

```

import cv2
from mpl_toolkits.basemap import Basemap

import os
import glob
img_dir = "../input/uni-campus-dataset/RGB-img" # Enter Directory of all images
Category = ['img']
for category in Category:
    folder = os.path.join(img_dir,category)

for files in os.listdir(folder):
    img_path = os.path.join(folder,files)

```

```

import exifread
from PIL import Image
from PIL.ExifTags import TAGS, GPSTAGS
import pandas as pd

def get_exif_data(image):
    """Returns a dictionary from the exif data of an PIL Image item. Also converts the GPS Tags"""
    exif_data = {}
    info = image._getexif()
    if info:
        for tag, value in info.items():
            decoded = TAGS.get(tag, tag)
            if decoded == "GPSInfo":
                gps_data = {}
                for t in value:
                    sub_decoded = GPSTAGS.get(t, t)
                    gps_data[sub_decoded] = value[t]

                exif_data[decoded] = gps_data
            else:
                exif_data[decoded] = value

    return exif_data

def _get_if_exist(data, key):
    if key in data:
        return data[key]

    return None

def _convert_to_degress(value):
    """Helper function to convert the GPS coordinates stored in the EXIF to degress in float format"""
    d0 = value[0]

    d = float(d0)

    m1 = value[1]
    m = float(m1)

    s0 = value[2]
    s = float(s0)

    return d + (m / 60.0) + (s / 3600.0)

def get_lat_lon(exif_data):
    """Returns the latitude and longitude, if available, from the provided exif_data (obtained through get_exif_data)"""
    lat = None
    lon = None

    if "GPSInfo" in exif_data:
        gps_info = exif_data["GPSInfo"]

        gps_latitude = _get_if_exist(gps_info, "GPSLatitude")
        gps_latitude_ref = _get_if_exist(gps_info, 'GPSLatitudeRef')
        gps_longitude = _get_if_exist(gps_info, 'GPSLongitude')
        gps_longitude_ref = _get_if_exist(gps_info, 'GPSLongitudeRef')

```

```

    if gps_latitude and gps_latitude_ref and gps_longitude and gps_longitude_ref:
        lat = _convert_to_degress(gps_latitude)
        if gps_latitude_ref != "N":
            lat = 0 - lat

        lon = _convert_to_degress(gps_longitude)
        if gps_longitude_ref != "E":
            lon = 0 - lon

    return lon,lat

#####
# Example #####
#####
if __name__ == "__main__":
    img_dir = "../input/uni-campus-dataset/RGB-img" # Enter Directory of all images
    Category = ['img']
    for category in Category:
        folder = os.path.join(img_dir,category)

        for files in os.listdir(folder):
            data = []
            img_path = os.path.join(folder,files)
            image = Image.open(img_path)
            exif_data = get_exif_data(image)
            (lon,lat) = get_lat_lon(exif_data)
            print(lat)

```

```

14.073395685277777
14.067295146944444
14.066033934722222
14.067816876111111
14.073615746666666
14.068714821666667
14.067315155833333
14.070374245555556
14.066079850277779
14.069061996111111
14.065617676666667
14.076382165833333
14.065556144166667
14.0716736525
14.076558371666666
14.0730685475
14.074675351111111
14.0744416875
14.06999668611111
14.073803905833334
14.074819631666667

```

14.075748374722222  
14.073501125  
14.072125149722222  
14.06825170361111  
14.071696830555556  
14.073372526944445  
14.075596137222222  
14.073519987777777  
14.074927719444444  
14.06690060138889  
14.07119514138889  
14.06483170138889  
14.071757514722222  
14.06826166  
14.072542919166667  
14.074432080555555  
14.072731221944444  
14.069804981388888  
14.075677443055556  
14.072887901666666  
14.066494802222223  
14.070473504999999  
14.074882087499999  
14.070377882222221  
14.06812354388889  
14.066456868055557  
14.066872836666667  
14.075550119999999  
14.067910681944443  
14.070431859166666  
14.06555201527778  
14.070859  
14.069208139444445  
14.0761281775  
14.065652469166668  
14.071565122777777  
14.07217762638889  
14.07374881111111  
14.074735128611112  
14.065141133611112  
14.0704546675  
14.068337470833333  
14.070757575277778  
14.0735513975  
14.070026666944445  
14.070801319722221  
14.065611943055556  
14.065863849444446  
14.06771304  
14.069583486944444  
14.076209597222222  
14.066945713055555  
14.0739962175  
14.068246753055556  
14.071690227777777  
14.065928587222222  
14.073841130277778  
14.066876255277778  
14.07257915111111  
14.074777709166666  
14.071652068611112  
14.073031058055555  
14.072995874444445  
14.074025648333333  
14.076093186388889  
14.07703792861111  
14.066803030555555  
14.075694565  
14.0756543625  
14.072285802222222  
14.064777432500001  
14.074382155  
14.072441728333333  
14.07475166361111  
14.067820470555555  
14.073435035555555  
14.07036391

