

Google Earth Report

Vishal Ranjan

Department of Computer Science

North Dakota State University

Fargo, ND 58102

vishal.ranjan@ndsu.edu

Abstract:

This project is implemented to address the problem on what are the conditions of endangered species and where they found the most and which of them is endangered, vulnerable and critically endangered. In this project we also tried to mark down the places according to the temperature i.e. the places which are hot we tried to mark with a flame icon and place which are cold we marked it with a snow icon. As we all know how deforestation is increasing day by day, in this project we tried to mark the places where deforestation is happening a lot. While searching for the deforestation we came to know where the deforestation is happening a lot. According to pbs.org Indonesia, Brazil, Nigeria has the highest rate in the world. In this project we tried to mark the deforested area in Brazil. So the main aim of this project is to develop an idea that how much the humans are destroying the Earth. For implementation of this project Google Earth Pro is required.

Introduction:

Google Earth is a program that renders a 3D representation of Earth based on satellite imagery. The program maps the Earth by satellite images, aerial photography and GIS data on a 3D globe. User can explore the world by just entering the coordinates, or by using the mouse or keyboard. In Google Earth user can add place mark or can add polygon around the area. User can make a KML (Keyhole Markup language) file and create a place mark or polygon around the place by just writing the latitude and longitude. In this project we tried to use Python to make a KML file and then implement it on Google Earth.

Project Work:

a) Weather Data by NOAA

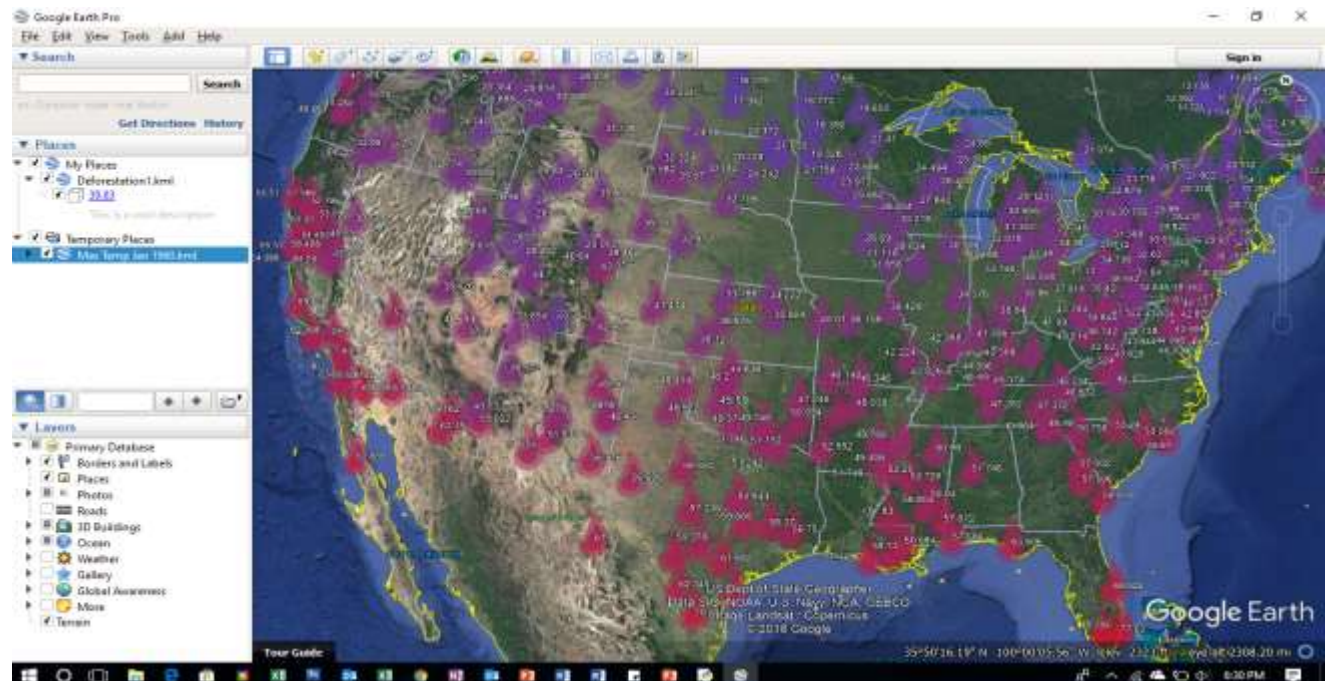
In the project we downloaded the 7.21GB weather data from NOAA which contained around 102,284 CSV files. The files contained various data from stations over the course of many years. We created fire and snow icons to mark the place according to the temperature. “cElementTree” Library is helpful for creating a new place mark in Google Earth. In the python file we parsed the CSV files and according to the latitude and longitude we created the place mark with the icon. So the places which had temp greater than 40 we put a fire icon and if it is less than 0 we put a snow icon on that location. The temperature in between has an icon of light blue which indicates neither hot nor cold.



