Project 3: Single View Modeling

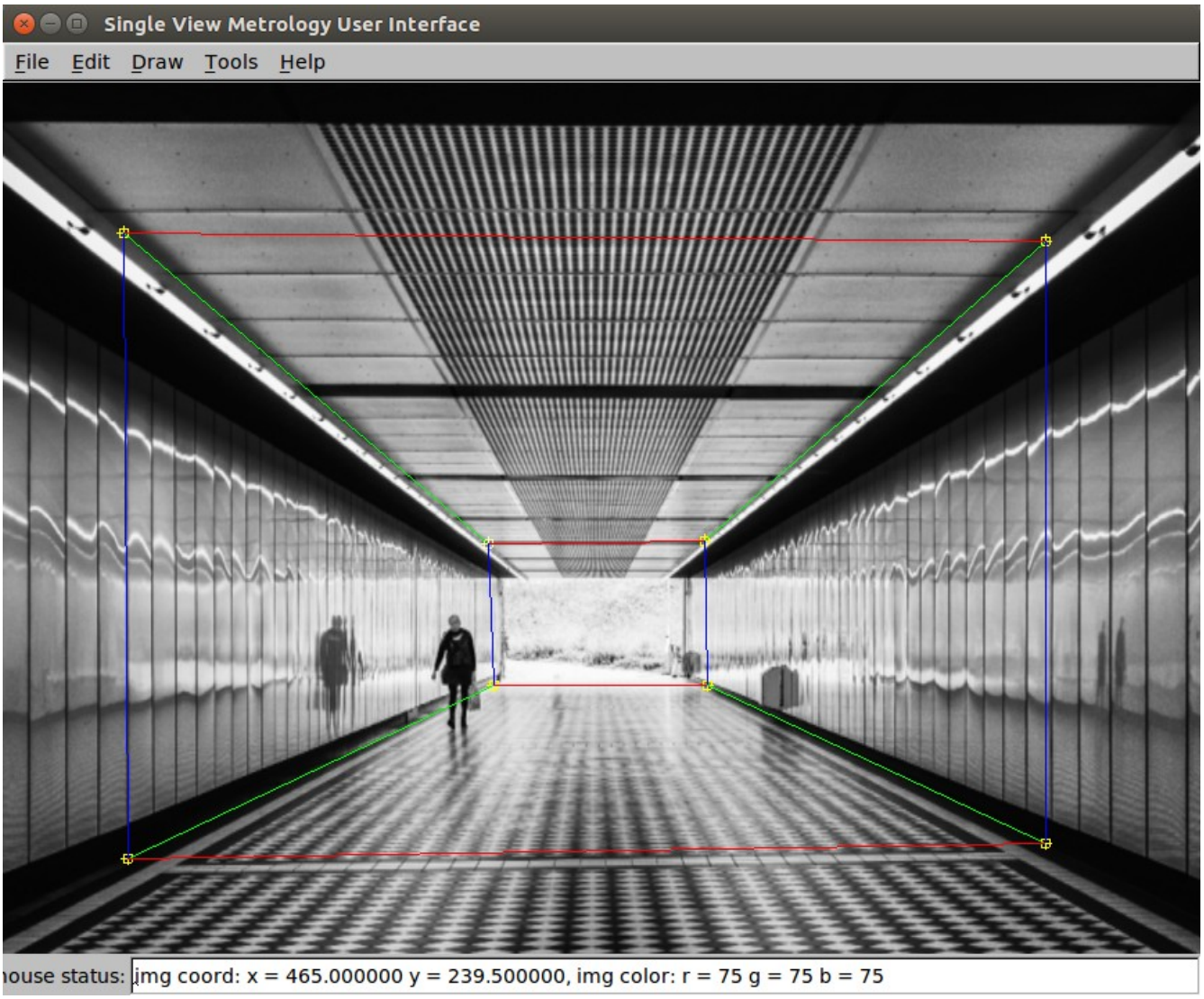
CSE559A - Computer Vision

Daniel de Cordoba Gil

11/13/2016

This image was downloaded from the internet:
https://s3.amazonaws.com/images.hiresphoto.com/along-the-corridor.jpg

Calculating vanishing points:
Once I had at least 2 lines in every direction (x, y z), I could compute VP.



Terminal output after clicking Compute VPs:

```
eigenvector associated with smallest eigenvalue:
0.999717
0.023804
0.000056

eigenvector associated with smallest eigenvalue:
0.891944
0.452144
```

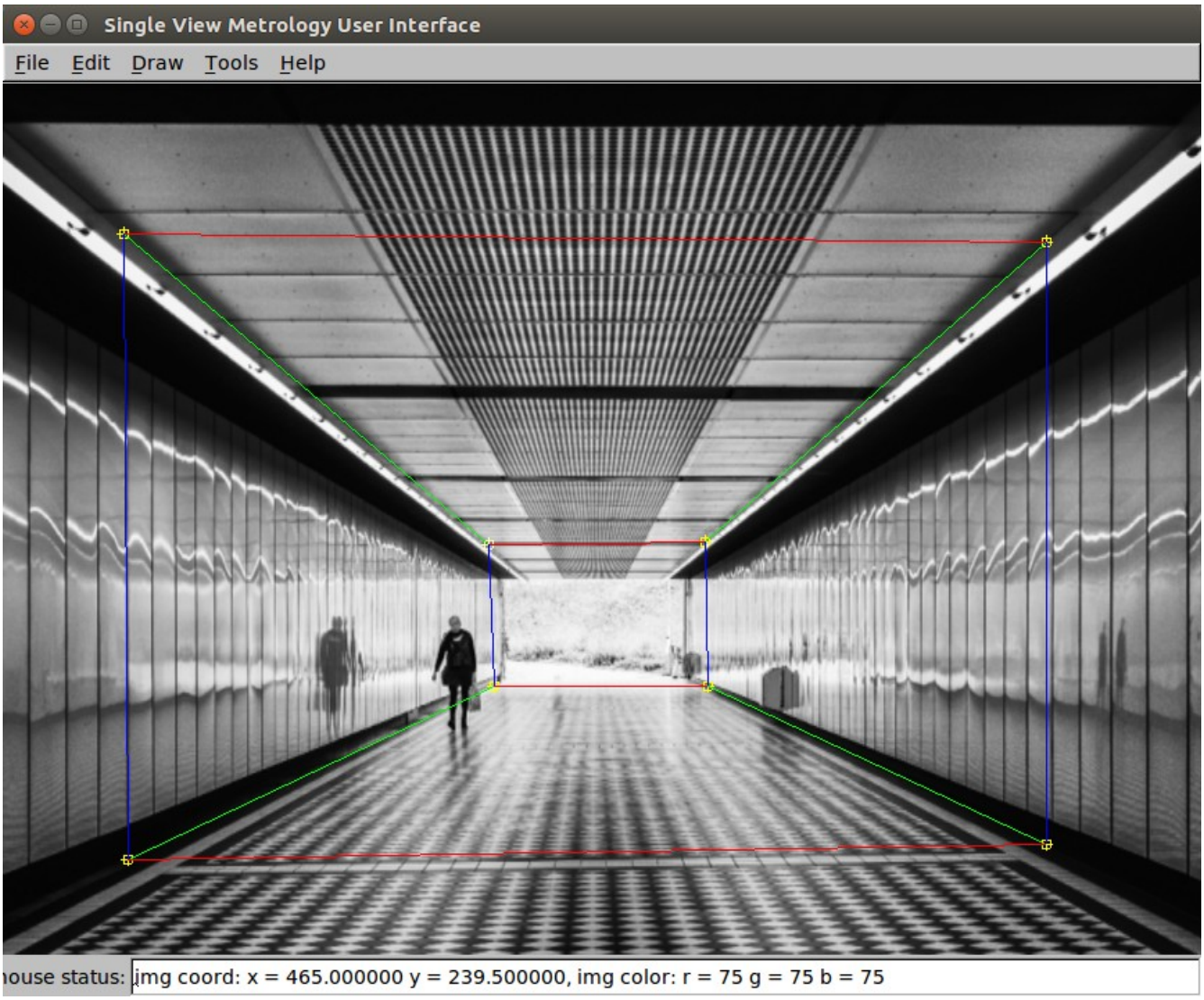
0.001526

eigenvector associated with smallest eigenvalue:
-0.010445
0.999945
-0.000010

VANISHING POINTS
xvp=[17982.613596 428.184395] yvp=[584.563994 296.326974]
zvp=[1042.055885 -99763.132555]

Calculating homographies:

I pushed the points on the base to the stack to calculate the homography.



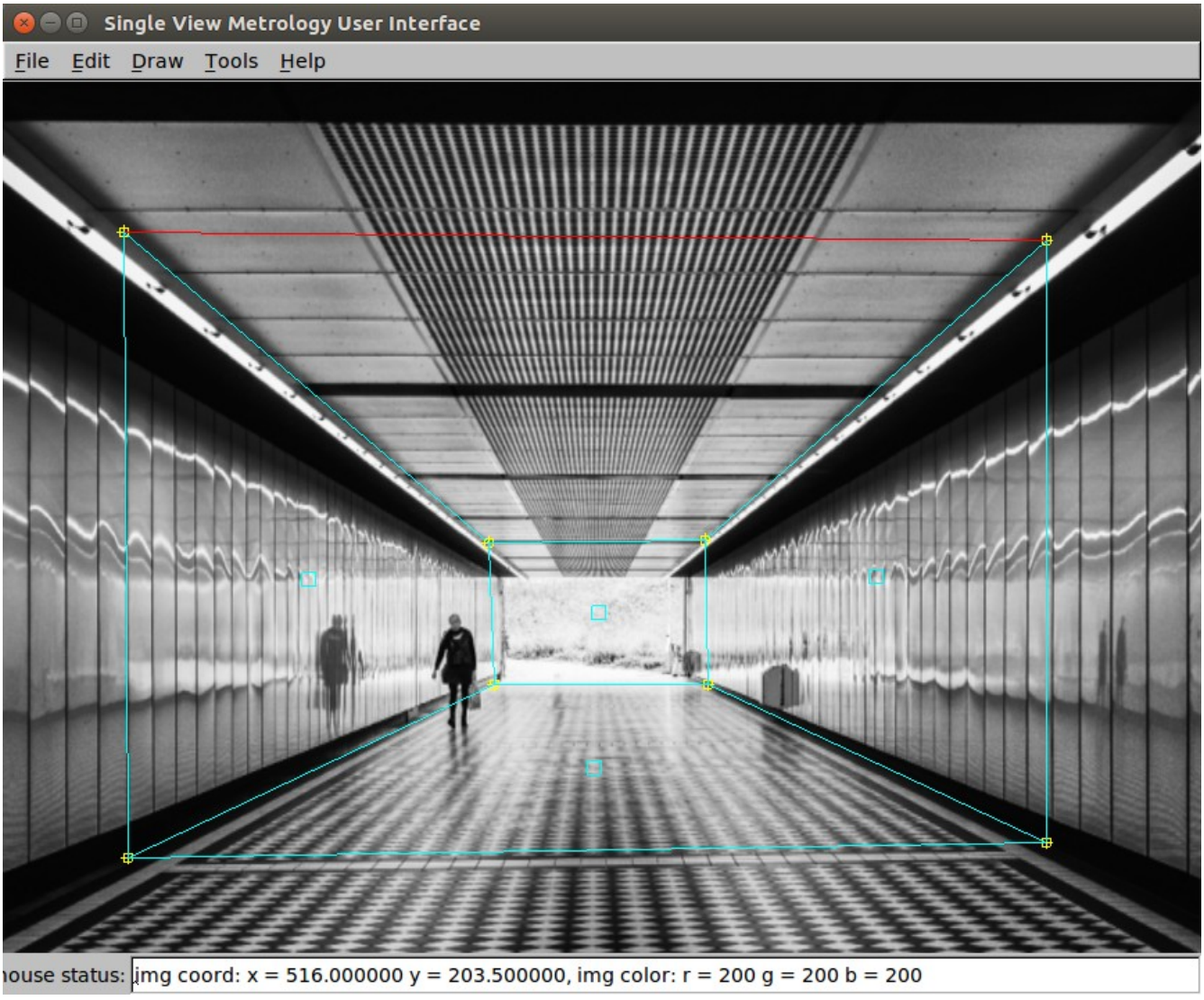
Terminal output after pushing all the points and clicking Compute Homography:

```
pushed point 2168770 onto stack, current size 1
Point information:
     (u, v, w) = (246.000000, 135.500000, 1.000000);
     (X, Y, Z, W) = (0.000000, 0.000000, 0.000000, 1.000000);

pushed point 22d8470 onto stack, current size 2
Point information:
     (u, v, w) = (510.000000, 260.500000, 1.000000);
     (X, Y, Z, W) = (0.000000, 1.000000, 0.000000, 1.000000);

pushed point 2178bc0 onto stack, current size 3
```

