Homework 3 -- Space Resection & Intersection

Find the file: photo1_10_hw3.zip which will contain the oblique image file, a matlab script file resect.m, two function scripts lsq_res.m, collin.m, and GCP data.

See the annotated image with approximate point locations, accompanying drawing with point labels, and detailed sketches defining the actual point, then measure all of the points and format them, together with GCP info, into resect.inp (exclude 1016 at this stage). GCP data is UTM zone 16, so convert the given ellipsoid heights, h, to “cartesian” with a delta-z correction as shown in lecture. Use point 4016 as the reference point for this conversion.

For this conversion use R=6371000m

For the camera interior orientation, use: principal point sample=2436, principal point line=1624, focal length=5524.3 (all pixel units). Images were taken by a company which would not give camera calibration data so these values were “reverse engineered” and we have no lens distortion data. Use E~506500, N~4472500, h~1500, (meters) as approximate exposure station. Estimate the angles omega, phi, and kappa. Use this information to populate the cam.inp file. The last number, option, should define the order and meaning of your image coordinates. Be aware that even if your measurements are good, very poor parameter approximations can cause the solution to diverge.

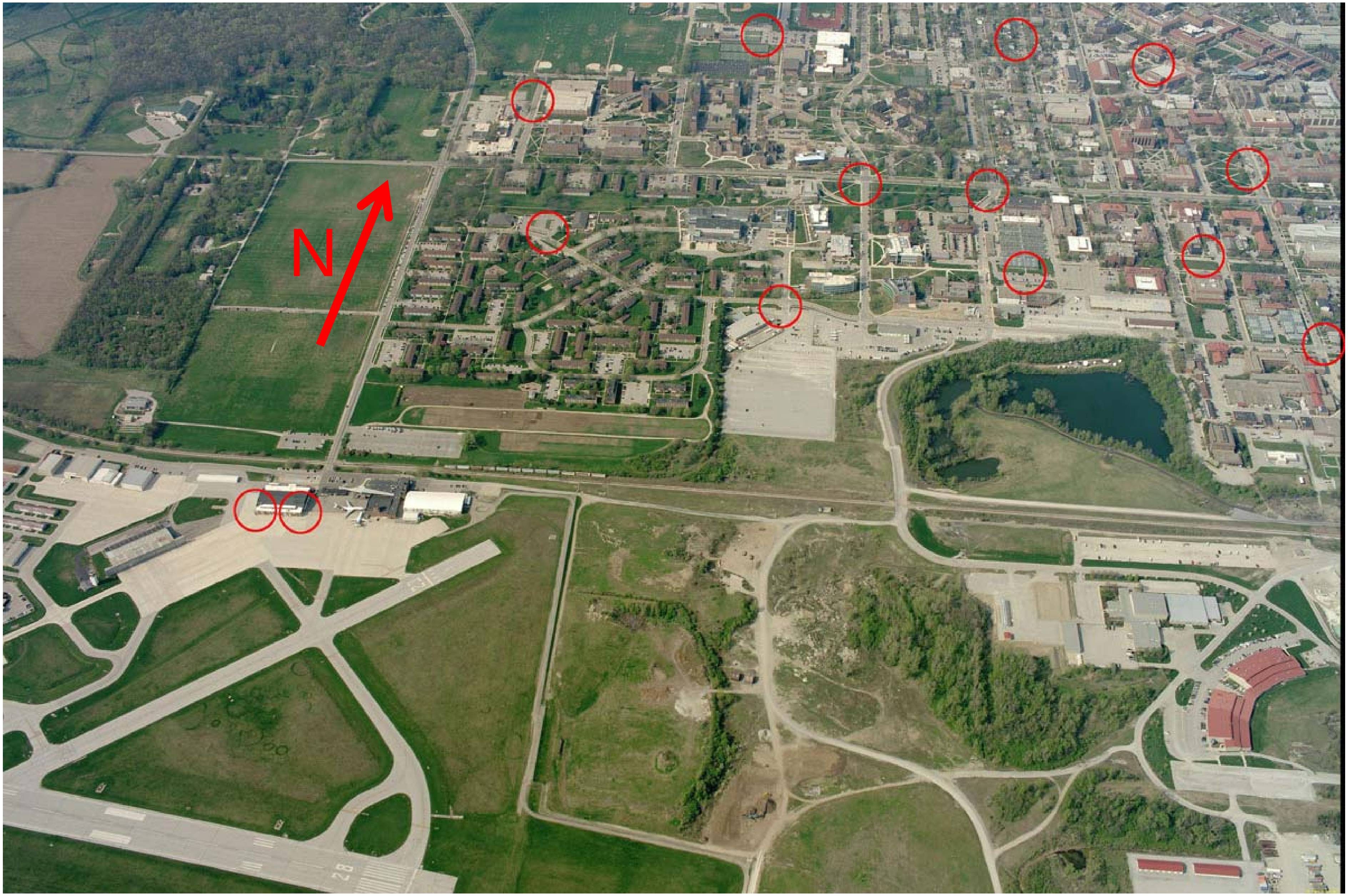
Format your measurements and the GCP data into the file resect.inp. When these two input files are prepared, cam.inp & resect.inp, then run the resect program (data files and scripts should all be in the same folder). Confirm that solution converges nicely (parameter corrections should get very small). Residuals are given in pixel units and should be small, no more than a “few” pixels. The last numbers shown are the final parameters omega, phi, kappa, XL,YL, ZL.

