Geometric & Graphics Programming Lab: Lecture 16

Alberto Paoluzzi

November 25, 2016

1. Workshop N.7

2. Minimal git/github instructions

Workshop N.7

Modeling wooden doors and windows

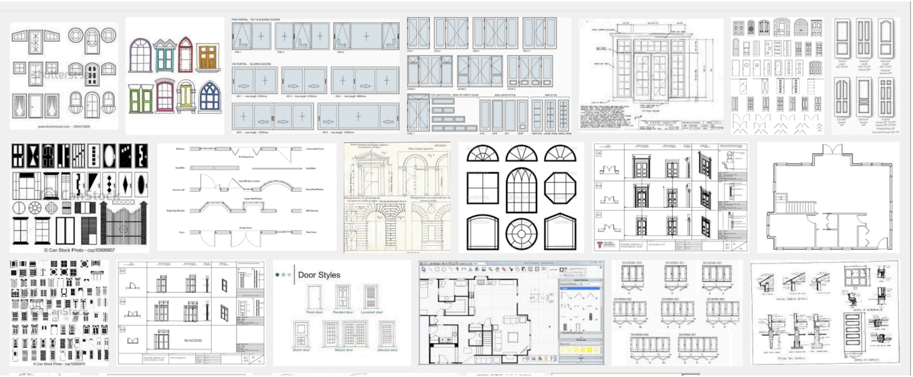


Figure 1: Images from Google

Look at some examples

- latest design of wooden doors and windows
- modern doors and windows
- glass doors and windows
- sliding glass windows
- sliding glass windows with screens
- sliding glass windows design

Design ideas and terminology

- French Doors With Side Windows, Sliding French Doors, Tudor Windows, Sliding Glass Door, Rear Sliding, French Door Screens, French Exterior, Interior French . . .

Solution hint (1/3)

Use enumerative representation with variable resolution

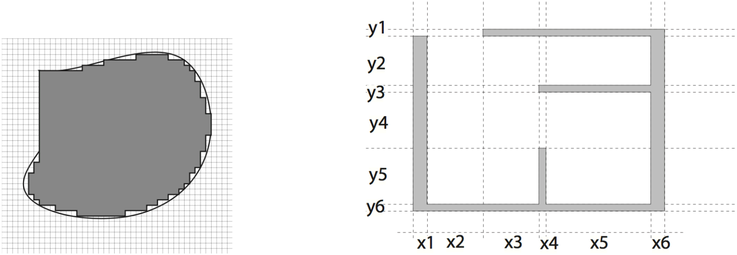


Figure 13.3 Enumerative representations: (a) with constant resolution (b) with variable resolution

Figure 2: From chapt.13, Geometric Programming for Computer-Aided Design, Wiley, 2003. (free download from `uniroma3.it` domain)

Solution hint (2/3)

Use enumerative representation with variable resolution

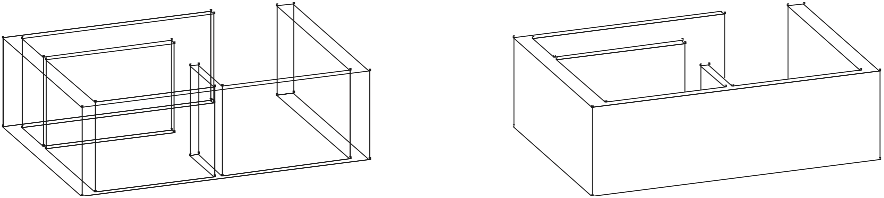
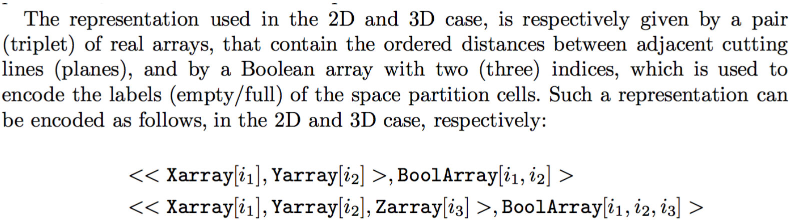


Figure 13.4 3D enumerative representation

Figure 3: From chapt.13, Geometric Programming for Computer-Aided Design, Wiley, 2003. (free download from `uniroma3.it` domain)

Solution hint (2/3)

Use enumerative representation with variable resolution



The representation used in the 2D and 3D case, is respectively given by a pair (triplet) of real arrays, that contain the ordered distances between adjacent cutting lines (planes), and by a Boolean array with two (three) indices, which is used to encode the labels (empty/full) of the space partition cells. Such a representation can be encoded as follows, in the 2D and 3D case, respectively:

$$<< \mathtt{Xarray}[i_1], \mathtt{Yarray}[i_2] >, \mathtt{BoolArray}[i_1, i_2] >$$

$$<< \mathtt{Xarray}[i_1], \mathtt{Yarray}[i_2], \mathtt{Zarray}[i_3] >, \mathtt{BoolArray}[i_1, i_2, i_3] >$$

Figure 4: From chapt.13, Geometric Programming for Computer-Aided Design, Wiley, 2003. (free download from `uniroma3.it` domain)

REQUIREMENTS

- Write a single notebook, named `workshop_07.ipynb`
- Choose a notebook Title, for example `<Design_of_Doors_and_Windows>`
- 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
- Include the coding of two main parametric function named `door` and `window`
- Both functions are curried (second level):
- Provide 4+3 formal parameters
  - `X, Y, Z`, corresponding to lateral quotes, of type `[float]`
  - `occupancy`, of type `[bool]`
  - `dx,dy,dz`, for box dimensioning, of type `float`
- Provide the images generated by some executions with very different actual parameters.
- Use measures in meters (m)

Style specs

- use meaningfull identificators (variables and parameters)
- use camelCase ids
- add Python `docstrings` (google for it)
- produce a single notebook file, named workshop_07.ipynb
- file path: `your_repo/2016-11-25/workshop_07.ipynb`

Minimal git/github instructions

Minimal git/github instructions (1/2)

create your local repository

```
$ mkdir 2016-11-25
$ cd 2016-11-25
$ touch workshop_07.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
```