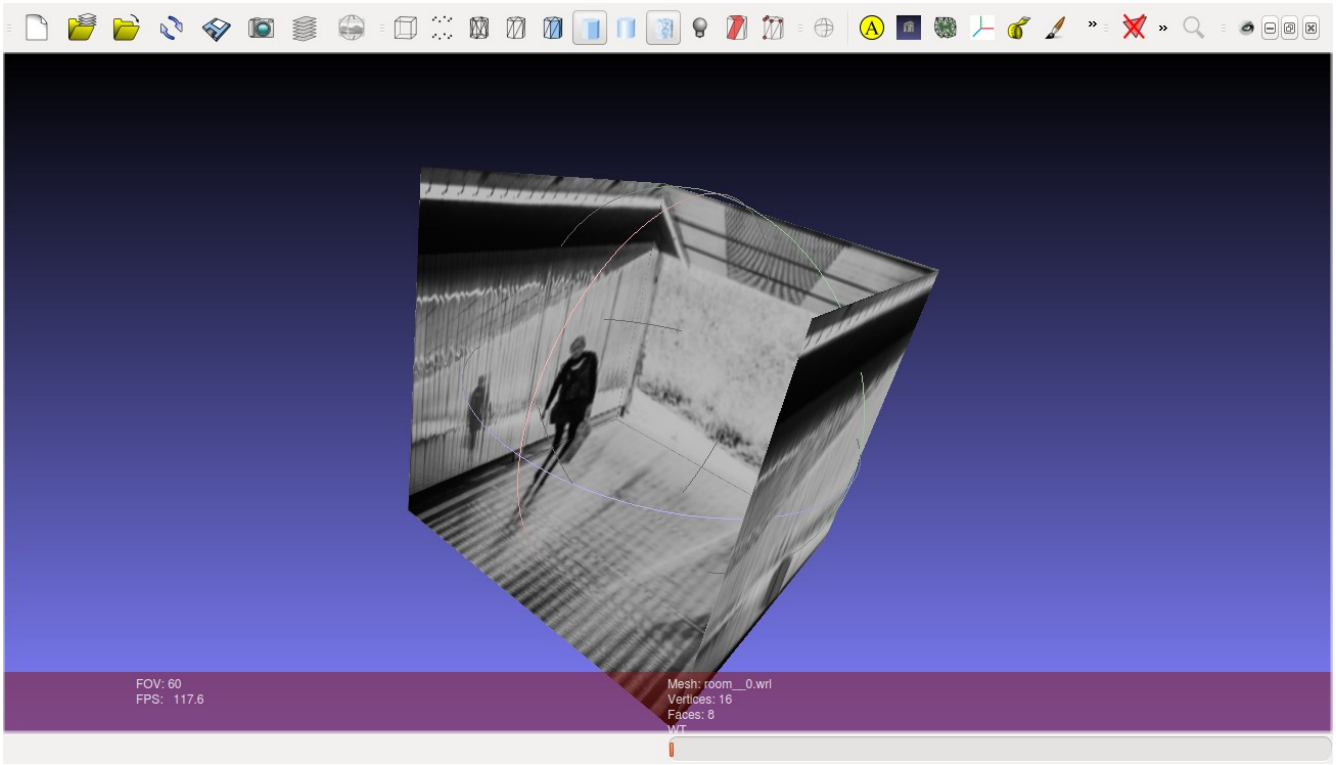
```
Point information:
    (u, v, w) = (908.000000, 146.500000, 1.000000);
    (X, Y, Z, W) = (1.000000, 0.000000, 0.000000, 1.000000);

0: (1000.000000 , 0.000000, 1.000000)
1: (0.000000 , 0.000000, 1.000000)
2: (0.000000 , 1.000000, 1.000000)
3: (1.000000 , 0.000000, 1.000000)
0:  (1000.000000,  0.000000,  1.000000,  0.000000,  0.000000,  0.000000,
-17982613.596152, -0.000000, -17982.613596)
1:  (0.000000,  0.000000,  0.000000,  1000.000000,  0.000000,  1.000000,
-428184.394694, -0.000000, -428.184395)
2:  (0.000000,  0.000000,  1.000000,  0.000000,  0.000000,  0.000000,
-0.000000, -0.000000, -246.000000)
3:  (0.000000,  0.000000,  0.000000,  0.000000,  0.000000,  1.000000,
-0.000000, -0.000000, -135.500000)
4:  (0.000000,  1.000000,  1.000000,  0.000000,  0.000000,  0.000000,
-0.000000, -510.000000, -510.000000)
5:  (0.000000,  0.000000,  0.000000,  0.000000,  1.000000,  1.000000,
-0.000000, -260.500000, -260.500000)
6:  (1.000000,  0.000000,  1.000000,  0.000000,  0.000000,  0.000000,
-908.000000, -0.000000, -908.000000)
7:  (0.000000,  0.000000,  0.000000,  1.000000,  0.000000,  1.000000,
-146.500000, -0.000000, -146.500000)
eigenvector associated with smallest eigenvalue:
0.000000
0.890551
0.000000
0.000000
0.454880
0.000000
0.000000
0.001746
0.000000


H=[
6.962755e+02   1.054733e+15   2.460053e+02;
1.656778e+01   5.387410e+14   1.354874e+02;
```

```
3.773305e-02   2.068104e+12   1.000000e+00;
]
Hinv=[
7.229078e-04   -1.526597e-03  2.899555e-02;
-3.203083e-17  1.920919e-15   -2.523807e-13;
3.896556e-05   -3.915057e-03  1.000000e+00;
]
```

