-RSIP Career Basic AI 090

Project Title:

Rock Identification using Deep Convolution Neural Network

Submitted by,

Gayatri Kumari

1.INTRODUCTION:

Overview:

Rock are a fundamental component of earth. The automatic identification of rock type in the field would aid geological surveying, education and automatic mapping.

Visual inspection assesses properties such as colour,shape,etc.It is a basic part of geological surveying, research and education. The traditional method of rock classification is a time-consuming process and has less accuracy rate.

So, this project basically aims on making a system using deep CNN which can detect the rock type and help people in knowing the types instantly.

Purpose:

To get the type of rock within seconds or mins (decrease time in detecting the type of rock.)

Detecting the type of rock plays a major role in geological surveying. Type of the rock reflects their chemical composition and in turn would be very useful if the time to predicting the type would be less. This model is also helpful for those who don't have that much geological knowledge but want to know the type of the rock.

2.Literature survey:

This section summarises some of the scholarly and research works in the field of deep convolution neural network for classifying the type of rock.

Existing problem:

The main aim of this project is to classify the rocks.

So, I have to create a system that can identify rocks with high accuracy than the traditional method (less accurate and time consuming).

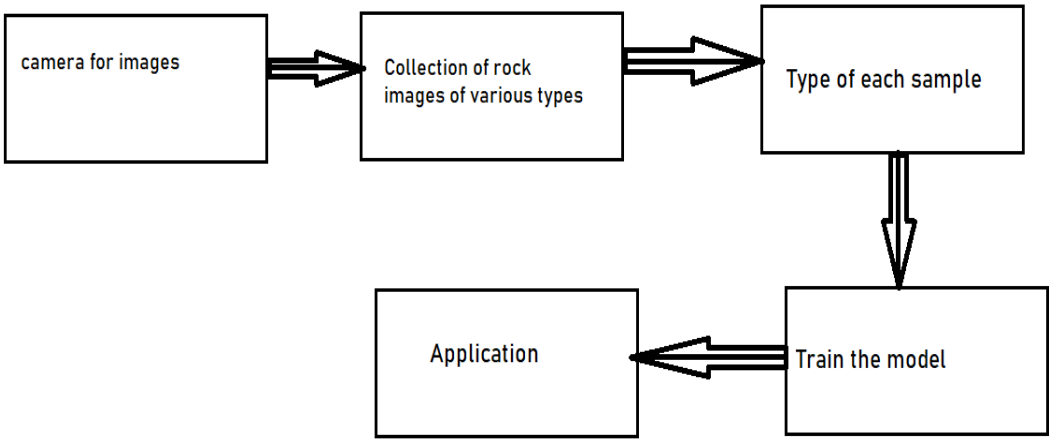
Proposed solution:

First model is optimization model for minima and maxima values. Second module is for colour similarity measure. Third is for time complexity measure. Fourth is for collection of images. Fifth module is Deep learning CNN. Lastly, they have performance evaluation.

referred pdf:https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9042306

3.Theoritical Analysis:

Block diagram:



Hardware/Software design:

Hardware tools:

Camera

Software tools:

