

## preLab5.py

```

"""
CIVE 6374 - Optical Imaging Metrology
Professor: Dr. Craig Glennie
Author: Joshua Genova
PreLab # 5
Description: Absolute Orientation
Deadline: April 19, 2023 10:00 AM
"""

import numpy as np
import math
from math import sin, cos
from statistics import mean

def rot_mat(W, P, K):
    rot_mat = np.array([
        [cos(P)*cos(K), cos(W)*sin(K)+sin(W)*sin(P)*cos(K), sin(W)*sin(K)-cos(W)*sin(P)*cos(K)],
        [-cos(P)*sin(K), cos(W)*cos(K)-sin(W)*sin(P)*sin(K), sin(W)*cos(K)+cos(W)*sin(P)*sin(K)],
        [sin(P), -sin(W)*cos(P), cos(W)*cos(P)]
    ])

    return rot_mat

def similarity_transform(xc, yc, xf, yf):
    n = len(xc)
    mat_size = 2*n

    # create l-vector
    l_mat = np.zeros(shape=(mat_size,1))
    idx = 0
    for i in range(0, mat_size, 2):
        l_mat[i] = xf[idx]
        l_mat[i+1] = yf[idx]
        idx += 1

    # create A-matrix
    A_mat = np.zeros(shape=(mat_size,4))
    idx = 0
    for i in range(0, mat_size, 2):
        A_mat[i] = [xc[idx], -yc[idx], 1, 0]
        A_mat[i+1] = [yc[idx], xc[idx], 0, 1]
        idx += 1

    # calculate the unknowns x_hat
    x_hat = np.dot(np.dot(np.linalg.inv(np.dot(np.transpose(A_mat),A_mat)), np.transpose(A_mat)),
l_mat)
    A, B, Dx, Dy = float(x_hat[0]), float(x_hat[1]), float(x_hat[2]), float(x_hat[3])
    scale = math.sqrt(A**2 + B**2)
    theta = math.atan(B/A)

    return A, B, Dx, Dy

```