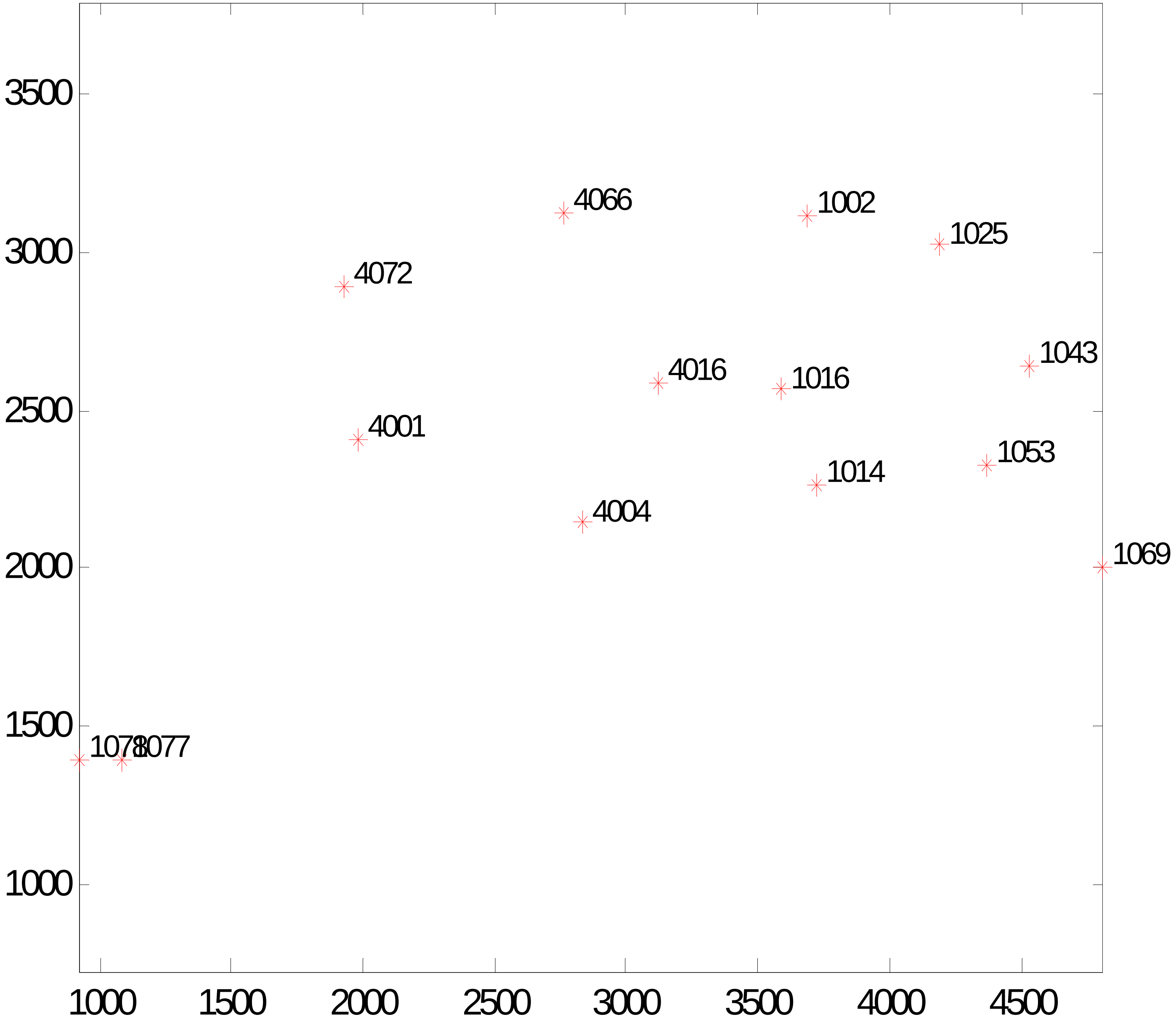
Use the final estimates of exterior orientation of the image together with your measurements of point 1016, and the intersection form of the collinearity equations (given h for the point) to find E,N for point 1016.