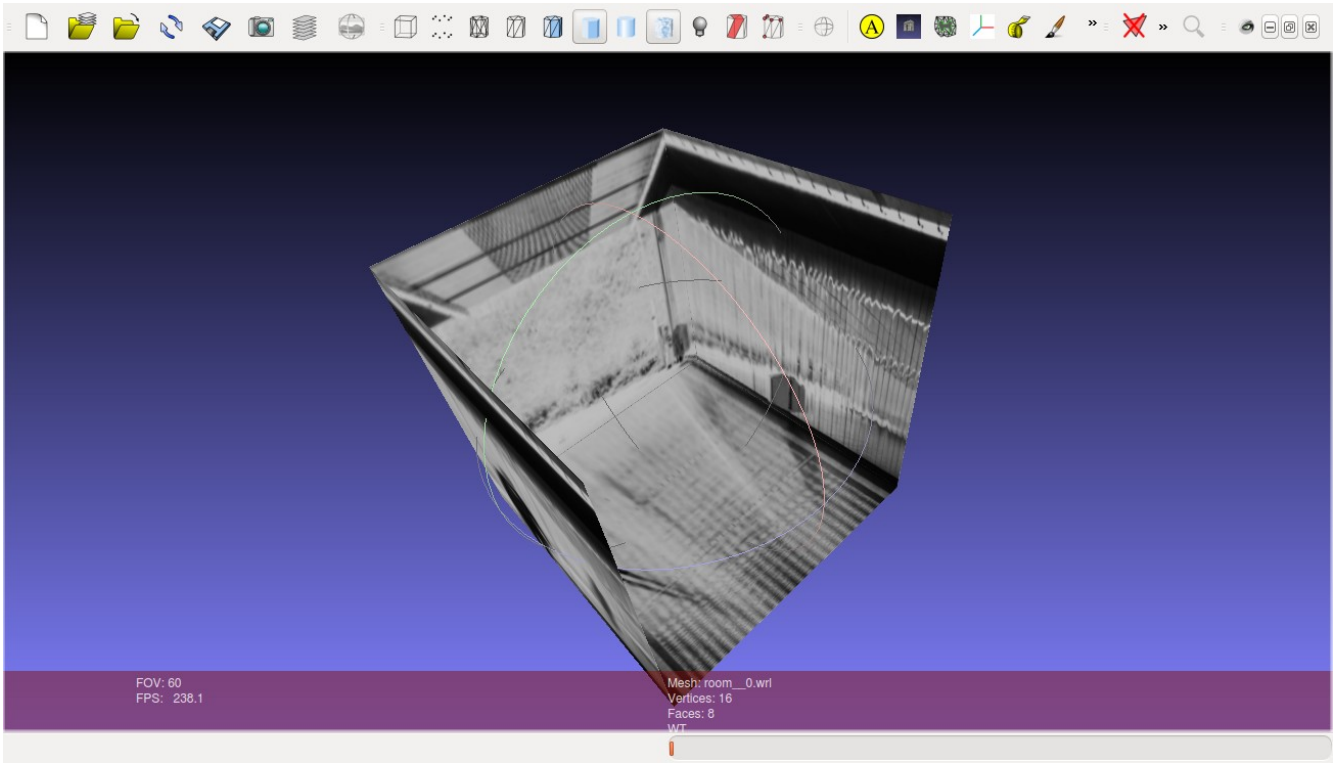
Now, we generate the full model. We can use sameXY to get the coordinates of the points on top, and use sameZ to get the coordinates of the rest of the points. Using ZX Rectangle and SweepRectangle also work. We will finish with the model:



And, after exporting it, we get:



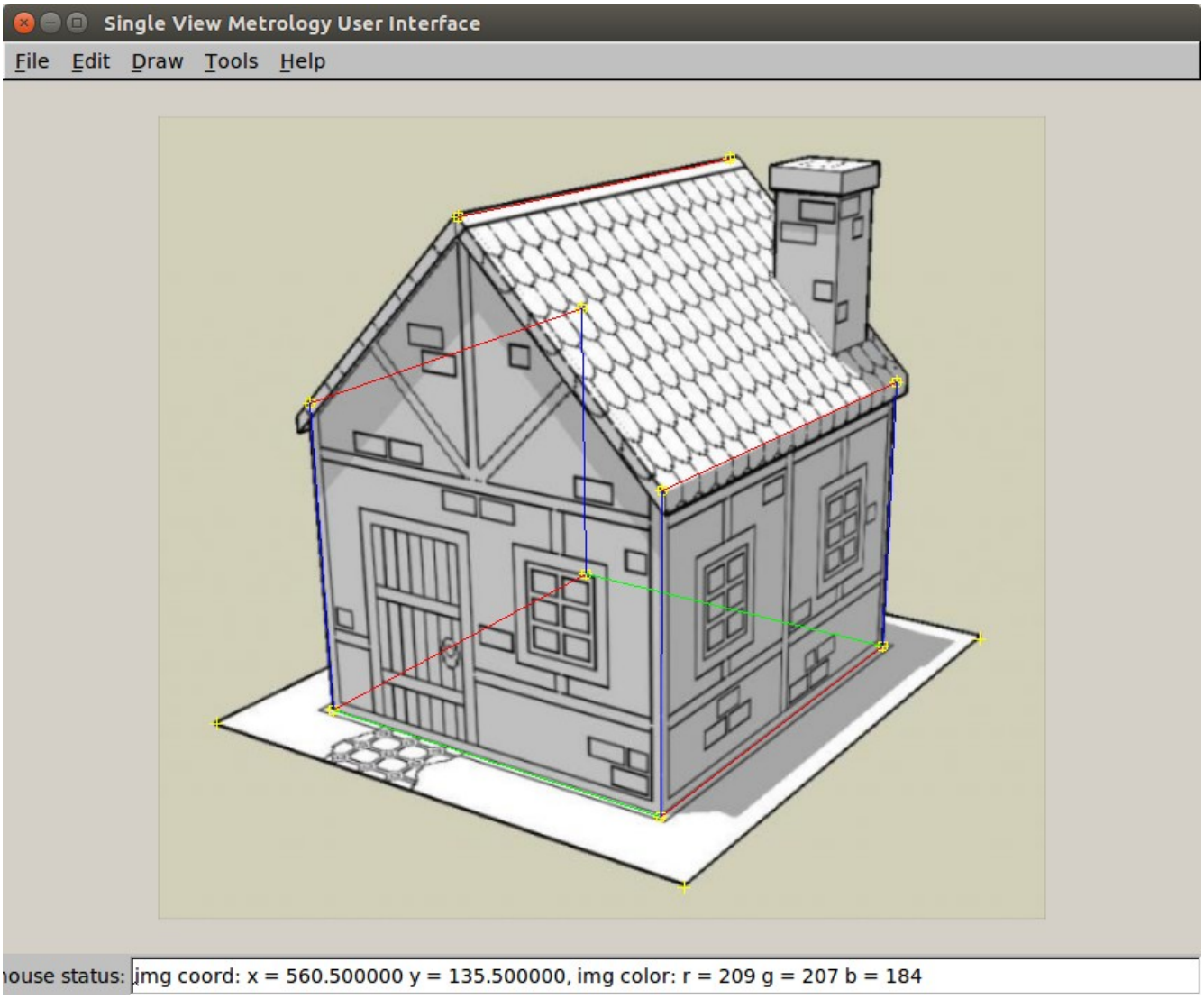
This image was downloaded from the internet:



http://i317.photobucket.com/albums/mm374/mauther/old%20west/2014%20original%20models/simplehaus1sheet001_zpse9e37018.jpg

Calculating vanishing points:

Once I had at least 2 lines in every direction (x, y z), I could compute VP.



