HowTo GridGen

This will walk you through creating a grid for your monitor-setup.

Installation

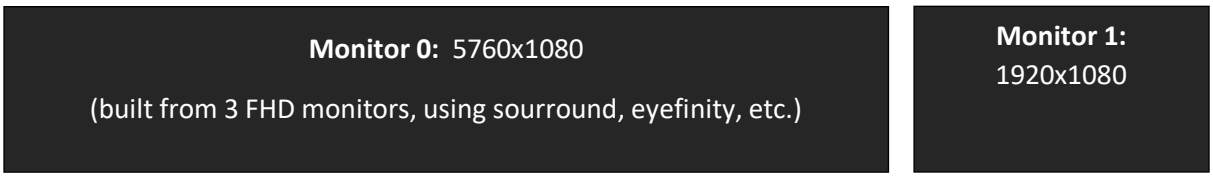
Install Python 3 and the psd_tools package.

Step by step:

1) install Python 3.x from https://www.python.org/
2) install psd_tools:
   a. open commandline: [win]+[r], open: “cmd”, [ok]
   b. type: “pip install psd_tools” and follow process
3) you should be now able to open the program:
   a. double click on GridGen.cmd

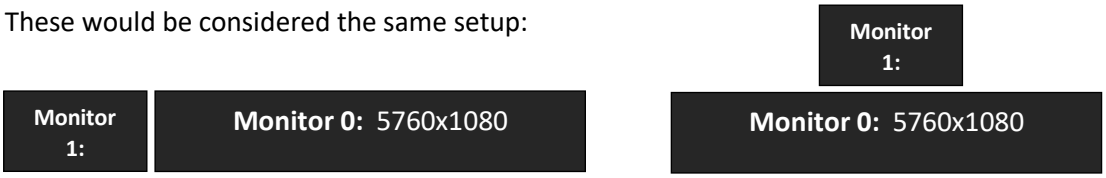
Monitor Setup

Assume you have the following monitor-setup:

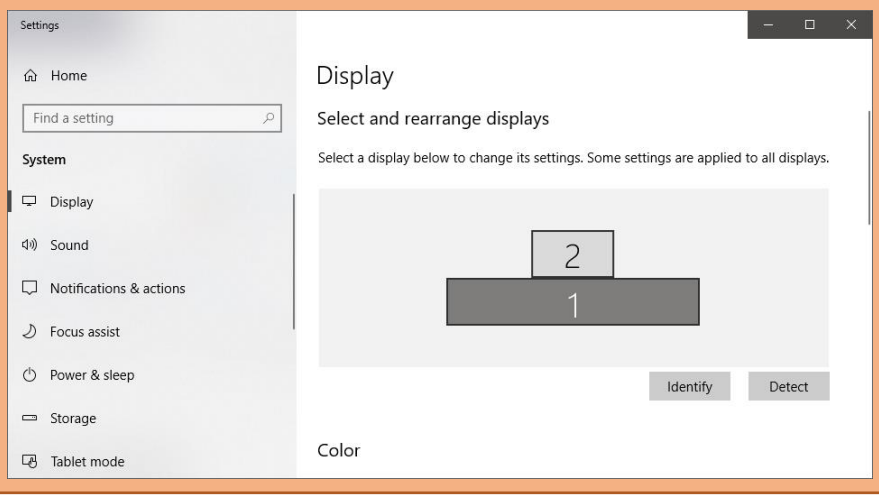


It is not relevant, how your monitors are placed. You just have to know which **.gird** you want to have on which monitor (and how it should be repeated).

These would be considered the same setup:



In the program counting always starts with 0, even if your operating system names the monitors different! eg.: Windows 10



Grid Definition

To define a grid, you have to create a .psd-file.

Use Gimp or the .psd-programm of your choice to create grid-definitions.

(I prefer Paint.Net, it needs an addon for .psd-files however!)

Define the grid as follows:

- use complete transparent layers
- draw one rectangle per layer
- Each pair of layers defines a grid (the lower one) and its trigger.
  The first layer is the first grid, the second layer the first trigger, the third layer is the 2nd grid, and so on...
- Triggers must be defined (they can be duplicated from the grid-layer however)
- overlapping of grids / triggers is possible, but take care of the right order.

Important

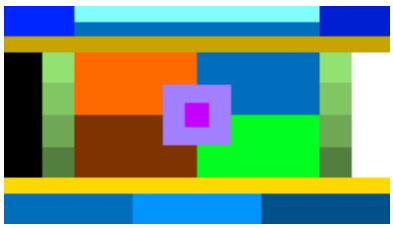
- Make sure that all layers have the same size.
- .psd files with the same ratio of your monitor may make it easier to define grids
- prefer small resolutions in .psd files, for better runtime (eg. if you just want a grid that divides a monitor in half, consider using a 2x1-pixel image)
  A tenth of the Monitor dimensions (192x108 for a FHD monitor) seems to be a good compromise between ease of use, accuracy and runtime.

