23., 27.,  0.],\n",
"        [23., 29.,  1.],\n",
"        [21., 27.,  1.]],\n",
"\n",
"       [[16., 19.,  0.],\n",
"        [23., 28.,  0.],\n",
"        [26., 31.,  1.],\n",
"        ...,\n",
```

```
          "       [19., 23.,  0.],\n",
          "       [25., 27.,  5.],\n",
          "       [19., 24.,  1.]],\n",
          "\n",
          "      [[17., 20.,  1.],\n",
          "       [22., 26.,  0.],\n",
          "       [26., 31.,  1.],\n",
          "       ...,\n",
          "       [18., 21.,  0.],\n",
          "       [20., 24.,  1.],\n",
          "       [21., 25.,  2.]]], dtype=float32)"
        ]
      },
      "metadata": {},
      "execution_count": 20
    }
  ]
},
{
  "cell_type": "code",
  "source": [
    "# Expanding dimensions\n",
    "\n",
    "x = np.expand_dims(x,axis=0)\n",
    "x"
  ],
  "metadata": {
    "colab": {
      "base_uri": "https://localhost:8080/"
    },
    "id": "knGc0Z99tyH_",
    "outputId": "76bcfe1e-be3d-4613-e804-f5367f010b90"
```

```
},
"execution_count": 21,
"outputs": [
  {
    "output_type": "execute_result",
    "data": {
      "text/plain": [
        "array([[[[ 3.,  6.,  0.],\n",
        "         [ 8.,  9.,  1.],\n",
        "         [ 7.,  8.,  0.],\n",
        "         ...,\n",
        "         [35., 44.,  1.],\n",
        "         [35., 44.,  1.],\n",
        "         [35., 43.,  2.]],\n",
        "\n",
        "        [[ 5.,  8.,  1.],\n",
        "         [ 6.,  7.,  0.],\n",
        "         [ 6.,  7.,  1.],\n",
        "         ...,\n",
        "         [30., 36.,  0.],\n",
        "         [30., 38.,  0.],\n",
        "         [31., 39.,  0.]],\n",
        "\n",
        "        [[ 7.,  8.,  2.],\n",
        "         [ 7.,  8.,  2.],\n",
        "         [ 5.,  8.,  0.],\n",
        "         ...,\n",
        "         [19., 22.,  1.],\n",
        "         [19., 24.,  1.],\n",
        "         [23., 29.,  3.]],\n",
        "\n",
        "        ...,\n",
```

```
        "\n",
        "       [[17., 20.,  1.],\n",
        "        [23., 27.,  2.],\n",
        "        [24., 30.,  2.],\n",
        "        ...,\n",
        "        [23., 27.,  0.],\n",
        "        [23., 29.,  1.],\n",
        "        [21., 27.,  1.]],\n",
        "\n",
        "       [[16., 19.,  0.],\n",
        "        [23., 28.,  0.],\n",
        "        [26., 31.,  1.],\n",
        "        ...,\n",
        "        [19., 23.,  0.],\n",
        "        [25., 27.,  5.],\n",
        "        [19., 24.,  1.]],\n",
        "\n",
        "       [[17., 20.,  1.],\n",
        "        [22., 26.,  0.],\n",
        "        [26., 31.,  1.],\n",
        "        ...,\n",
        "        [18., 21.,  0.],\n",
        "        [20., 24.,  1.],\n",
        "        [21., 25.,  2.]]], dtype=float32)"
      ]
     },
     "metadata": {},
     "execution_count": 21
    }
   ]
  },
  {
```

```
  "cell_type": "code",
  "source": [
    "model.predict(x)"
  ],
  "metadata": {
    "colab": {
      "base_uri": "https://localhost:8080/"
    },
    "id": "FuKiB9vht1EX",
    "outputId": "9d4d156b-f9c0-41f8-ba9d-e7cdfc45e49d"
  },
  "execution_count": 22,
  "outputs": [
    {
      "output_type": "execute_result",
      "data": {
        "text/plain": [
          "array([[8.1779763e-02, 9.1822016e-01, 2.1105427e-24, 1.6366634e-27]],\n",
          "       dtype=float32)"
        ]
      },
      "metadata": {},
      "execution_count": 22
    }
  ]
},
{
  "cell_type": "code",
  "source": [
    "op = ['daisy','dandelion','rose','sunflower','tulip']\n",
    "pred = np.argmax(model.predict(x))\n",
    "op[pred]"
```

```
 ],
 "metadata": {
  "colab": {
   "base_uri": "https://localhost:8080/",
   "height": 35
  },
  "id": "j3gbpKghuLMn",
  "outputId": "25028b1a-439c-4986-e99f-e18bc5583c3d"
 },
 "execution_count": 25,
 "outputs": [
  {
   "output_type": "execute_result",
   "data": {
    "text/plain": [
     "'dandelion'"
    ],
    "application/vnd.google.colaboratory.intrinsic+json": {
     "type": "string"
    }
   },
   "metadata": {},
   "execution_count": 25
  }
 ]
},
{
 "cell_type": "code",
 "source": [
  "# Testing\n",
  "\n",
```

```
      "img = image.load_img('/content/drive/MyDrive/flowers/daisy/1031799732_e7f4008c03.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",
      "op[pred]"
    ],
    "metadata": {
      "colab": {
        "base_uri": "https://localhost:8080/",
        "height": 35
      },
      "id": "JHoB5Iqjt9Ef",
      "outputId": "ebbffb2f-aece-43e0-d7b0-080bc000a0dc"
    },
    "execution_count": 26,
    "outputs": [
      {
        "output_type": "execute_result",
        "data": {
          "text/plain": [
            "'dandelion'"
          ],
          "application/vnd.google.colaboratory.intrinsic+json": {
            "type": "string"
          }
        },
        "metadata": {},
        "execution_count": 26
      }
    ]
  }
```

]
}