Finally, notice that we did not make any correction for atmospheric refraction. Using the formula for K, what would delta-alpha have been for point 1016? How many pixels does that represent in the image (assume it is along the optical axis)? Is that a significant error which we should have corrected?

(Note: in case you are comparing these coordinates with those from HW2, those from HW2 were from a map that had an unfortunately big shift error.)

Due Thursday, 14 Oct., 2010


N


3500
3000
2500
2000
1500
1000
1000
1500
2000
2500
3000
3500
4000
4500
4066
1002
1025
4072
4016
1016
1043
4001
1053
1014
4004
1069
1078
1077

cam.inp

```
omega (radians)
phi (radians)
kappa (radians)
XL (meters)
YL (meters)
ZL (meters)
x0 (pixels)
y0 (pixels)
f (pixels)
option (see below)

==========================================
image coordinate option:
1=conventional x,y cartesian
2=photoshop x,y
3=line,sample

note that the x0,y0 must be in the same order
as the image coordinates
==========================================
```

gcp.txt

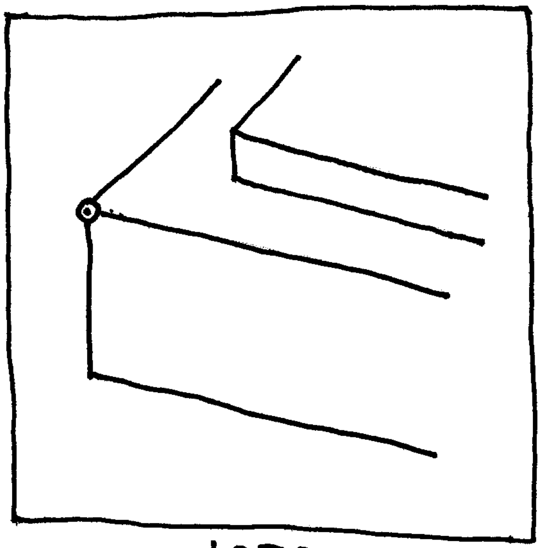
```
1053 507182.684 4474652.620 154.678
4072 506019.942 4475089.265 154.509
4004 506502.951 4474461.737 152.837
4001 506114.115 4474637.873 156.284
1078 505826.347 4473934.451 158.819
1077 505887.377 4473934.848 158.872
1069 507320.990 4474426.703 152.232
4016 506630.763 4474830.353 157.237
4066 506440.060 4475399.675 154.763
1043 507309.038 4474936.626 154.822
1002 506949.044 4475416.655 155.793
1025 507205.922 4475325.697 156.161
1014 506890.328 4474579.625 154.932
1016 ______.___ _______.___ 156.090
```

resect.inp

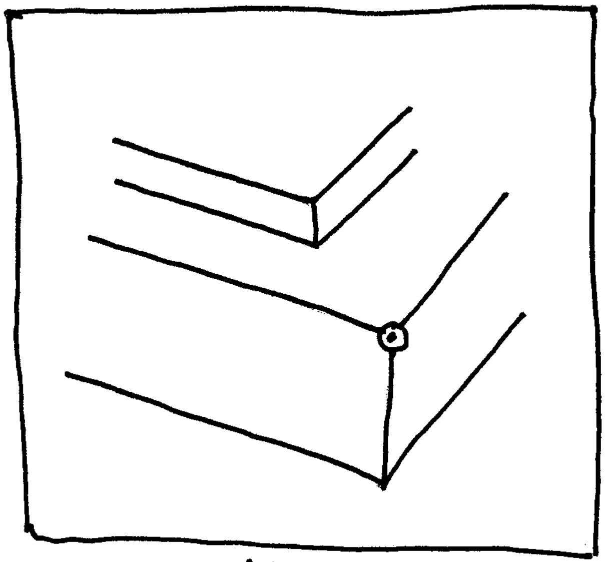
```
point_id  xxx  yyy    EEEEEEE    NNNNNNNN   hhhh
point_id  xxx  yyy    EEEEEEE    NNNNNNNN   hhhh
point_id  xxx  yyy    EEEEEEE    NNNNNNNN   hhhh
point_id  xxx  yyy    EEEEEEE    NNNNNNNN   hhhh
point_id  xxx  yyy    EEEEEEE    NNNNNNNN   hhhh
point_id  xxx  yyy    EEEEEEE    NNNNNNNN   hhhh
point_id  xxx  yyy    EEEEEEE    NNNNNNNN   hhhh
point_id  xxx  yyy    EEEEEEE    NNNNNNNN   hhhh
```

note: image coordinates xxx,yyy have to match definition
option given in cam.inp, use h “converted” to cartesian