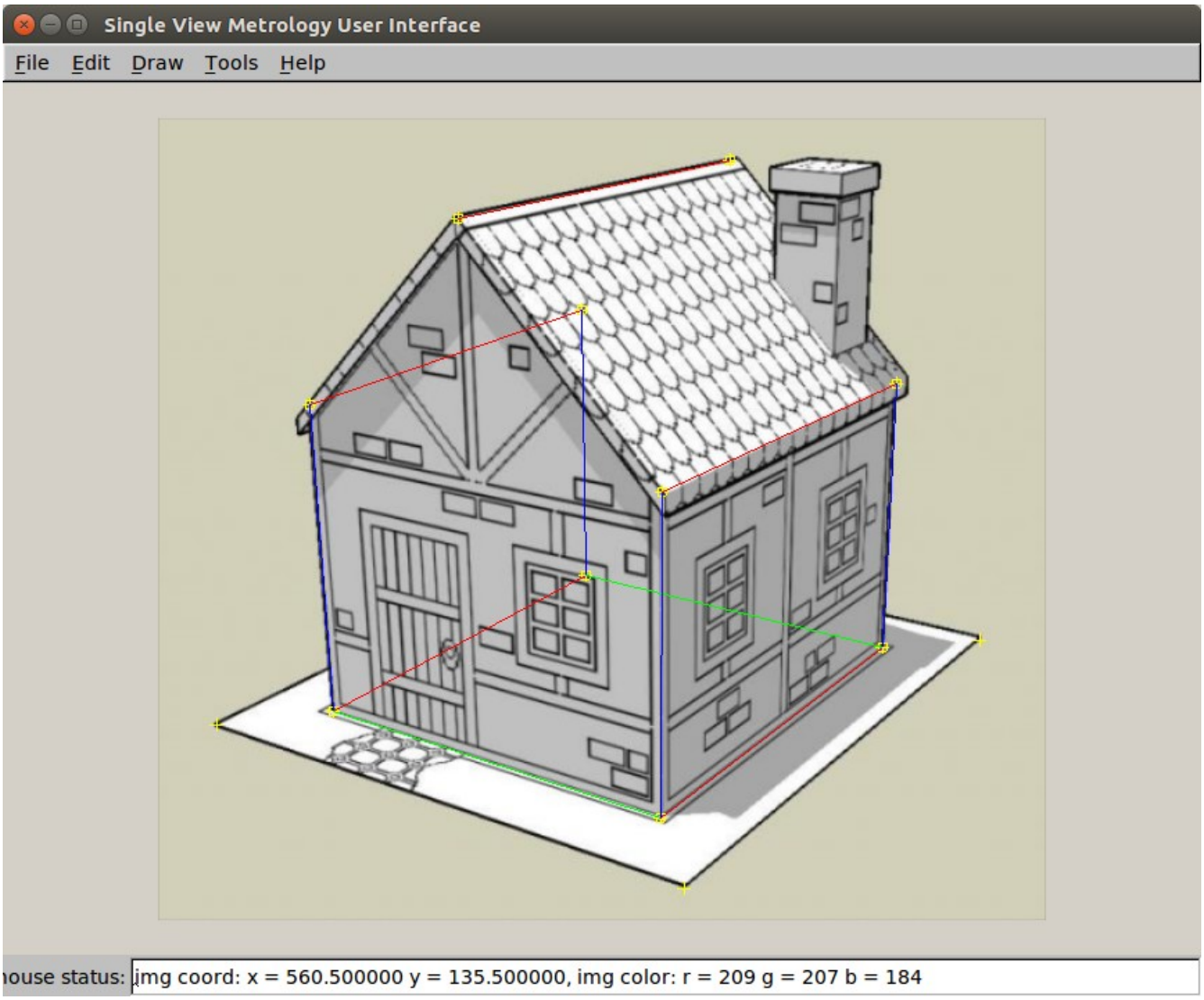
Terminal output after clicking Compute VPs:

```
eigenvector associated with smallest eigenvalue:
0.857678
0.514186
0.000701

eigenvector associated with smallest eigenvalue:
0.912757
-0.408503
```

```
-0.000577

eigenvector associated with smallest eigenvalue:
-0.123342
0.992364
-0.000346


VANISHING POINTS
xvp=[1223.203718     733.321852]     yvp=[-1581.273657     707.697337]
zvp=[356.201983 -2865.869086]
```

Calculating homographies:

I pushed the points on the base of the house to the stack to calculate the homography.



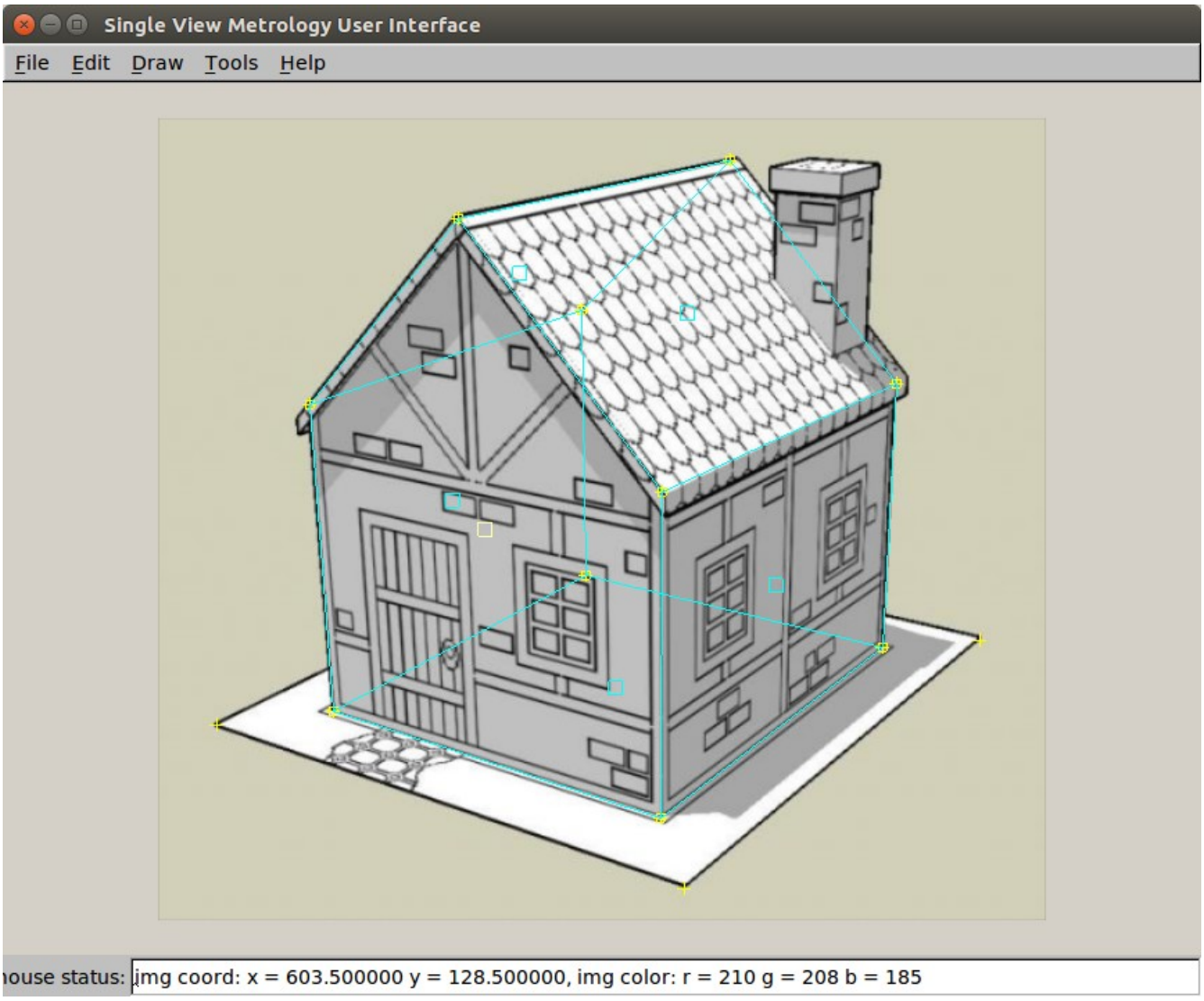
Terminal output after pushing all the points and clicking Compute Homography:

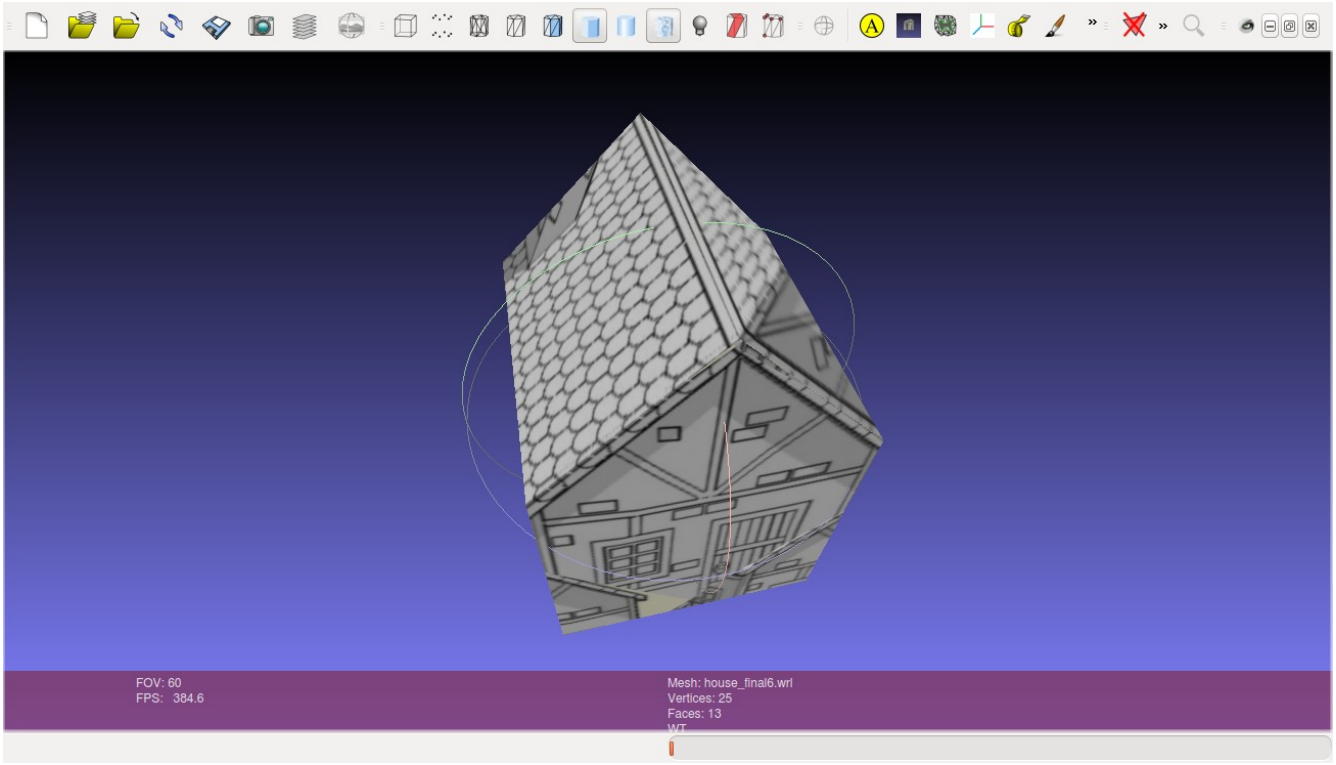
```
pushed point 11d75d0 onto stack, current size 1
Point information:
     (u, v, w) = (126.500000, 150.500000, 1.000000);
     (X, Y, Z, W) = (0.000000, 1.000000, 0.000000, 1.000000);

pushed point 11d75d0 onto stack, current size 2
Point information:
     (u, v, w) = (362.500000, 73.500000, 1.000000);
     (X, Y, Z, W) = (0.000000, 0.000000, 0.000000, 1.000000);

pushed point 11d75d0 onto stack, current size 3
```

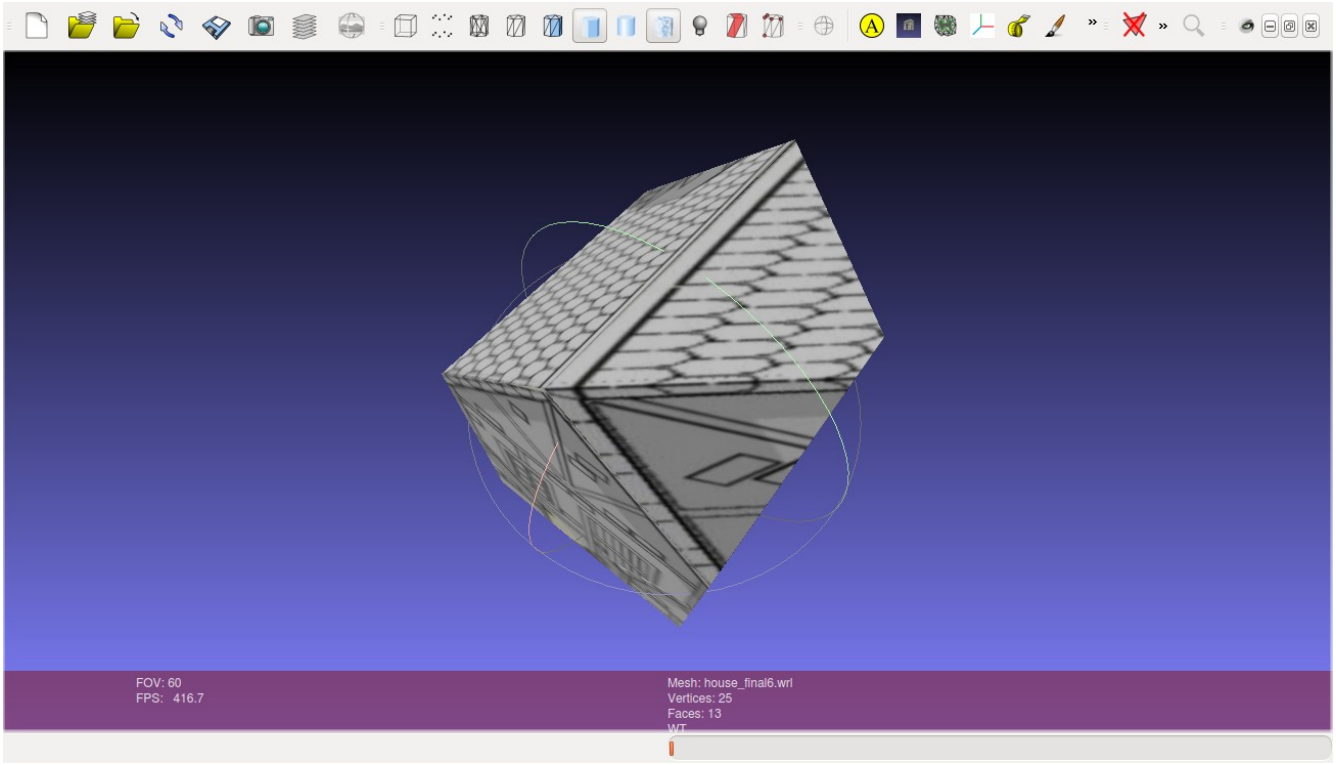
```
Point information:
    (u, v, w) = (522.500000, 196.500000, 1.000000);
    (X, Y, Z, W) = (1.000000, 0.000000, 0.000000, 1.000000);

0: (1000.000000 , 0.000000, 1.000000)
1: (0.000000 , 1.000000, 1.000000)
2: (0.000000 , 0.000000, 1.000000)
3: (1.000000 , 0.000000, 1.000000)
0:  (1000.000000,  0.000000,  1.000000,  0.000000,  0.000000,  0.000000,
-1223203.718184, -0.000000, -1223.203718)
1:  (0.000000,  0.000000,  0.000000,  1000.000000,  0.000000,  1.000000,
-733321.851613, -0.000000, -733.321852)
2:  (0.000000,  1.000000,  1.000000,  0.000000,  0.000000,  0.000000,
-0.000000, -126.500000, -126.500000)
3:  (0.000000,  0.000000,  0.000000,  0.000000,  1.000000,  1.000000,
-0.000000, -150.500000, -150.500000)
4:  (0.000000,  0.000000,  1.000000,  0.000000,  0.000000,  0.000000,
-0.000000, -0.000000, -362.500000)
5:  (0.000000,  0.000000,  0.000000,  0.000000,  0.000000,  1.000000,
-0.000000, -0.000000, -73.500000)
6:  (1.000000,  0.000000,  1.000000,  0.000000,  0.000000,  0.000000,
-522.500000, -0.000000, -522.500000)
7:  (0.000000,  0.000000,  0.000000,  1.000000,  0.000000,  1.000000,
-196.500000, -0.000000, -196.500000)
eigenvector associated with smallest eigenvalue:
-0.000000
0.643422
-0.000000
-0.000000
0.765495
-0.000000
-0.000000
0.005086
-0.000000


H=[
2.778813e+02   -5.080247e+15  3.630051e+02;
1.674296e+02   -6.044088e+15  7.422635e+01;
```

```
2.264718e-01   -4.016005e+13  1.000000e+00;
]
Hinv=[
3.695197e-03   1.145786e-02   -2.191851e+00;
1.816979e-16   -2.360452e-16  -4.843649e-14;
6.460139e-03   -1.207447e-02  1.000000e+00;
]
```

Now, we generate the full model. We can use sameXY to get the coordinates of the points above the ground, and use sameZ to get the coordinates of the rest of the points, including the hidden points. Using ZX Rectangle and SweepRectangle also work. We will finish with the model:







See the textures used in the VRML folder.