Example:

see the **GridCreator\example_psd** folder !

- **Grid1.psd**
- **Grid2.psd**
- **test1.psd** (unrelated to previous)
- **test2.psd** (unrelated to previous)

The trigger-boxes of “Grid2.psd” -->

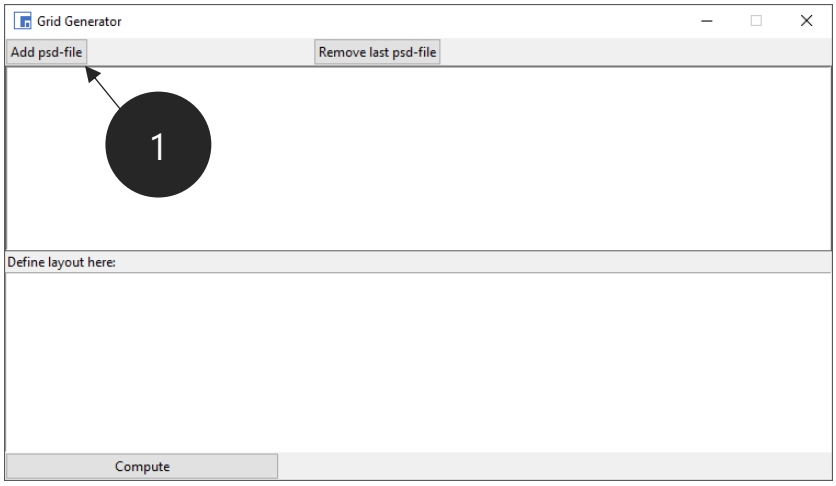


.psd to .grid conversion

To convert your .psd file, run the program.

Under Windows do:

- In the folder of Main.py press [SHIFT]+[Right Click]
- click [open PowerShell window here]
- in the PowerShell type “python Main.py” and hit [ENTER]
- a new window should open:



First load all .psd files you want to use.

Use the file-dialog that opens, when clicking on: 1

The read files are then shown in the upper textbox, next to their **IDs**. 2

The next step is to define the Layout:

- for each monitor define the layout in a new row of the lower textbox.

One .psd per Monitor

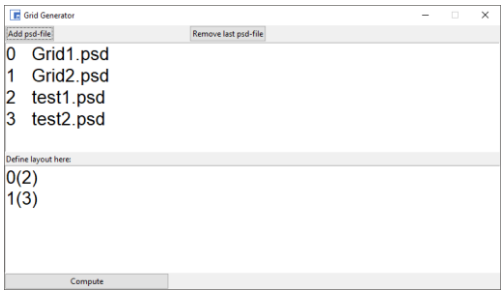
use the following structure:

<monitor_id>**(**<grid_id>**)**

replace <monitor_id> and <grid_id> with appropriate numbers.

Example:

Two monitors, the first one uses *test1.psd* and the second one *test2.psd* .



Multiple .psd per Monitor

It is also possible to use multiple .psd files for one monitor.

This is useful for Monitor Setups like in the second section.

If Monitor 0 is built from 3 physical monitors, using 3 grids (same or different ones) might be convenient.

To do so use the following structure:

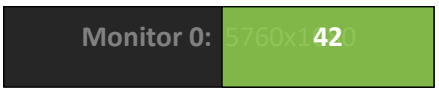
<monitor_id>**(**<pos_x> <pos_y> <grid_id>**)**

The <pos> arguments state, where the grid should be placed in the monitor.

Example:

0(1 0 42)

would place Grid 42 on Monitor 0 like:



However, arbitrary many subgrids can be used by appending their definitions comma-seperated.

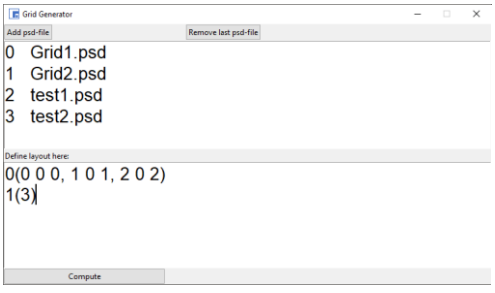
Example:

0(0 0 41, 1 0 42, 2 0 43)

would place the grids on Monitor 0 like:

alternatively:

This



would place the grids similar as before.

Grid1.psd is used for the orange area,

Grid2.psd for the green one

and test1.psd for the blue one.

Compute

Clicking [Compute] opens a dialog to specify the save-location and starts the computation. This will take a while (when using big images / .psd files) and the program will be unresponsive.

Use your .grid

The last step is moving the grid, you created before to your GridMove folder.

Paste it in **[your gridmove installation path]/Grids**. There should already be other .grid files.

Now, in GridMove select **Tiles**, **refresh this list**. And finally select the gird.