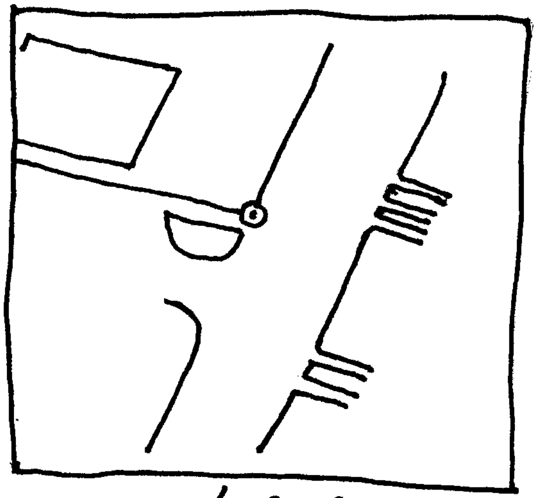
1/3



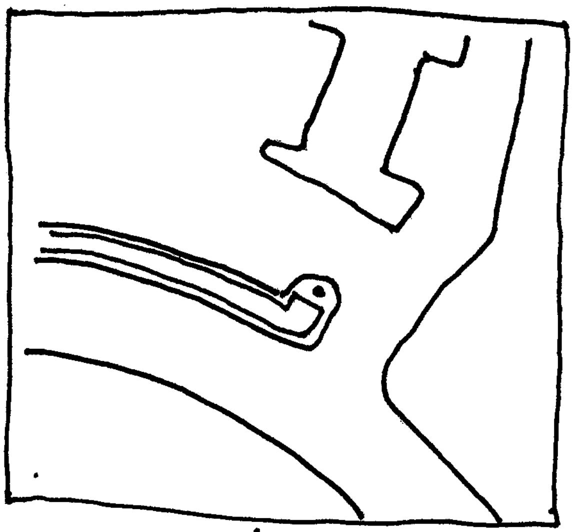
1078
building corner



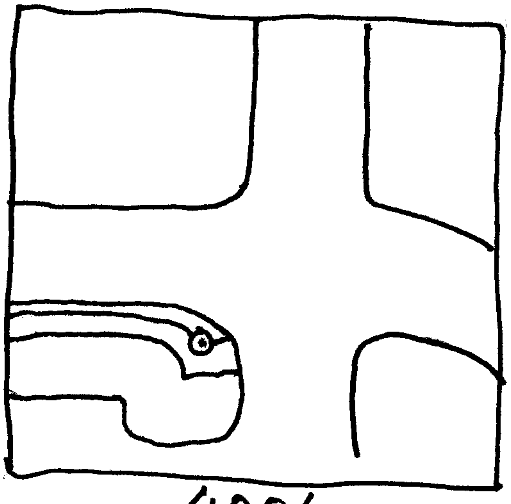
1077
building corner



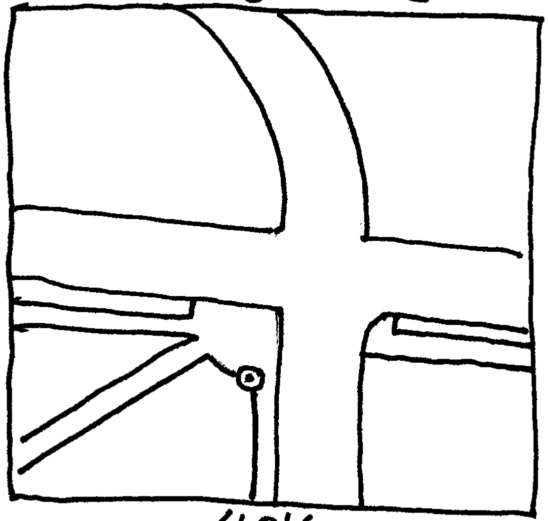
4072
grass corner



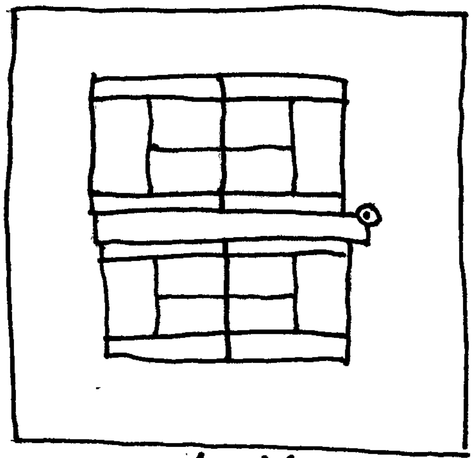
4001
center of semi-circle shape on parking island



