

Geometric & Graphics Programming Lab: Lecture 18

Alberto Paoluzzi

December 2, 2016

1 Workshop N.8

2 Minimal git/github instructions

Workshop N.8

Solid models from .SVG drawings

The goal of this workshop is to quickly produce a solid model of a house starting from an image of its plan drawing.

Summary of steps

- get an image of the architectural plan
- produce the 1D wireframe of main building subsystems
- transform the SVG files into bunches of lines
- apply `pyplasm` transformations to lines to get 3D parts
- apply colors and/or textures to model parts

Dowload the inkshape drawing tool

It's free and open source

Inkscape is a professional vector graphics editor for Windows, Mac OS X and Linux.

- Flexible drawing tools
- Broad file format compatibility
- Powerful text tool
- Bezier and spiro curves

DOWNLOAD

InkScape

Choose a house (or apartment) image to model

Find a nice and fair architectural plan



Figure 1: Images from Google

Import the image in InkShape

- **Freeze** the image of the chosen architectural plan
- Consider the main building subsystems:
 - external enclosures
 - internal partitions
 - floor
- produce a simplified 2D cellular complex (wire frame drawing) using different colors
- select and export the subsystems in different .SVG files
- save the various produced .SVG files in your repository

Transform your partial SVG drawing in a bunch of lines

- Use **Chrome**
- Use the web service <http://cvdlab.github.io/svg2lines/>
- save the various produced **.LINES** files in **your repository**

Use lines2lar

- Start your ipython interactive session in your 2016-12-02 repository
- import `pyplasm`
- from each file read a list of pairs of 2D points

Transform in 2D and/or 3D solids via PLaSM

- Give 2D **solid shape** via proper **OFFSET** primitive
- Transform in 3D **solid parts** via product times a 1D interval

Open doors and windows via proper subtractions

- Invent some trick to open **doors** and **windows**
- **Repeat** until necessary for the various subsystems
- Apply **colors** and/or **textures**

look in **pyplasm sources**

for **fenvs.py** file and find the appropriate syntax

Put all together and see ... :o)

ENJOY !!



Figure 2: Images from Google

REQUIREMENTS

- Write a single notebook, named `workshop_08.ipynb`
- Choose a notebook Title, for example `<House_modeling>`
- Start the notebook with a `web reference` and one/more `image/s` of your `type of designe`
- List the `variables` used in your code, with a `textual definition`
- Provide a `short description` of used `geometric methods` you are going to implement
- Provide some `images` of your model from different viewpoints
- Use measures (where necessary) in `meters (m)`

Style specs

- use **meaningfull identifiers** (variables and parameters)
- use **camelCase** ids
- add **Python docstrings** (google for it)
- produce a **single** notebook file, named **workshop_08.ipynb**
- file path: **your_repo/2016-12-02/workshop_08.ipynb**

Minimal git/github instructions

Minimal git/github instructions (1/2)

create your local repository

```
$ mkdir 2016-12-02  
$ cd 2016-12-02  
$ touch workshop_08.ipynb
```

Minimal git/github instructions (2/2)

commit your work

```
$ git add -A .
$ git commit -m "add a short note to commit"
$ git push origin master
```