**Gps track py:**

|  |
| --- |
| import pandas as pd |
|  |

|  |
| --- |
| import numpy as np |
|  |

|  |
| --- |
| import matplotlib.pyplot as plt |
|  |

|  |
| --- |
| from PIL import Image, ImageDraw |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| data\_path = 'data.csv' |
|  |

|  |
| --- |
| data = pd.read\_csv(data\_path, names=['LATITUDE', 'LONGITUDE'], sep=',') |
|  |

|  |
| --- |
| gps\_data = tuple(zip(data['LATITUDE'].values, data['LONGITUDE'].values)) |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| image = Image.open('map.png', 'r') # Load map image. |
|  |

|  |
| --- |
| img\_points = [] |
|  |

|  |
| --- |
| for d in gps\_data: |
|  |

|  |
| --- |
| x1, y1 = scale\_to\_img(d, (image.size[0], image.size[1])) # Convert GPS coordinates to image coordinates. |
|  |

|  |
| --- |
| img\_points.append((x1, y1)) |
|  |

|  |
| --- |
| draw = ImageDraw.Draw(image) |
|  |

|  |
| --- |
| draw.line(img\_points, fill=(255, 0, 0), width=2) # Draw converted records to the map image. |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| image.save('resultMap.png') |
|  |

|  |
| --- |
| x\_ticks = map(lambda x: round(x, 4), np.linspace(lon1, lon2, num=7)) |
|  |

|  |
| --- |
| y\_ticks = map(lambda x: round(x, 4), np.linspace(lat1, lat2, num=8)) |
|  |

|  |
| --- |
| y\_ticks = sorted(y\_ticks, reverse=True) # y ticks must be reversed due to conversion to image coordinates. |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| fig, axis1 = plt.subplots(figsize=(10, 10)) |
|  |

|  |
| --- |
| axis1.imshow(plt.imread('resultMap.png')) # Load the image to matplotlib plot. |
|  |

|  |
| --- |
| axis1.set\_xlabel('Longitude') |
|  |

|  |
| --- |
| axis1.set\_ylabel('Latitude') |
|  |

|  |
| --- |
| axis1.set\_xticklabels(x\_ticks) |
|  |

|  |
| --- |
| axis1.set\_yticklabels(y\_ticks) |
|  |

|  |
| --- |
| axis1.grid() |
|  |

plt.show()