-Tensor Flow

-Keras

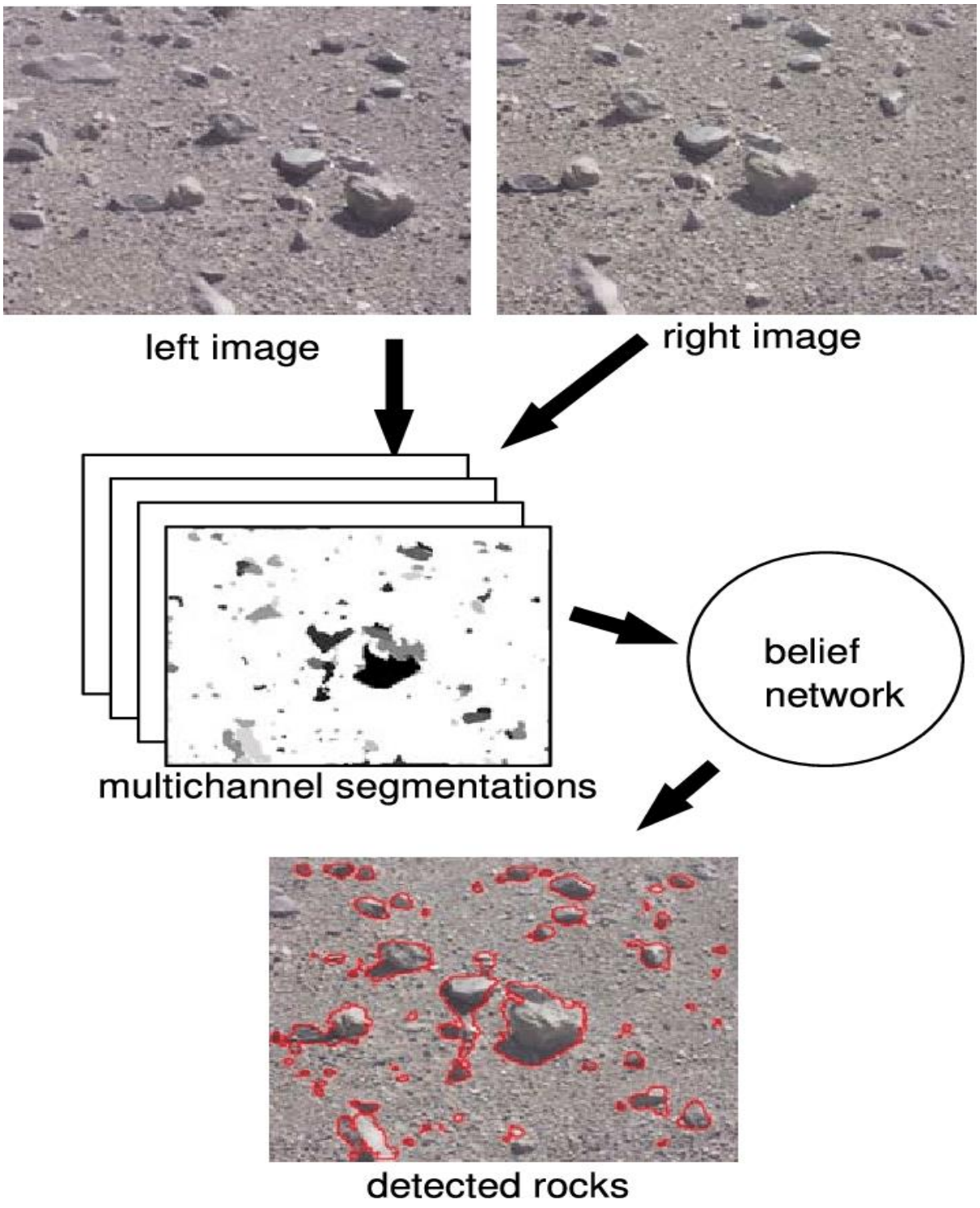
-OpenCV

-Flask

-Operating system for python,html,css,js

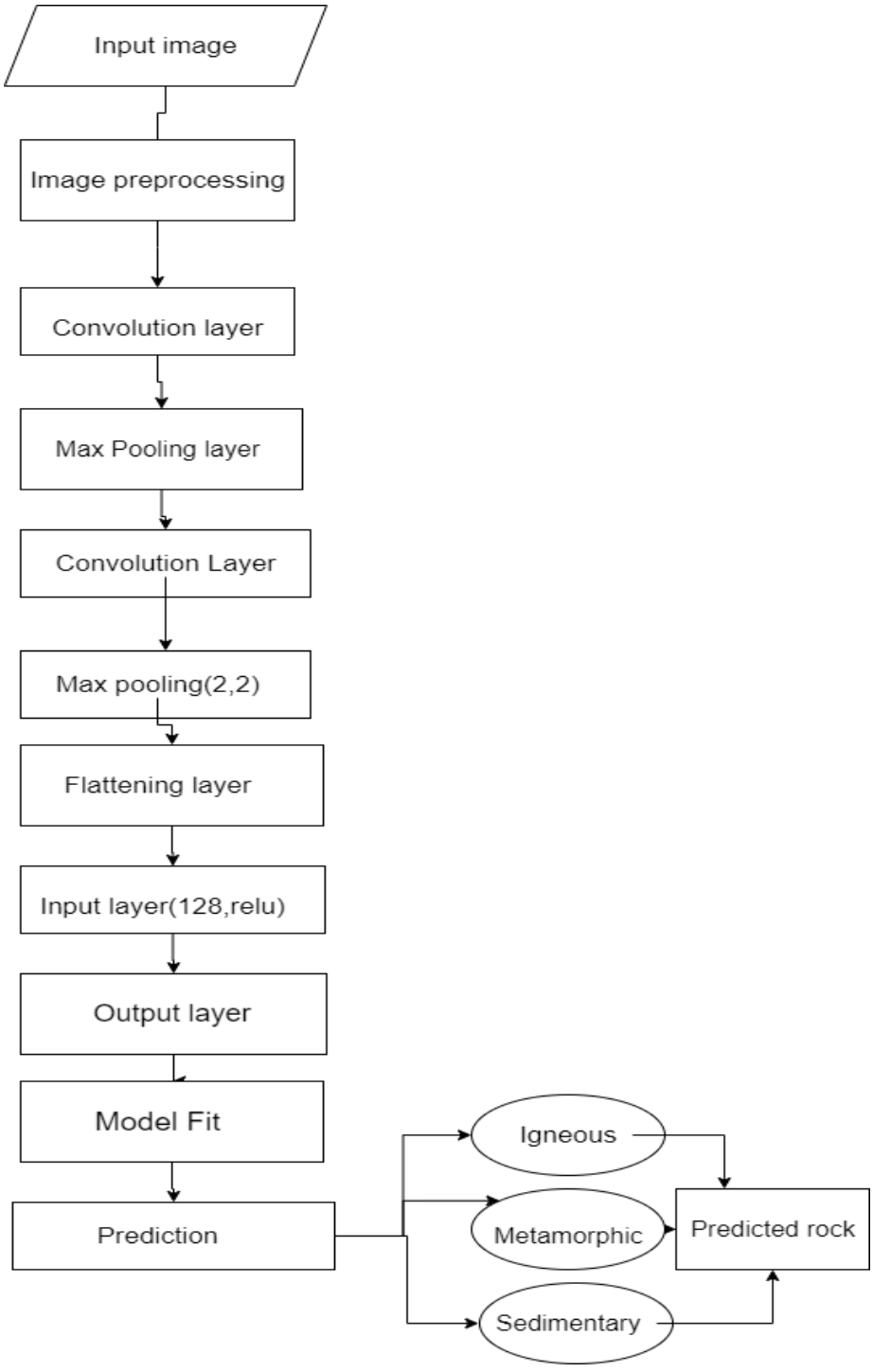
4.Experimental Investigation:

A convolution layer extracts the features of the input images and by convolution and outputs the feature maps. It is composed of series of fixed size filters, known as convolution kernels, which are used to perform convolution.



This image describes the procedure of rock detection.

5.Flowchart:



6.Result:

I got an accuracy of 90.6 percent which is a good percentile for CNN.

7.Advantages and Disadvantages:

Advantages:

1.High speed.

2.Predicts the type just by having clear image

3.Effective and predicts with 90 per or more accuracy.

4.Efficiency can be increased by training the model.

Disadvantages:

1.Requires clear images

2.High computational cost.

8.Application:

Can be used for geological research, even for education purpose.

The traditional method has many drawbacks such as less accuracy and time consuming so this can be used to overcome those drawbacks.

9.Conclusion:

This project helps in identifying the rock type effectively using CNN. This experiment shows that this has high reliability whether in HSV or RGB colour space. In Rgb colour space the efficiency acquired is 91 percent which is good. In view of using CNN for rock classification this can be considered a good way for classifying rocks.

10.Future scope:

rock classification helps in identifying the type of rock fastly .

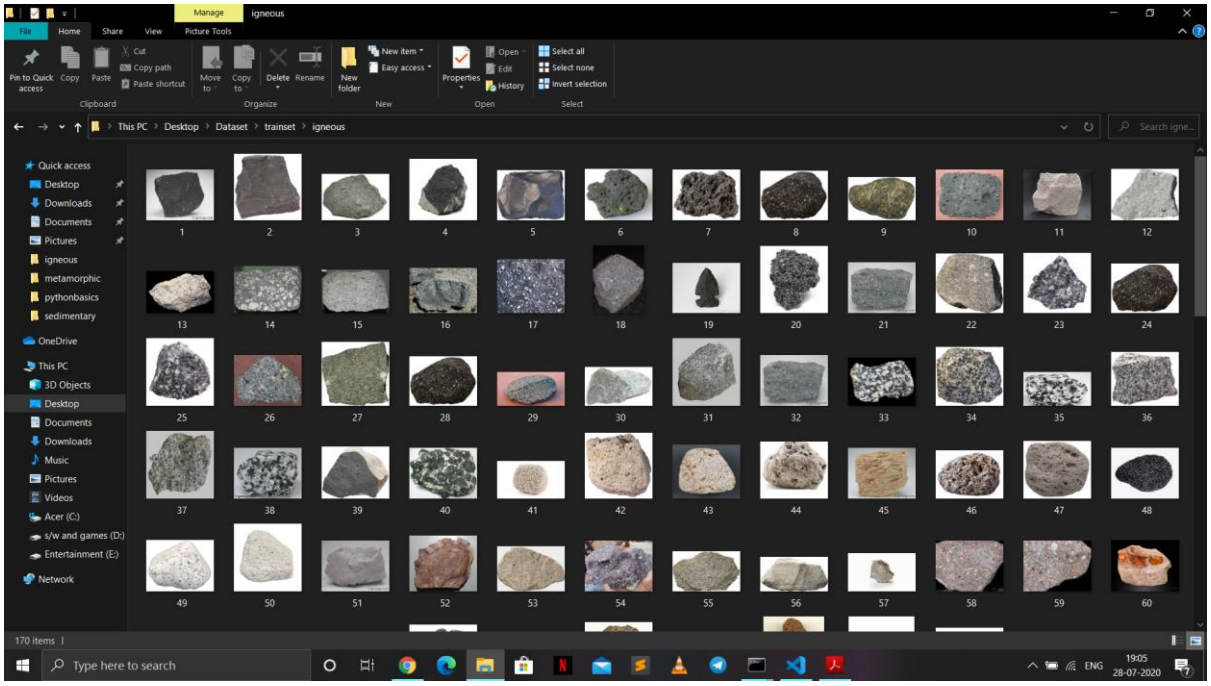
11.Biblography:

idea:

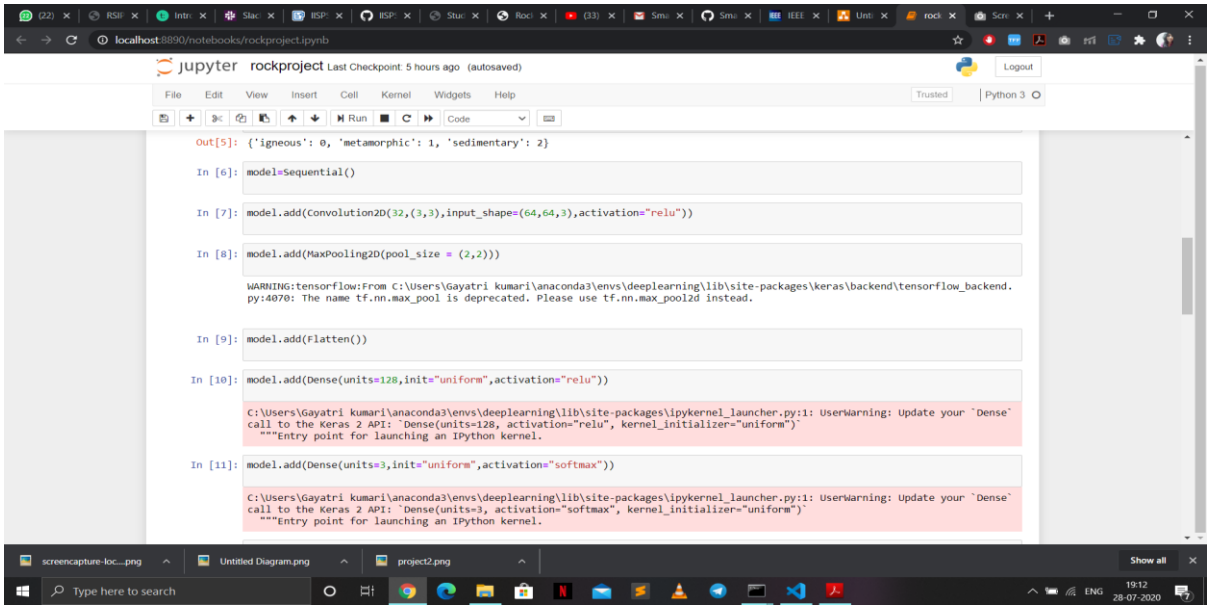
https://www.researchgate.net/publication/319590805_Rock_images_classification_by_using_deep_convolution_neural_networkhttps://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9042306

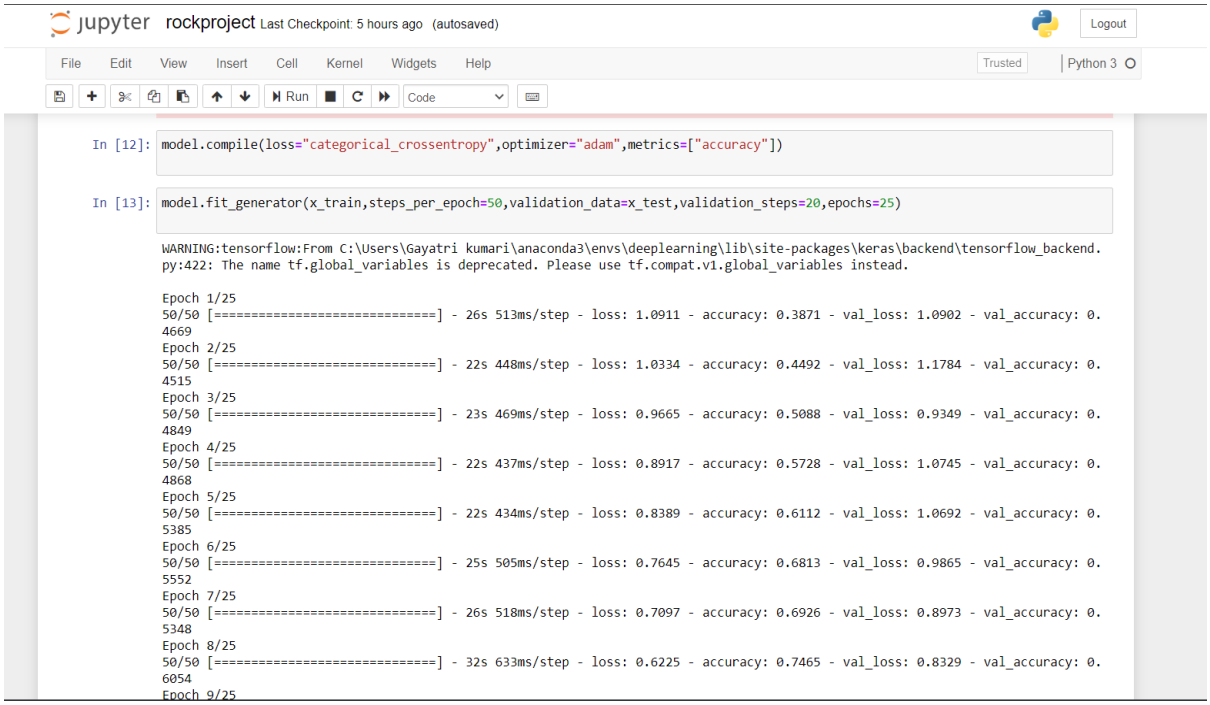
12.Appendix:

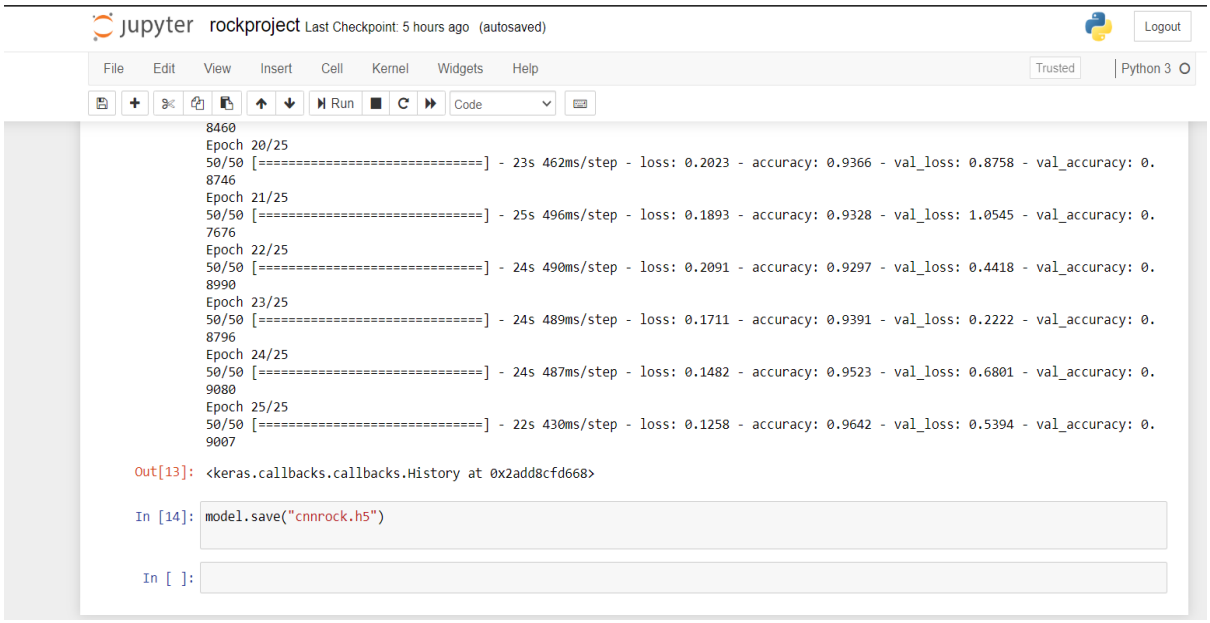
Dataset



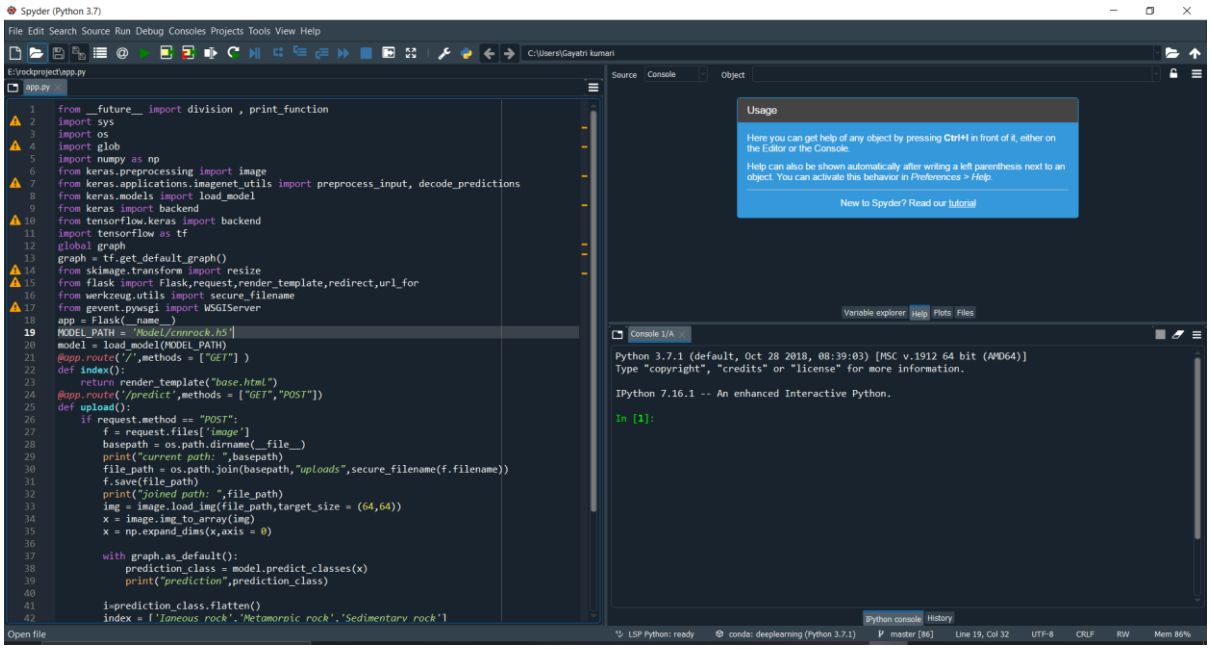
Model training:

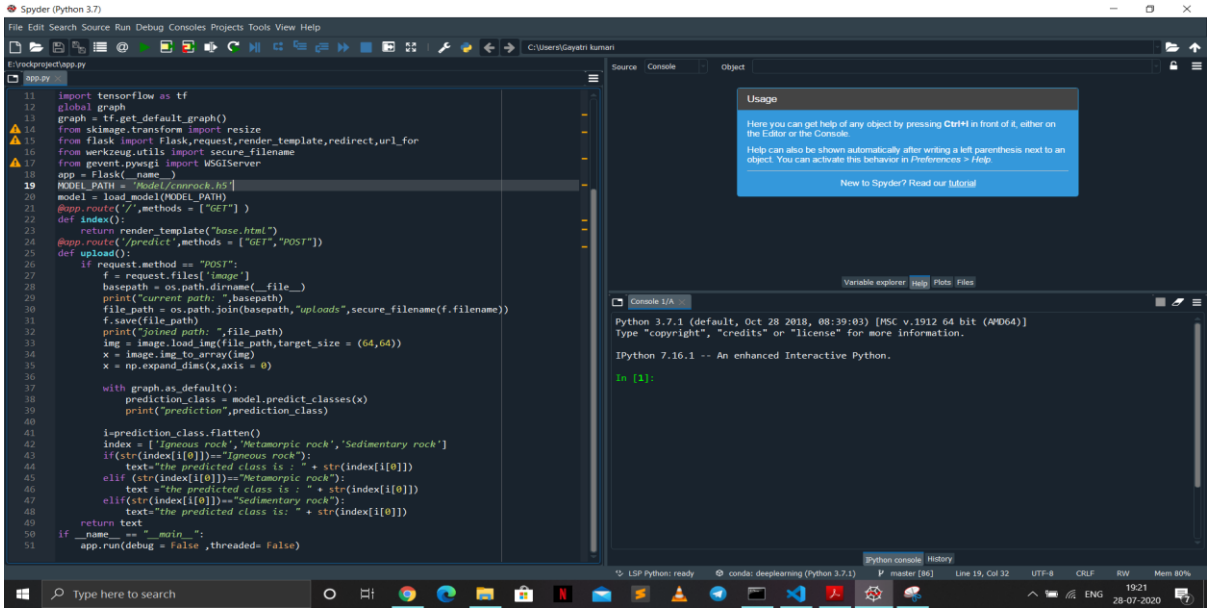


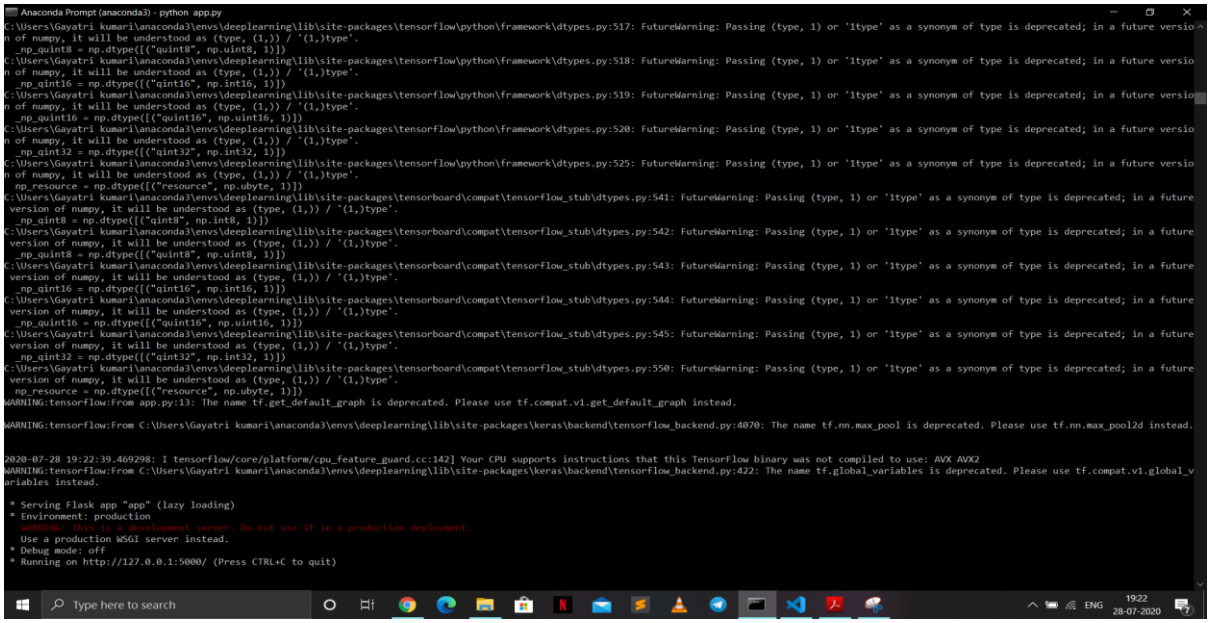

jupyter rockproject Last Checkpoint: 5 hours ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
In [12]: model.compile(loss="categorical_crossentropy",optimizer="adam",metrics=["accuracy"])
In [13]: model.fit_generator(x_train,steps_per_epoch=50,validation_data=x_test,validation_steps=20,epochs=25)
WARNING:tensorflow:From C:\Users\Gayatri kumari\anaconda3\envs\deeplearning\lib\site-packages\keras\backend\tensorflow_backend.py:422: The name tf.global_variables is deprecated. Please use tf.compat.v1.global_variables instead.
Epoch 1/25
50/50 [==============================] - 26s 513ms/step - loss: 1.0911 - accuracy: 0.3871 - val_loss: 1.0902 - val_accuracy: 0.4669
Epoch 2/25
50/50 [==============================] - 22s 448ms/step - loss: 1.0334 - accuracy: 0.4492 - val_loss: 1.1784 - val_accuracy: 0.4515
Epoch 3/25
50/50 [==============================] - 23s 469ms/step - loss: 0.9665 - accuracy: 0.5088 - val_loss: 0.9349 - val_accuracy: 0.4849
Epoch 4/25
50/50 [==============================] - 22s 437ms/step - loss: 0.8917 - accuracy: 0.5728 - val_loss: 1.0745 - val_accuracy: 0.4868
Epoch 5/25
50/50 [==============================] - 22s 434ms/step - loss: 0.8389 - accuracy: 0.6112 - val_loss: 1.0692 - val_accuracy: 0.5385
Epoch 6/25
50/50 [==============================] - 25s 505ms/step - loss: 0.7645 - accuracy: 0.6813 - val_loss: 0.9865 - val_accuracy: 0.5552