STATION	DATE	LATITUDE	LONGITUDE	ELEVATION	NAME	CDSD	CDSD_ATTRIBUTES	CLDD
ACW00011604	1949-01	17.11667	-61.78333	10.1	ST JOHNS COOLIDGE FIELD, AC	182.6		182.6
ACW00011604	1949-02	17.11667	-61.78333	10.1	ST JOHNS COOLIDGE FIELD, AC	348		165.4
ACW00011604	1949-03	17.11667	-61.78333	10.1	ST JOHNS COOLIDGE FIELD, AC	544.7		196.7
ACW00011604	1949-04	17.11667	-61.78333	10.1	ST JOHNS COOLIDGE FIELD, AC	764.5		219.8
ACW00011604	1949-05	17.11667	-61.78333	10.1	ST JOHNS COOLIDGE FIELD, AC	1020.6		256.1
ACW00011604	1949-06	17.11667	-61.78333	10.1	ST JOHNS COOLIDGE FIELD, AC	1275		254.4
ACW00011604	1949-07	17.11667	-61.78333	10.1	ST JOHNS COOLIDGE FIELD, AC	1552.6		277.6

(A sample of data from NOAA)

Demo:



Sample Python File

```
TV "C:\Users\Arun\Desktop\NDGU Courses\computational tech\Project>CreateRMFromCSV.py"
File Edit Format Run Options Windows Help

#Find TAVU in first row
firstRow = reader.next()

for i in range(0, len(firstRow)):
    if firstRow[i] == "TAVU":
        if firstRow[i] == "TAVU":
            tempInFile = True
            tempIndex = i
            break

#If there is a TAVU get the first value
#If there is a TAVU get the first value
if tempInFile:
    #Find year 1900
    containsYear = False
    yearRow = []

    #Look for 1900 in year string
    for row in reader:
        if row[i].find("1900-01") != -1:
            row[tempIndex] += " "
            containsYear = True
            yearRow = row
            break

    #If row was found add it to the stations array
    if containsYear:
        tempStation = Station()
        tempStation.name = yearRow[5]
        tempStation.date = yearRow[1]
        tempStation.latitude = yearRow[2]
        tempStation.longitude = yearRow[3]
        tempStation.temperature = yearRow[tempIndex]
        stations.append(tempStation)

#Get List of temps
temps = []
for i in range(0, len(stations)):
    if stations[i].temperature != "":
        temps.append(float(stations[i].temperature))

max = max(temps)
```

b) Endangered Species List

In this part of project we created a CSV file by taking the data from WWF (World Wide Life). In the CSV file we created different columns like the name of the species, the scientific name, place where they found the most, their condition and most importantly Longitude and Latitude. We made the icons of species according to their condition. Red icon depicted Critically Endangered species, Orange depicted Endangered Species and Yellow suggested vulnerable species. We parsed the CSV file with the same logic and created place mark according to coordinates with the suitable icon. We tried to add all the things in place mark which contained scientific names, place and the website link so that anyone can get the information about the species.



