

# How to georeference an OCAD map

In the modern era it is preferable for OCAD maps to be georeferenced. This can be tricky with old maps that started life in the 1970s and have evolved since then. The following procedure has been used to convert a number of maps and works well.

First some background. OCAD maps have two coordinate systems. The paper system which measures everything in mm from a point 0,0, and optionally a global system such as the British National Grid. In georeferencing a map we are establishing a link between these two systems. The most important point on the map is the point 0,0 in paper coordinates. When we georeference a map we define this point in a global system. The map will probably have been rotated to match magnetic north at some point in the past and it is important to initially remove this rotation. Later we can use the facilities built into OCAD to re-establish the magnetic north correction in a better way. Here is the recipe, however, I recommend that before you follow this procedure you first convert all non map objects such as text and legend to be layer objects as they do not need to be rotated.

1. **Find two grid references on the area.** Go to the web site <https://gridreferencefinder.com/> and find the location of your map. Find two points which you can identify on that web site and on the map. Good points are road junctions, clear boundary corners or the like. Ideally the two points should be as far apart as possible so one in the bottom left corner and one in the top right corner for example. Once you have identified these points, right click on them in the web page and note the two numbers displayed as X(Easting) Y(Northing).

**SD547759**

X (Easting) :

Y (Northing) :

**354752**

**475921**

2. **Move origin to match a grid reference.** In OCAD use map>transform>move to change the point 0,0 to correspond to the first of these points. To do this note the current coordinate of the point and enter the negative of those points into the move map dialog. You can read this from the bottom of the OCAD screen when your mouse is over the location. So for example if the point you are going to use it at 145.6 -55.2, put -145.6 and 55.2 into the dialog. You have now set the origin of the map at an identifiable point.
3. **Georeference the OCAD map.** To do this do map>Set scale and coordinate system. Select real world coordinates and enter the easting and northing values obtained in step 1 for the origin of the map you set in 2. In the coordinate system box choose British National Grid.
4. **Find out how much the map is rotated.** In OCAD move your mouse to the second location identified in step 1. Note the grid reference at the bottom of the screen. We now have the information we need to find out roughly how much the map has been rotated and if the scale is wrong. I have written a python program to do this which is reproduced below along with the formula it is based on. Enter the information into the program or use the formula and it will return how much the maps differ in scale and how much to rotate the map by. If the scale difference is large then the map has the wrong scale set. Adjust this in the “set scale and coordinate system” dialog and then repeat this step. You can also adjust the map using map>transform>stretch/shrink to remove slight differences. If you do this, then recalculate the angle.

5. **Rotate the map.** Go to map>transform>rotate map and rotate the map by the amount suggested by my program. Now go to “Set scale and coordinate system” and set the angle to zero. Check that the second point on the ocad map now has the correct grid reference. If not you may like to repeat the last two steps.
6. **Fine tuning.** You can now open an aerial photo as a background map. See separate instructions on the BOF web site for how to obtain georeferenced maps from Bing or Google. You will almost certainly find things are not quite perfect. You can use the OCAD rubbersheeting command to tweak the map to fit the aerial photograph. This needs care, in particular with areas with a lot of buildings as they can all become distorted. Note. If you decide to rotate the map again, first close all georeferenced background maps then reopen them after you have performed the rotation as they will otherwise be rotated as well which is almost certainly not what you want.
7. **Correct to magnetic north.** Use the menu item map>transform> “rotate map to magnetic north”
8. Save the map with a new filename.

The python program is reproduced below. It works out the distance between the origin and the second point in the two systems. It then works out the distance between the two coordinates for the second point. It takes the arcsin of the ratio of these points to calculate the angle. See diagram below for an explanation

### Python program for calculating angle and rotation:

```
import PySimpleGUI as sg
# import the math module
import math
def calcLength(x1,y1,x2,y2):
    return math.sqrt((float(y2)-float(y1))*(float(y2)-float(y1))+(float(x2)-float(x1))*(float(x2)-float(x1)))
layout = [ [sg.Text('Calculate Rotation For OCAD map')
            ],
            [sg.Text('Origin of OCAD map    x:') , sg.InputText(size=(10,1)) , sg.Text(', y:') , sg.InputText(size=(10,1)) ],
            [sg.Text('2nd point on reference    x:') , sg.InputText(size=(10,1)) , sg.Text(', y:') , sg.InputText(size=(10,1)) ],
            [sg.Text('2nd point on OCAD map    x:') , sg.InputText(size=(10,1)) , sg.Text(', y:') , sg.InputText(size=(10,1)) ],
            [sg.Button('Calculate'), sg.Button('Finished')]
          ]
sg.ChangeLookAndFeel('Green')
sg.SetOptions(font='Helvetica 11')
window = sg.Window('Adjust OCAD map', layout)
while True:
    event, values = window.read()
    if event in (None, 'Finished'): # if user closes window or clicks cancel
        break

    distanceOnReference=calcLength(values[0],values[1],values[2],values[3])
    distanceOnOCAD=calcLength(values[0],values[1],values[4],values[5])
    distanceP2P=calcLength(values[2],values[3],values[4],values[5])

    meanDistance=(distanceOnReference+distanceOnOCAD)/2.0
    ratio=distanceP2P/2.0/meanDistance
    angle=math.degrees(2.0*(math.asin(ratio)))
    scale=distanceOnOCAD/distanceOnReference
    degAng=str(angle)
    text=" Distance on OCAD is "+str(scale)+" times distance on reference \n Rotate OCAD map by "+degAng+" degrees anticlockwise."
    sg.popup(text)
```

This diagram shows the basis of the formula used to calculate the angle to rotate the map.