Epoch 7/25
50/50 [==============================] - 26s 518ms/step - loss: 0.7097 - accuracy: 0.6926 - val_loss: 0.8973 - val_accuracy: 0.5348
Epoch 8/25
50/50 [==============================] - 32s 633ms/step - loss: 0.6225 - accuracy: 0.7465 - val_loss: 0.8329 - val_accuracy: 0.6054
Epoch 9/25


jupyter rockproject Last Checkpoint: 5 hours ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
8460
Epoch 20/25
50/50 [==============================] - 23s 462ms/step - loss: 0.2023 - accuracy: 0.9366 - val_loss: 0.8758 - val_accuracy: 0.8746
Epoch 21/25
50/50 [==============================] - 25s 496ms/step - loss: 0.1893 - accuracy: 0.9328 - val_loss: 1.0545 - val_accuracy: 0.7676
Epoch 22/25
50/50 [==============================] - 24s 490ms/step - loss: 0.2091 - accuracy: 0.9297 - val_loss: 0.4418 - val_accuracy: 0.8990
Epoch 23/25
50/50 [==============================] - 24s 489ms/step - loss: 0.1711 - accuracy: 0.9391 - val_loss: 0.2222 - val_accuracy: 0.8796
Epoch 24/25
50/50 [==============================] - 24s 487ms/step - loss: 0.1482 - accuracy: 0.9523 - val_loss: 0.6801 - val_accuracy: 0.9080
Epoch 25/25
50/50 [==============================] - 22s 430ms/step - loss: 0.1258 - accuracy: 0.9642 - val_loss: 0.5394 - val_accuracy: 0.9007
Out[13]: <keras.callbacks.callbacks.History at 0x2add8cfd668>
In [14]: model.save("cnnrock.h5")
In [ ]:

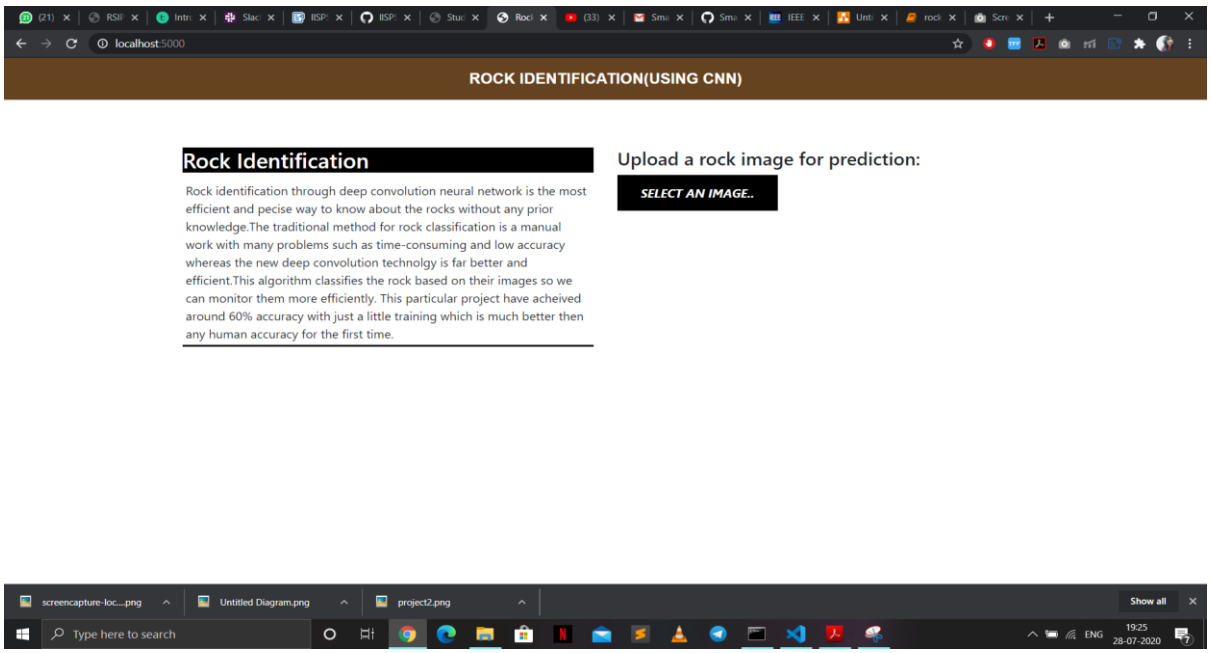
Flask:


Spyder (Python 3.7)
File Edit Search Source Run Debug Consoles Projects Tools View Help
E:\rockproject\app.py
app.py
from __future__ import division , print_function
import sys
import os
import glob
import numpy as np
from keras.preprocessing import image
from keras.applications.imagenet_utils import preprocess_input, decode_predictions
from keras.models import load_model
from keras import backend
from tensorflow.keras import backend
import tensorflow as tf
global graph
graph = tf.get_default_graph()
from skimage.transform import resize
from flask import Flask,request,render_template,redirect,url_for
from werkzeug.utils import secure_filename
from gevent.pywsgi import WSGIServer
app = Flask(__name__)
MODEL_PATH = 'Model/cnnrock.h5'
model = load_model(MODEL_PATH)
@app.route('/',methods = ["GET"])
def index():
return render_template("base.html")
@app.route('/predict',methods = ["GET","POST"])
def upload():
if request.method == "POST":
f = request.files['image']
basepath = os.path.dirname(__file__)
print("current path: ",basepath)
file_path = os.path.join(basepath,"uploads",secure_filename(f.filename))
f.save(file_path)
print("joined path: ",file_path)
img = image.load_img(file_path,target_size = (64,64))
x = image.img_to_array(img)
x = np.expand_dims(x,axis = 0)
with graph.as_default():
prediction_class = model.predict_classes(x)
print("prediction",prediction_class)
i=prediction_class.flatten()
index = ['Igneous rock','Metamorpic rock','Sedimentary rock']
Source Console Object
Usage
Here you can get help of any object by pressing Ctrl+I in front of it, either on the Editor or the Console.
Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in Preferences > Help.
New to Spyder? Read our tutorial
Variable explorer Help Plots Files
Console 1/A
Python 3.7.1 (default, Oct 28 2018, 08:39:03) [MSC v.1912 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 7.16.1 -- An enhanced Interactive Python.
In [1]:
IPython console History
Open file
LSP Python: ready
conda: deeplearning (Python 3.7.1)
master [86]
Line 19, Col 32
UTF-8
CRLF
RW
Mem 86%

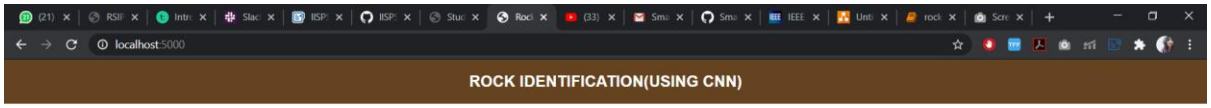

Spyder (Python 3.7)
File Edit Search Source Run Debug Consoles Projects Tools View Help
E:\rockproject\app.py
app.py
import tensorflow as tf
global graph
graph = tf.get_default_graph()
from skimage.transform import resize
from flask import Flask,request,render_template,redirect,url_for
from werkzeug.utils import secure_filename
from gevent.pywsgi import WSGIServer
app = Flask(__name__)
MODEL_PATH = 'Model/cnnrock.h5'
model = load_model(MODEL_PATH)
@app.route('/',methods = ["GET"])
def index():
return render_template("base.html")
@app.route('/predict',methods = ["GET","POST"])
def upload():
if request.method == "POST":
f = request.files['image']
basepath = os.path.dirname(__file__)
print("current path: ",basepath)
file_path = os.path.join(basepath,"uploads",secure_filename(f.filename))
f.save(file_path)
print("joined path: ",file_path)
img = image.load_img(file_path,target_size = (64,64))
x = image.img_to_array(img)
x = np.expand_dims(x,axis = 0)
with graph.as_default():
prediction_class = model.predict_classes(x)
print("prediction",prediction_class)
i=prediction_class.flatten()
index = ['Igneous rock','Metamorpic rock','Sedimentary rock']
if(str(index[i[0]])=="Igneous rock"):
text="the predicted class is : " + str(index[i[0]])
elif (str(index[i[0]])=="Metamorpic rock"):
text ="the predicted class is : " + str(index[i[0]])
elif(str(index[i[0]])=="Sedimentary rock"):
text="the predicted class is: " + str(index[i[0]])
return text
if __name__ == "__main__":
app.run(debug = False ,threaded= False)
Source Console Object
Usage
Here you can get help of any object by pressing Ctrl+I in front of it, either on the Editor or the Console.
Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in Preferences > Help.
New to Spyder? Read our tutorial
Variable explorer Help Plots Files
Console 1/A
Python 3.7.1 (default, Oct 28 2018, 08:39:03) [MSC v.1912 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 7.16.1 -- An enhanced Interactive Python.
In [1]:
IPython console History
LSP Python: ready
conda: deeplearning (Python 3.7.1)
master [86]
Line 19, Col 32
UTF-8
CRLF
RW
Mem 86%
Type here to search
ENG
19:21
28-07-2020


Anaconda Prompt (anaconda3) - python app.py
C:\Users\Gayatri kumari\anaconda3\envs\deeplearning\lib\site-packages\tensorflow\python\framework\dtypes.py:517: FutureWarning: Passing (type, 1) or '1type' as a synonym of type is deprecated; in a future version of numpy, it will be understood as (type, (1,)) / '(1,)type'.
_np_quint8 = np.dtype([("quint8", np.uint8, 1)])
C:\Users\Gayatri kumari\anaconda3\envs\deeplearning\lib\site-packages\tensorflow\python\framework\dtypes.py:518: FutureWarning: Passing (type, 1) or '1type' as a synonym of type is deprecated; in a future version of numpy, it will be understood as (type, (1,)) / '(1,)type'.
_np_qint16 = np.dtype([("qint16", np.int16, 1)])
C:\Users\Gayatri kumari\anaconda3\envs\deeplearning\lib\site-packages\tensorflow\python\framework\dtypes.py:519: FutureWarning: Passing (type, 1) or '1type' as a synonym of type is deprecated; in a future version of numpy, it will be understood as (type, (1,)) / '(1,)type'.
_np_quint16 = np.dtype([("quint16", np.uint16, 1)])
C:\Users\Gayatri kumari\anaconda3\envs\deeplearning\lib\site-packages\tensorflow\python\framework\dtypes.py:520: FutureWarning: Passing (type, 1) or '1type' as a synonym of type is deprecated; in a future version of numpy, it will be understood as (type, (1,)) / '(1,)type'.
_np_qint32 = np.dtype([("qint32", np.int32, 1)])
C:\Users\Gayatri kumari\anaconda3\envs\deeplearning\lib\site-packages\tensorflow\python\framework\dtypes.py:525: FutureWarning: Passing (type, 1) or '1type' as a synonym of type is deprecated; in a future version of numpy, it will be understood as (type, (1,)) / '(1,)type'.
np_resource = np.dtype([("resource", np.ubyte, 1)])
C:\Users\Gayatri kumari\anaconda3\envs\deeplearning\lib\site-packages\tensorboard\compat\tensorflow_stub\dtypes.py:541: FutureWarning: Passing (type, 1) or '1type' as a synonym of type is deprecated; in a future version of numpy, it will be understood as (type, (1,)) / '(1,)type'.
_np_qint8 = np.dtype([("qint8", np.int8, 1)])
C:\Users\Gayatri kumari\anaconda3\envs\deeplearning\lib\site-packages\tensorboard\compat\tensorflow_stub\dtypes.py:542: FutureWarning: Passing (type, 1) or '1type' as a synonym of type is deprecated; in a future version of numpy, it will be understood as (type, (1,)) / '(1,)type'.
_np_quint8 = np.dtype([("quint8", np.uint8, 1)])
C:\Users\Gayatri kumari\anaconda3\envs\deeplearning\lib\site-packages\tensorboard\compat\tensorflow_stub\dtypes.py:543: FutureWarning: Passing (type, 1) or '1type' as a synonym of type is deprecated; in a future version of numpy, it will be understood as (type, (1,)) / '(1,)type'.
_np_qint16 = np.dtype([("qint16", np.int16, 1)])
C:\Users\Gayatri kumari\anaconda3\envs\deeplearning\lib\site-packages\tensorboard\compat\tensorflow_stub\dtypes.py:544: FutureWarning: Passing (type, 1) or '1type' as a synonym of type is deprecated; in a future version of numpy, it will be understood as (type, (1,)) / '(1,)type'.
_np_quint16 = np.dtype([("quint16", np.uint16, 1)])
C:\Users\Gayatri kumari\anaconda3\envs\deeplearning\lib\site-packages\tensorboard\compat\tensorflow_stub\dtypes.py:545: FutureWarning: Passing (type, 1) or '1type' as a synonym of type is deprecated; in a future version of numpy, it will be understood as (type, (1,)) / '(1,)type'.
_np_qint32 = np.dtype([("qint32", np.int32, 1)])
C:\Users\Gayatri kumari\anaconda3\envs\deeplearning\lib\site-packages\tensorboard\compat\tensorflow_stub\dtypes.py:550: FutureWarning: Passing (type, 1) or '1type' as a synonym of type is deprecated; in a future version of numpy, it will be understood as (type, (1,)) / '(1,)type'.
np_resource = np.dtype([("resource", np.ubyte, 1)])
WARNING:tensorflow:From app.py:13: The name tf.get_default_graph is deprecated. Please use tf.compat.v1.get_default_graph instead.
WARNING:tensorflow:From C:\Users\Gayatri kumari\anaconda3\envs\deeplearning\lib\site-packages\keras\backend\tensorflow_backend.py:4070: The name tf.nn.max_pool is deprecated. Please use tf.nn.max_pool2d instead.
2020-07-28 19:22:39.469298: I tensorflow/core/platform/cpu_feature_guard.cc:142] Your CPU supports instructions that this TensorFlow binary was not compiled to use: AVX AVX2
WARNING:tensorflow:From C:\Users\Gayatri kumari\anaconda3\envs\deeplearning\lib\site-packages\keras\backend\tensorflow_backend.py:422: The name tf.global_variables is deprecated. Please use tf.compat.v1.global_variables instead.
* Serving Flask app "app" (lazy loading)
* Environment: production
WARNING: This is a development server. Do not use it in a production deployment.
Use a production WSGI server instead.
* Debug mode: off
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
Type here to search
ENG
19:22
28-07-2020

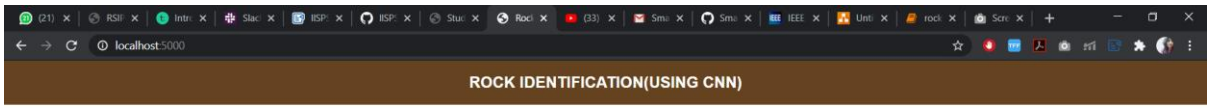
OUTPUT:


localhost:5000
ROCK IDENTIFICATION(USING CNN)
Rock Identification
Rock identification through deep convolution neural network is the most efficient and pecise way to know about the rocks without any prior knowledge.The traditional method for rock classification is a manual work with many problems such as time-consuming and low accuracy whereas the new deep convolution technolgy is far better and efficient.This algorithm classifies the rock based on their images so we can monitor them more efficiently. This particular project have acheived around 60% accuracy with just a little training which is much better then any human accuracy for the first time.
Upload a rock image for prediction:
SELECT AN IMAGE..

screencapture-loc....png
Untitled Diagram.png
project2.png
Show all
Type here to search
ENG
19:25
28-07-2020


localhost:5000
ROCK IDENTIFICATION(USING CNN)
Rock Identification
Rock identification through deep convolution neural network is the most efficient and pecise way to know about the rocks without any prior knowledge.The traditional method for rock classification is a manual work with many problems such as time-consuming and low accuracy whereas the new deep convolution technolgy is far better and efficient.This algorithm classifies the rock based on their images so we can monitor them more efficiently. This particular project have acheived around 60% accuracy with just a little training which is much better then any human accuracy for the first time.
Upload a rock image for prediction:
SELECT AN IMAGE..
click to get what type of rock it is


Type here to search
ENG
19:32
28-07-2020


localhost:5000
ROCK IDENTIFICATION(USING CNN)
Rock Identification
Rock identification through deep convolution neural network is the most efficient and pecise way to know about the rocks without any prior knowledge.The traditional method for rock classification is a manual work with many problems such as time-consuming and low accuracy whereas the new deep convolution technolgy is far better and efficient.This algorithm classifies the rock based on their images so we can monitor them more efficiently. This particular project have acheived around 60% accuracy with just a little training which is much better then any human accuracy for the first time.
Upload a rock image for prediction:
SELECT AN IMAGE..
Result: the predicted class is : Igneous rock


Type here to search
ENG
19:32
28-07-2020