14.067750812222222  
14.069897461388889  
14.076191377499999  
14.066960480277778  
14.075206482777778  
14.071330247777778  
14.071741647222222  
14.069068002222222  
14.067820155555555  
14.069145115277777  
14.075647957777777  
14.065485060277778  
14.069443406666666  
14.067682832222221  
14.073873807777778  
14.075108305  
14.066407844444445  
14.072062614722222  
14.075939321111111  
14.069926449722223  
14.067749432222222  
14.068584089722222  
14.06596895388889  
14.074052093055556  
14.074726896666666  
14.073850517222223  
14.072234441944444  
14.075630179166666  
14.06783731638889  
14.074314434444444  
14.072596050555555  
14.065940341111112  
14.069123903055555  
14.076423908611111  
14.06609690027778  
14.06762176888889  
14.06913346888889  
14.067775468055554  
14.070444974722221  
14.068492327777777  
14.065118730833333  
14.065146186111113  
14.069449800833333  
14.065989862222223  
14.065212633055557  
14.06904802  
14.075247661111112  
14.071234716666666  
14.071767549166667  
14.067402578888888  
14.066361074444446  
14.071216831666666  
14.0760359575  
14.069150490277778  
14.071334398333333  
14.076492083055555  
14.071800628333333  
14.076520399444444  
14.068773853888889  
14.073059708333334  
14.073413536666667  
14.064407653333333  
14.070336481944445  
14.070845935277777  
14.070816574166667  
14.071252876388888  
14.070022320833333  
14.076942396666666  
14.074806746388889  
14.072165958611111  
14.07531397638889  
14.06868309611111  
14.066744338333333  
14.07506192888889  
14.072193199444444  
14.072133985555555  
14.076998436944445

14.071683336666666  
14.068189390277778  
14.067374718888889  
14.076059890833333  
14.0713362975  
14.068269825277778  
14.069573510833333  
14.072621399166666  
14.065183168888889  
14.069557803333334  
14.075691736666666  
14.066439706111112  
14.072639405833334  
14.06892376361111  
14.066896410555556  
14.069654617222222  
14.0729678775  
14.068604926666666  
14.075294928333333  
14.070879702222221  
14.067182051944444  
14.06900025611111  
14.067678106944443  
14.068691598888888  
14.075190969166666  
14.065059176666667  
14.073012493888889  
14.06905538  
14.073318286944444  
14.064708865555556  
14.073985230555556  
14.075194726388888  
14.06950355138889  
14.074874598888888  
14.065112730277779  
14.074282001666667  
14.071357811111112  
14.068621044166667  
14.074832696666666  
14.064623788611112  
14.069876444444445  
14.072566659722222  
14.066299236944445  
14.067026234444445  
14.066369948611111  
14.065198174444445  
14.06731641861111  
14.0669352725  
14.071759103888889  
14.073080311666667  
14.076170531388888  
14.066064360277778  
14.068151633333333  
14.066434939166667  
14.0760847275  
14.075157653055555  
14.070017194722222  
14.07207634111111  
14.067845574166666  
14.073000002222223  
14.073470200833333  
14.068702429722222  
14.073509336944444  
14.06645954  
14.065564323333334  
14.0712820175  
14.074762908888889  
14.075987154444444  
14.070453432777777  
14.071256883333334  
14.074241591666667  
14.076684361944444  
14.069942412222222  
14.074316985  
14.072636516944444  
14.065993978611111  
14.066796007777777

14.075361444722223  
14.065178526666667  
14.066529934722222  
14.068056115277777  
14.070315305  
14.068655677222221  
14.074281368055555  
14.068279948333332  
14.065211677222223  
14.07693437388889  
14.070893321944444  
14.066840736111111  
14.069159240555555  
14.074629367777778  
14.070811162777778  
14.0758094325  
14.070252322222222  
14.073433875277777  
14.075617955  
14.073500082499999  
14.072955736666666  
14.069942991388888  
14.074345139444445  
14.064698528333334  
14.067332426111111  
14.0657170975  
14.067725866388889  
14.076132981666666  
14.07074961388889  
14.067396176388888  
14.069486695  
14.066954414444444  
14.069582809722222  
14.071325698055555  
14.069503378333334  
14.073950731666667  
14.068724506111112  
14.072197251111112  
14.075218428055555  
14.073956756388888  
14.065585486944444  
14.072974101111111  
14.066152820000001  
14.075322974722223  
14.076068084722221  
14.074886141944445  
14.069593443333334  
14.071194155833332  
14.072636127222221  
14.072496545833333  
14.070410923611111  
14.073078516944443  
14.067232594444443  
14.074316196944444  
14.073909867222222  
14.072108072499999  
14.072511757777777  
14.074750526388888  
14.067370488611111  
14.06549955138889  
14.0751923425  
14.066961200555555  
14.066587209722224  
14.067807285277777  
14.069940053055555  
14.07264462138889  
14.067243774166666  
14.07184515888889  
14.069368525555555  
14.065998670555556  
14.076526152222222  
14.067279734722222  
14.076852629444444  
14.066513560277778  
14.066015260555556  
14.069875917777777  
14.073164338611111