Vulnerable



Endangered

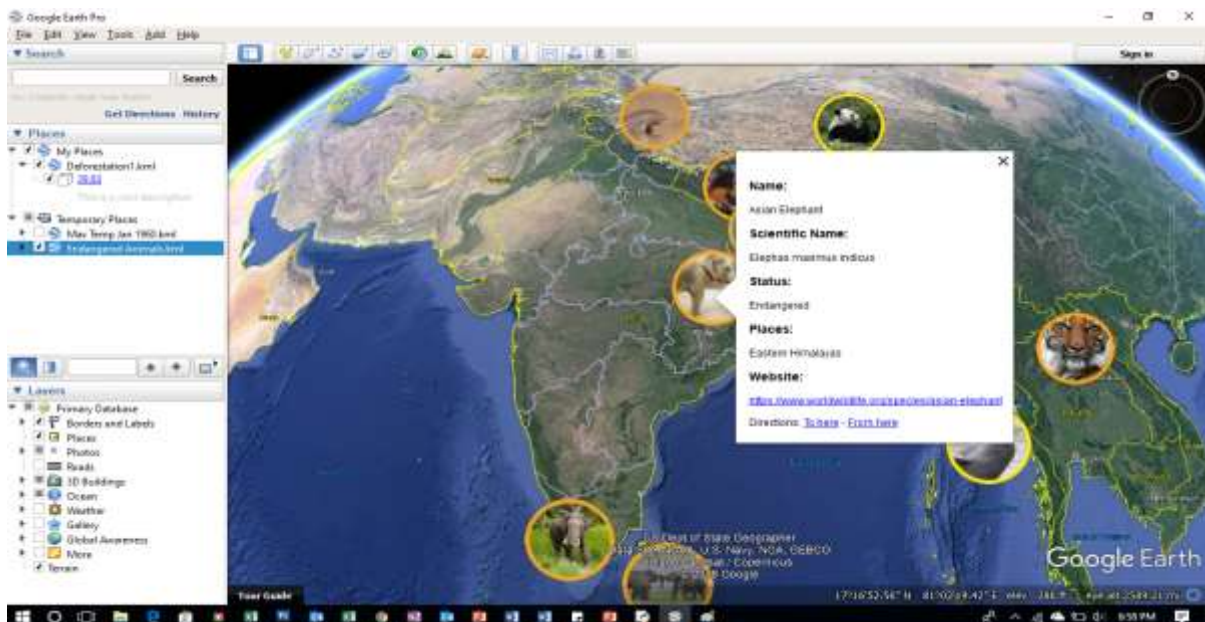


Critically Endangered

COMMON NAME	SCIENTIFIC NAME	CONSERVATION STATUS	Website	Places	Longitude(e/w)	Latitude(N/S)
African Elephant	<i>Loxodonta africana</i>	Vulnerable	https://www.worldwildlife.org/species/african-elephant	Mombo woodlands	36.716557	-8.68177
African Wild Dog	<i>Lycaon pictus</i>	Endangered	https://www.worldwildlife.org/species/african-wild-dog	East Africa	37.2972	1.9577
Amur Leopard	<i>Panthera pardus orientalis</i>	Critically Endangered	https://www.worldwildlife.org/species/amur-leopard	Amur-Hailong	108.770278	48.816389
Amur Tiger	<i>Panthera tigris altaica</i>	Endangered	https://www.worldwildlife.org/species/amur-tiger	Amur-Hailong	108.770278	48.816389
Asian Elephant	<i>Elephas maximus indicus</i>	Endangered	https://www.worldwildlife.org/species/asian-elephant	Eastern Himalayas	82.7	21.95
Bengal Tiger	<i>Panthera tigris tigris</i>	Endangered	https://www.worldwildlife.org/species/bengal-tiger	India, Nepal, Bhutan	87.855	22.9668
Bigeye Tuna	<i>Thunnus obesus</i>	Vulnerable	https://www.worldwildlife.org/species/bigeye-tuna	Coastal East Africa	39.170939	-26.700093

(Sample of the CSV file)

Demo:



Sample Python File:-

```
EndangeredKML.py - C:\Users\Vishal Ranjan\Desktop\NDSU Courses\computational techno\Endangered Species\Endangered Animal List Pictures\cont\EndangeredKML.py
File Edit Format Run Options Windows Help

import csv
import xml.etree.cElementTree as ET
import os
import string

#Define a Animal class
class Animal:
    def __init__(self, name = "", sciName = "", status = "", website = "", places = "", latitude = 0, longitude = 0):
        self.latitude = latitude
        self.longitude = longitude
        self.sciName = sciName
        self.name = name
        self.status = status
        self.website = website
        self.places = places

#Load CSV
#Create new list of Animals
animals = []

with open(str("C:\Users\Vishal Ranjan\Desktop\NDSU Courses\computational techno\Endangered Species\List.csv"), 'rb') as csvfile:
    reader = csv.reader(csvfile, delimiter=',', quotechar='"')

    #Get first row so its not read later
    firstRow = reader.next()

    #Look for 1960 in year string
    for row in reader:
        temp = Animal(row[0],row[1],row[2],row[3],row[4],row[5],row[6])
        animals.append(temp)
        #print(temp.name)

#BUILD XML
# build a tree structure
root = ET.Element("kml", xmlns="http://www.opengis.net/kml/2.2")
#Use document for multiple placemarkers
document = ET.SubElement(root, "Document")

#Go through all animals
for i in range(0, len(animals)):

```

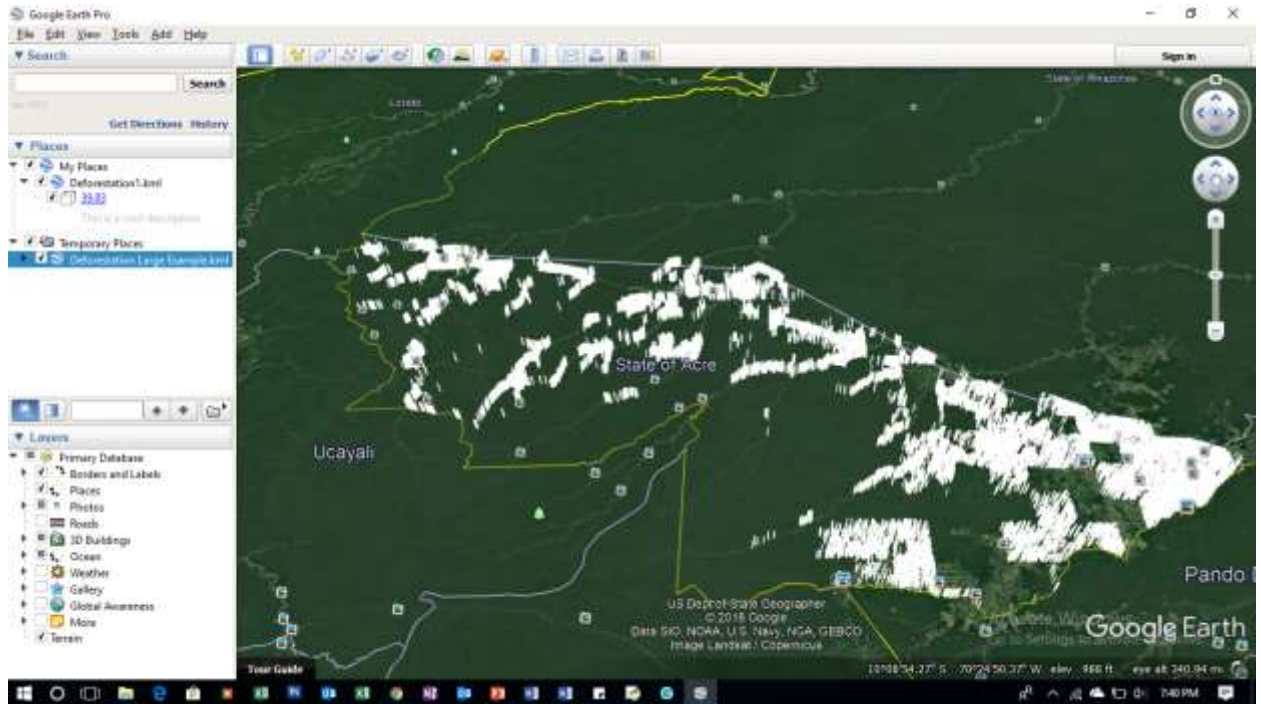
c) Deforested Land

In this part of the project, we found the data of Deforested lands in Brazil. The data was collected from PRODES (Brazilian Gov. Program). The file was about 2.1GB in size which was too large to open in Google Earth. We made a python file in which we parsed only the selected area with the help of coordinates. We created a polygon around the area where deforestation is happening. The red area in the pic depicted the area which is deforested.

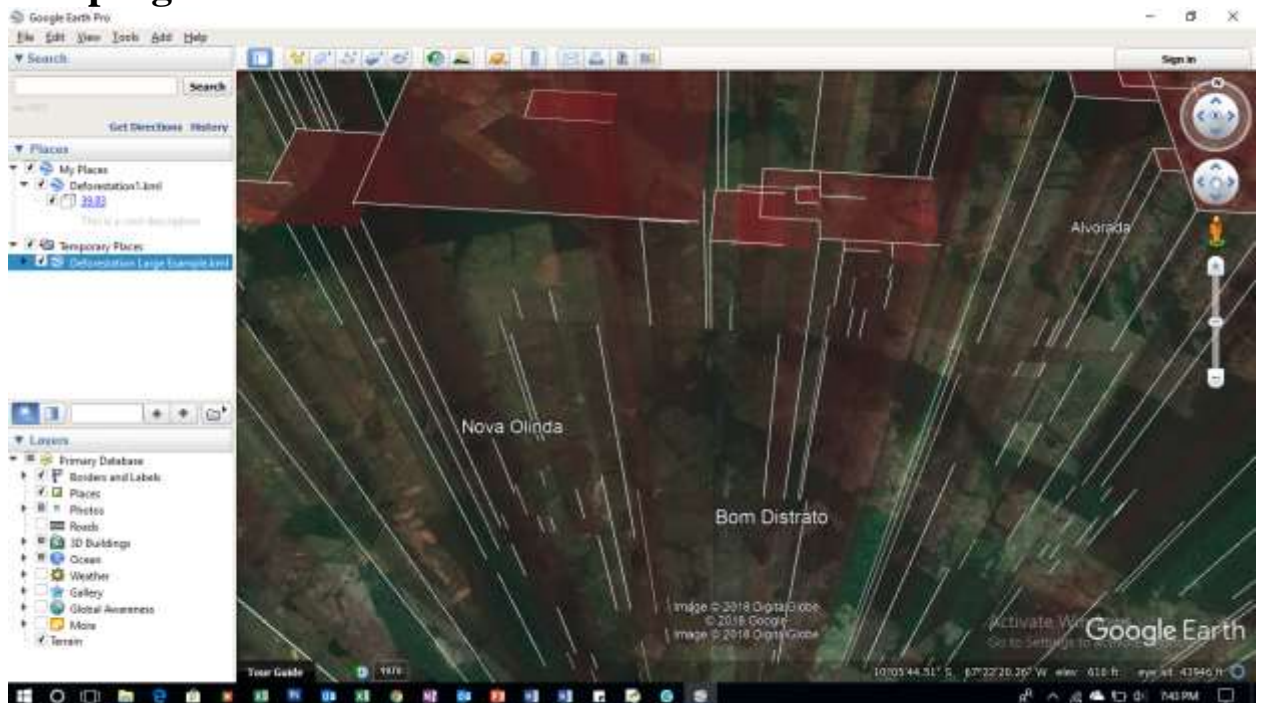


Later we decided to show the area when anyone can see the area when they open Google Earth. So we use an <extrude> tag which specifies whether to connect the Linear Ring to the ground.

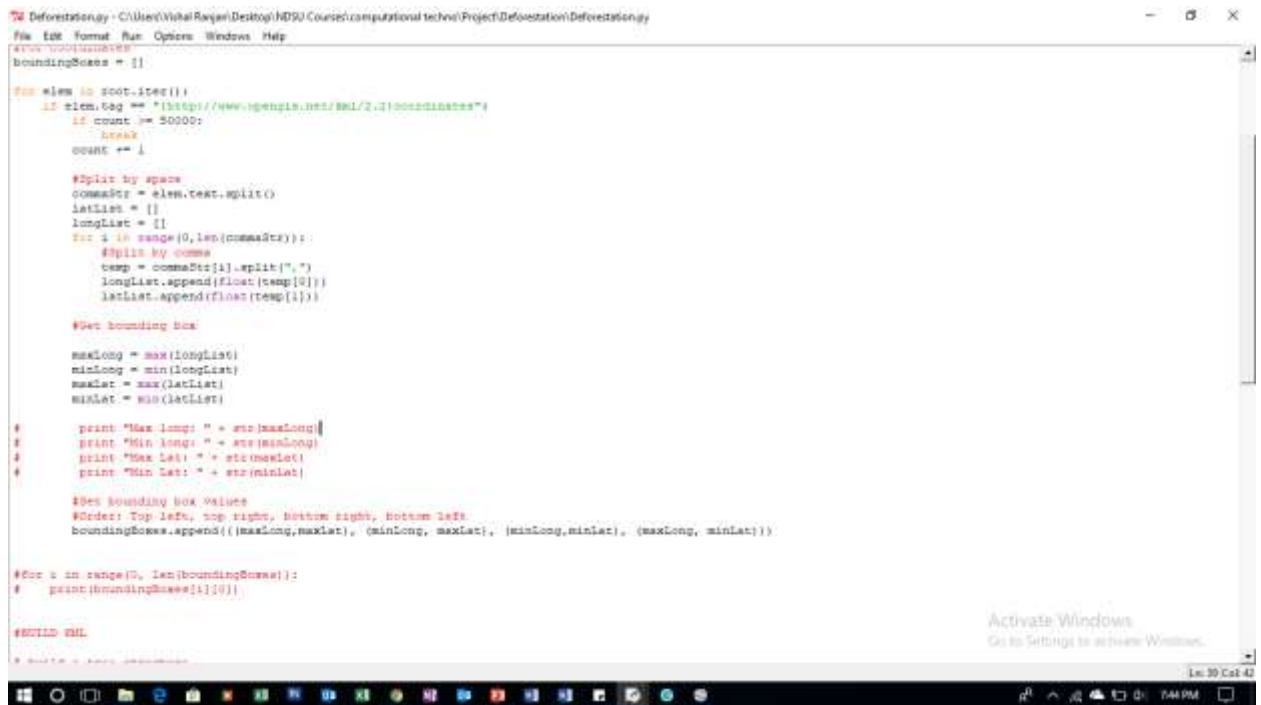
So after opening the Google Earth, the Google Earth is focused on Brazil and the program looks like this



The final program looks like:-



Sample Python File:-



```
Deforestation.py - C:\Users\Vishal Rajan\Desktop\HDSU Courses\computational techno\Project\Deforestation\Deforestation.py
File Edit Format Run Options Windows Help

#Get bounding boxes
boundingBoxes = []

for elem in root.iter():
    if elem.tag == "http://www.opengis.net/ML/2.1/coordinates":
        if count >= 50000:
            break
        count += 1

    #Split by space
    commaStr = elem.text.split()
    latList = []
    longList = []
    for i in range(0, len(commaStr)):
        #Split by comma
        temp = commaStr[i].split(",")
        longList.append(float(temp[0]))
        latList.append(float(temp[1]))

    #Get bounding box
    maxLong = max(longList)
    minLong = min(longList)
    maxLat = max(latList)
    minLat = min(latList)

    print "Max Long: " + str(maxLong)
    print "Min Long: " + str(minLong)
    print "Max Lat: " + str(maxLat)
    print "Min Lat: " + str(minLat)

    #Get bounding box values
    #Order: Top left, top right, Bottom right, Bottom Left
    boundingBoxes.append((maxLong, maxLat), (minLong, maxLat), (minLong, minLat), (maxLong, minLat))

for i in range(0, len(boundingBoxes)):
    print(boundingBoxes[i][0])

#END OF FILE
```

Conclusion:

We tried to show how humans are affecting the Earth by just adding place marks. We tried to show how many places are being deforested in Brazil which can lead to global warming. To show the effect of global warming we tried to add place marks where the temperature is high and where the temp is low. In addition to this, we showed the species which are on the verge of becoming extinct. We tagged the place where the species are found and provide a website link so that people can read about the species and try to stop things which can be harmful to the species.