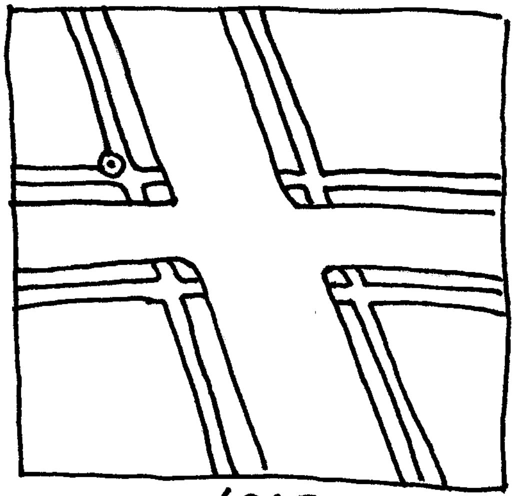
4004
sidewalk corner



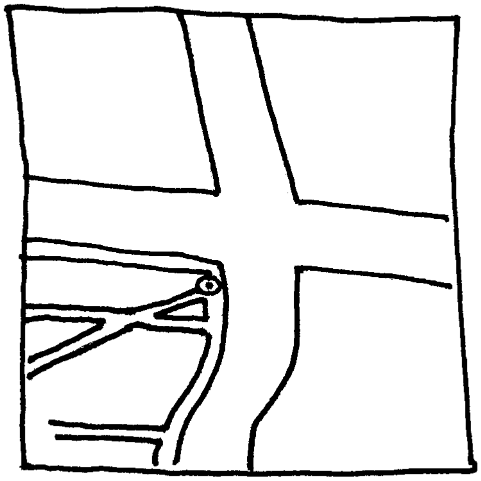
4016
sidewalk corner



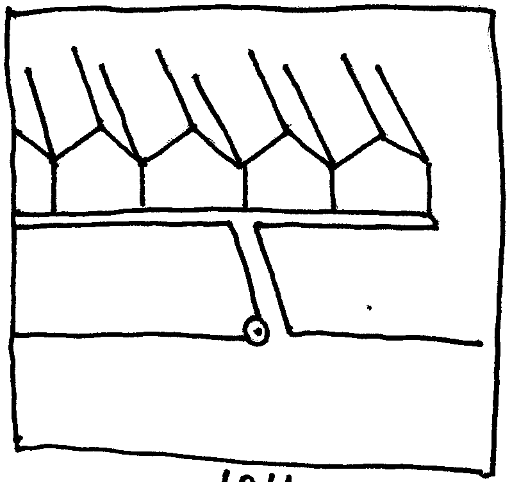
4066
sidewalk corner between
tennis courts



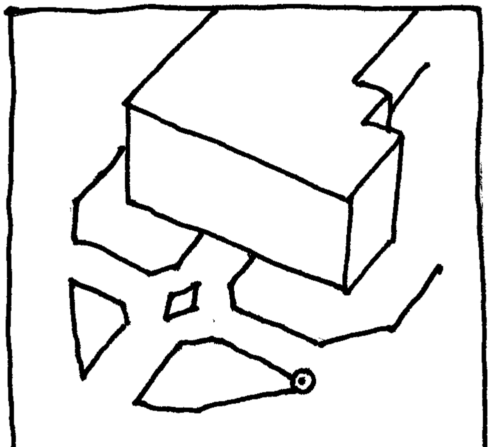
1002
sidewalk corner



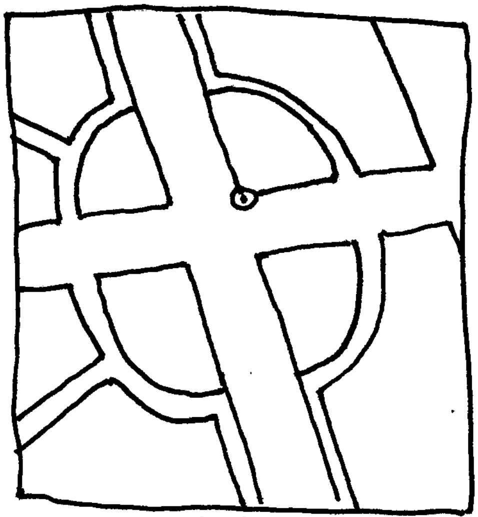
1016
sidewalk corner



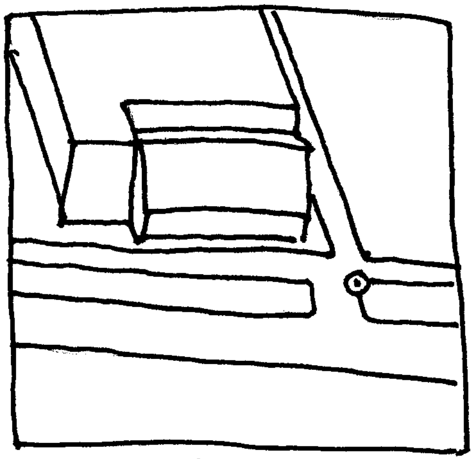
1014
side walk corner



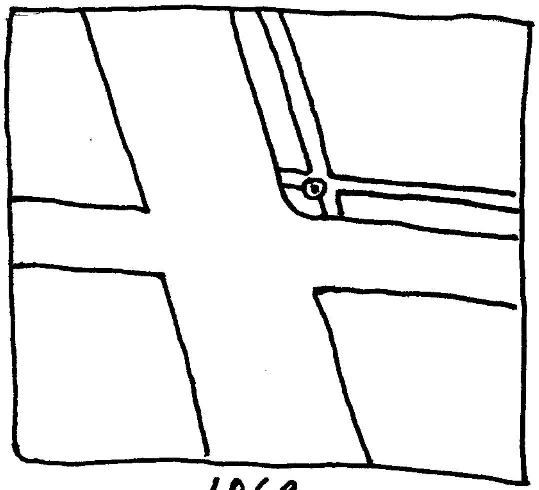
1025
sidewalk corner



1043
brick paving corner



1053
sidewalk corner



1069
sidewalk corner