14.070529790277778  
14.066429130000001  
14.065532057222223  
14.07656442138889  
14.075262720555555  
14.070918694722222  
14.067374276666666  
14.070454230555555  
14.065636791111112  
14.070816420555555  
14.06821620388889  
14.072082433333334  
14.070020579166666  
14.072562504166667  
14.070083869722222  
14.068569786111111  
14.075246122777777  
14.071717694166667  
14.076608329722221  
14.069569238888889  
14.070891365833333  
14.072130921944444  
14.076063255277777  
14.064732609722222  
14.073074477777778  
14.075261921111111  
14.074385203333334  
14.068569354444444  
14.076894803611111  
14.070312968888889  
14.073412426111111  
14.075628200833332  
14.068124862222222  
14.074392345  
14.066859158888889  
14.076235288055555  
14.070007932222222  
14.076465929166666  
14.072669703888888  
14.076121709166667  
14.064686510277778  
14.076813914166667  
14.074490426666667  
14.069020327777778  
14.073879195  
14.064983298888889  
14.076995597222222  
14.074450593888889  
14.065277711944445  
14.065653049444444  
14.070893896388888  
14.068182701111111  
14.068623251111111  
14.072205256111111  
14.071633468055555  
14.070768560277777  
14.072008618055555  
14.066519655555556  
14.069096052777777  
14.073925393055555  
14.073441455277777  
14.06953367388889  
14.067343481666667  
14.0743709875  
14.068694614166667  
14.071244609722221  
14.075737523333332  
14.073933853055555  
14.073875728611111  
14.0681670475  
14.074190449166666  
14.070690391666666  
14.076129800277778  
14.066515741666667  
14.067475043611111  
14.065422692222223  
14.065593513611113

```
14.000000000000000
14.071765259722222
14.069500344166666
14.06738730388889
14.071130331944444
14.070971196944445
14.074814933333332
14.072212920555556
14.069458940277778
14.072523029722221
14.072937372777778
14.068620645833333
14.069979235833333
14.064780804722224
14.07665553388889
14.070384186111111
14.066942925555555
14.064749588611113
14.065130389444445
14.075506999166667
14.076504796944445
14.075771819444444
14.071627879166666
14.065095706666668
14.068283371111111
14.076518168611111
14.066076245555557
14.066060646111112
14.0650635375
14.073107039722222
14.071408838333333
14.077083106388889
14.073515816666667
14.07569827388889
14.068185143333332
14.069005046666666
14.069132299722222
14.071297774444444
```

In [ ]:

data

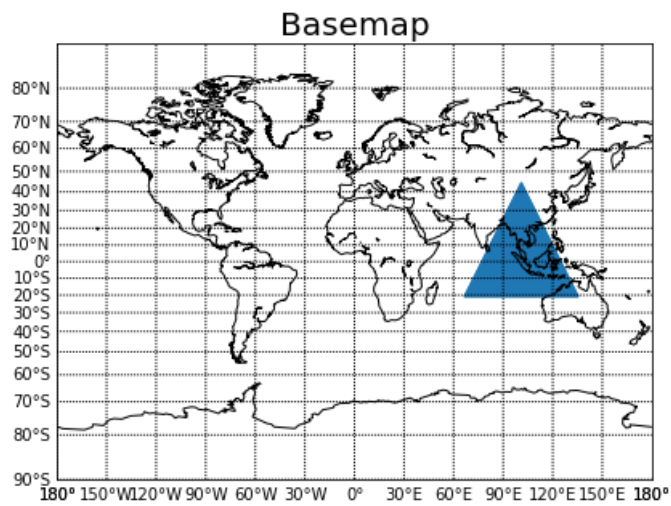
In [111]:

```
from mpl_toolkits.basemap import Basemap
import matplotlib.pyplot as plt
import numpy as np
fig = plt.figure(figsize=(12,5))
m = Basemap(projection='mill',
            llcrnrlat=-90,
            urcrnrlat=90,
            llcrnrlon=-180,
            urcrnrlon=180,
            resolution='c')
m.drawcoastlines()
m.drawparallels(np.arange(-90,90,10), labels=[True,False,False,False])
m.drawmeridians(np.arange(-180,180,30), labels=[0,0,0,1])
lon = [100.61806575638889,100.60708245472222,100.61606427166666], #100.60805817666666,100.61008061944444,1
#,100.61507294472221
#,100.61608368833333
#,100.60506821388888
#,100.61605421083333
#100.61704432944444
#,100.60905121666666
#,100.61403877388888
#,100.60708906638888
#,100.6171229736111
###,100.61708152555555
#,100.61106006222222
#,100.60802811277777
#,100.60303152611111
#,100.61105569777777
, #100.60608742916666
, #100.60807720916667]

lat = [14.073395685277777,14.067295146944444,14.066033934722222] #14.067816876111111,14.073615746666666
```



```
m.scatter(lon,lat,latlon=True,s=5000,marker='^')  
plt.title('Basemap', fontsize=20)  
plt.show()
```



In []: