



// PANELIZATION STUDIES, Maysam Ghaffari  
 // Coding Form, Satoru Sugihara  
 // Sci-Arc, Fall 2011

```
import processing.opengl.*;
import igeo.geo.*;
import igeo.core.*;
import igeo.util.*;

size( 800, 800, IGL.GL );
smooth();

//input geometry from 3dm file
IG.open("midterm.3dm");
ISurface[] surfs = IG.surfaces();

// Layers in rhino
ILayer layer1 = IG.layer("structure");
ILayer layer2 = IG.layer("infill");
ILayer layer3 = IG.layer("frame");

//input jpg map files
IImageMap map = new IImageMap("map1.jpg");
IImageMap map2 = new IImageMap("map5.jpg");

// surface subdivision
for ( ISurface surf : surfs ) {
  int hDiv = 15, vNum = 30;
  double vDiv = 1.0/hDiv, vinc = 1.0/vNum;

  IVec[] endpoints = new IVec[30][60];

  for (int i=0; i < hDiv; i++) {
    for (int j=0; j < vNum; j++) {
      IVec pt11 = surf.pt( i+1*vDiv, j*vinc );
      IVec pt21 = surf.pt( i+1*vDiv, (j+1)*vinc );
      IVec pt12 = surf.pt( i*vDiv, j*vinc );
      IVec pt22 = surf.pt( i*vDiv, (j+1)*vinc );
      ISurface panel = new ISurface(pt11, pt21, pt22, pt12).clr(map2.clr( i*vDiv, j*vinc )).layer(layer2);

      //Frame Extrusion
      int crvDegree = 1;
      ICurve crv = new ICurve(new IVec[] {pt11, pt21, pt22, pt12}, crvDegree, true);
      IG.extrude(crv, -1).clr(255, 50).layer(layer3);

      // diagrid structure
      IVec spt1 = surf.pt( i*vDiv, j*vinc, -10 );
      IVec spt2 = surf.pt( (i+1)*vDiv, (j-1)*vinc, -10 );
      IVec spt3 = surf.pt( (i+1)*vDiv, (j+1)*vinc, -10 );
      new ICylinder(spt1, spt2, 0.2).clr(map2.clr( i*vDiv, j*vinc )).layer(layer1);
      new ICylinder(spt1, spt3, 0.2).clr(map2.clr( i*vDiv, j*vinc )).layer(layer1);

      //Infill Scale
      double val = map.get( i*vDiv, j*vinc );
      IVec center = surf.pt( (i+0.5)*vDiv, (j+0.5)*vinc );
      panel.scale(center, 1 - val );

      //Infill Rotation
      IVec normal = surf.nml( (i+45)*vDiv, (j+25)*vinc );
      panel.rot(center, normal, val*PI/1);

      // Structure vertical connection and shading device
      IVec pt1 = surf.pt( i*vDiv, j*vinc );
      IVec pt2 = surf.pt( i*vDiv, j*vinc, val*20 );
      ICurve crv1 = new ICurve(pt1, pt2).clr(map2.clr( i*vDiv, j*vinc ));
      ICurve crv2 = new ICurve(IG.offset(crv1, val*10));
      IG.loft(crv1, crv2);
      new ICylinder(pt1, pt2, 0.3).clr(5);
    }
  }
  surf.del();
}

//save in rhino
IG.save("test_output_file.3dm");
```

