Bright Project – User documentation

Julien FAUCHER
24/08/2006

Introduction

Bright (*Bright Reconstruction with light*) is a project which aim is to build an affordable 3-D scanner. It uses inexpensive off-the-shelf products:

- a digital camera (or web-cam)
- a laser line
- a system
- to rotate the object

This document is the user documentation of the software. Its purposes are to help the user with the program and to help the reader of the scientific report to put theory into practice. If you want more information about the maths, algorithms and technologies used by the project and the software implementation, please read the scientific report.

Installation

To install **Bright**, you must first install on your computer the MIMAS library. Please note that **Bright** has only been tested so far with a Ubuntu 6.06 GNU/Linux distribution, even though there should be no problem with other distributions, as long as you can install MIMAS and the other required packages on it.

Before installing MIMAS, make sure you have QT libraries, LAPACK, MINPACK and Boost 1.33 set up on your computer, although they are not compulsory to compile MIMAS, they are required by **Bright**. Make also sure you compile MIMAS examples since one of the example is a camera calibration tool that you may need to calibrate your camera.

To display the results we propose you 2 alternatives:

1. Gnuplot for a first glimpse of the result, you will see a cloud of points.
2. For a 3-D surface reconstruction you may feed the 3-D points outputted by our software to PowerCrust . Then the result may be observed with Geomview.

Once MIMAS is installed on your computer, you can compile and run Bright with the commands :

```
$ qmake
$ make
$ ./bright
```

Requirements

To do the 3-D reconstruction you need:

- a laser line
- a camera (a web-cam is enough)
- a rotating platform
- a stand to hold everything together. Notice that a stand holding the camera and the laser line is enough, as long as you can make sure the rotating platform won't move once it's calibrated.

and to do the calibrations :

- a chessboard pattern (size 5 * 4)
- a bent chessboard pattern (same size)

Unless you want to change the source code (config.h) or give parameters to the program, you should use a chessboard of the same size as mine.

If you want for instance to change the size without changing the source code, you can provide some parameters :

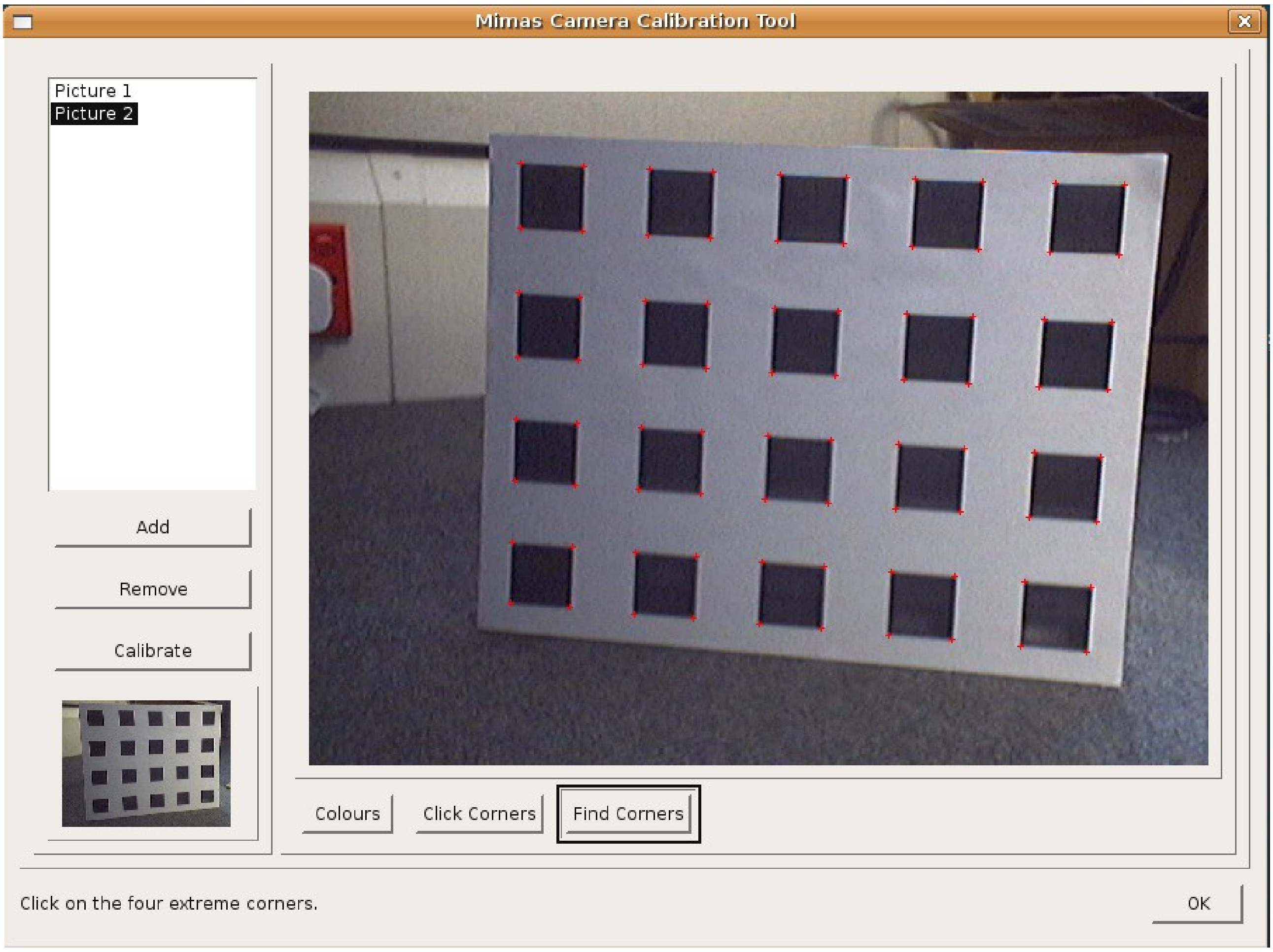
```
$ ./bright -?
Usage: bright [OPTION...]
  -d, --device=device-name   Specify video4linux device (default:"/dev/video")
  -w, --width=width          Width of the chessboard (number of black squares)
  -h, --height=height        Height of the chessboard (number of black squares)
  -u, --pwidth=width         Width of the chessboard for the platform calibration
  -v, --pheight=height       Height of the chessboard for the platform calibration
  -i, --min=min              Minimum threshold for laser detection (default: 230)
  -a, --max=max              Maximum threshold for laser detection (default: 240)
  --window=size              Size of the window for corner detection (default: 10)
  --correlation=corr         Specify minimum correlation to be calculated to assume
                             that the corners are accurately found (default: 0.8)
Help options:
  -?, --help                   Show this help message
  --usage                      Display brief usage message
```

Camera calibration

The first step is to make sure your camera is calibrated. You can of course use any camera calibration software, but if you don't have any you can use the one provided with MIMAS. It's available in the directory mimas2/examples/camera_calibration

Start it with the command :

```
$ ./calibration
```



To calibrate you camera, you have to take several pictures of a chessboard pattern. For each picture, you are required to click the four extreme corners of the chessboard, starting from the upper-left one, and going on clockwise.

For the first image (or if the luminosity changed to much during the acquisition), use the button *"Colours"* to select the values of black and white in your image. Click once on *"Colours"*, and click on a black square of the chessboard. Then click again on *"Colours"*, and click on a white area within the chessboard. Click a third time on *"Colours"* to finish.

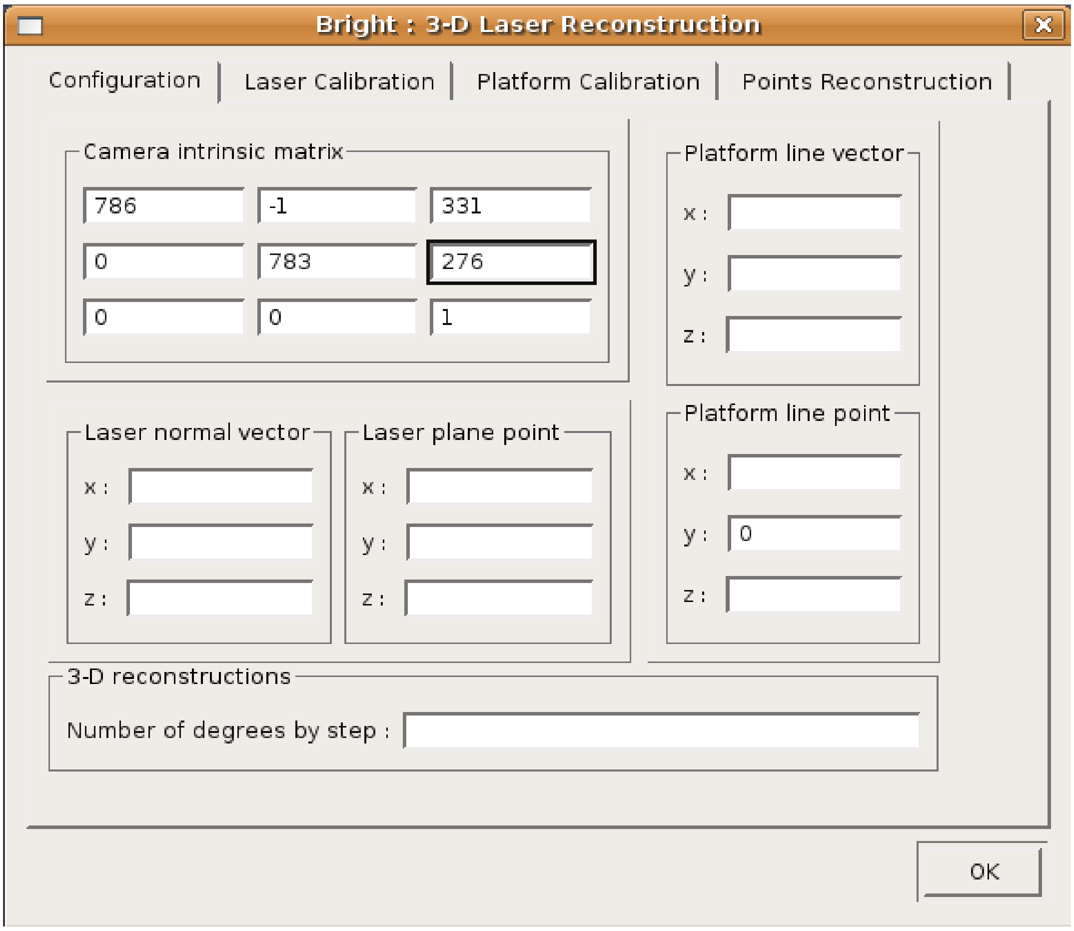
There is more information about the camera calibration software in the README file in the directory. Calibrate your camera, and save the result in a file, you will need it for the 3-D reconstruction.

3-D reconstruction

The 3-D reconstruction needs several steps, that are described and explained here. The graphical user interface (GUI) has been designed such that different tabs represent different steps.

The configuration

Unless you have already calibrated your platform and you laser line (and didn't move them since the last calibration of course), you won't have many things to fill in this tab.

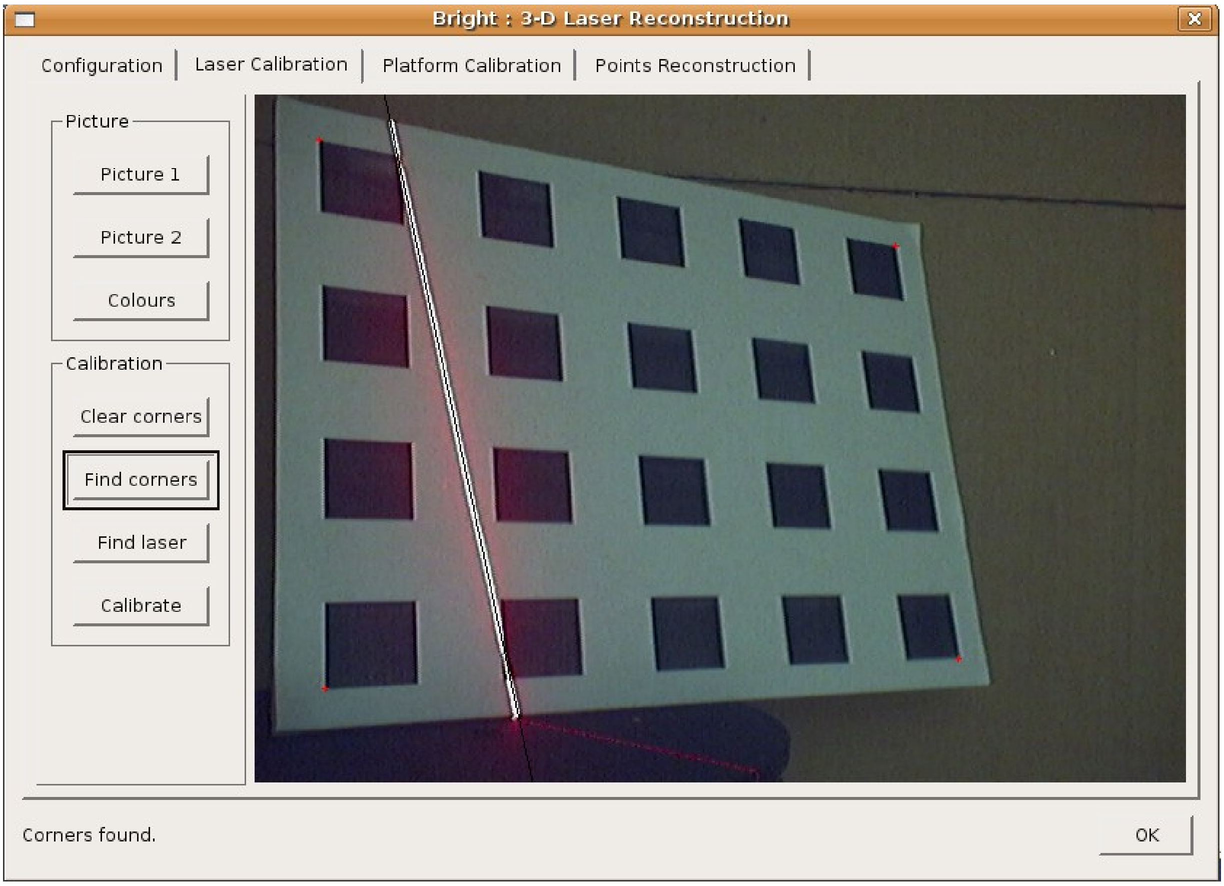


If you are just starting to use the program, just fill in the camera intrinsic matrix from the one you have calculated above.

If you have already calibrated the laser and/or the platform, you may use the values given by the previous calibration.

The laser calibration

The first step is to calibrate you laser. You need to take 2 pictures of the laser on the chessboard.



Use the button *“Picture 1”* to grab an image from the camera. Then click on *“Find laser”*. If the laser line is found, a black line will be printed over the laser line on the picture. Then you need to find the corners of the chessboard. If you have used the MIMAS camera calibration example, you will see that the method is very similar.

Set up the colours (read the camera calibration above for more information), then you are required to click on the four extreme corners of the chessboard. Start from the upper-left corner, and go on clockwise. You don't need to be very precise, the corner detection algorithm will refine your approximation.

Once you have clicked on the 4 corners, click on *“Find corners”*. If everything goes well, the correct location of the corners should be found. Otherwise, you can click on *“Click corners”* to give another approximation. If it's still not working, take another image and try again.

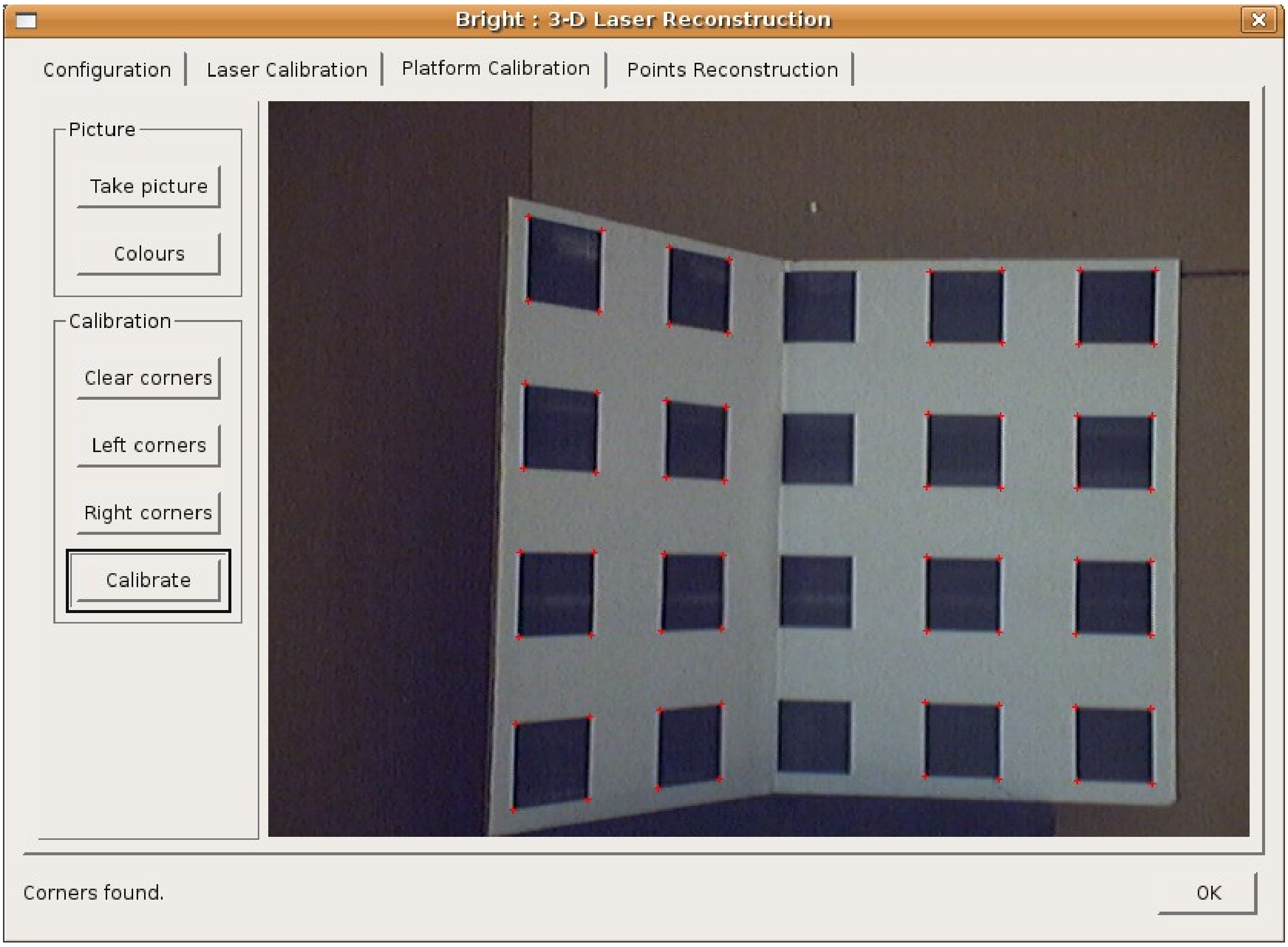
Then, once you have finished with the first image, repeat the procedure with the second image, using the button *“Picture 2”*. You should not have to set up the colours again, unless the luminosity has changed too much.

Then, you can calibrate the laser with the button *“Calibrate”*. The result should be shown in your terminal, and in the tab Configuration as well.

The platform calibration

Last but not least, you need to calibrate the position of the platform. This is done by taking a single picture of a dedicated calibration object : a bent chessboard. The angle between the 2 planes doesn't matter.

So put your chessboard on the platform such as the juncture of the chessboard matches the axis of the platform.

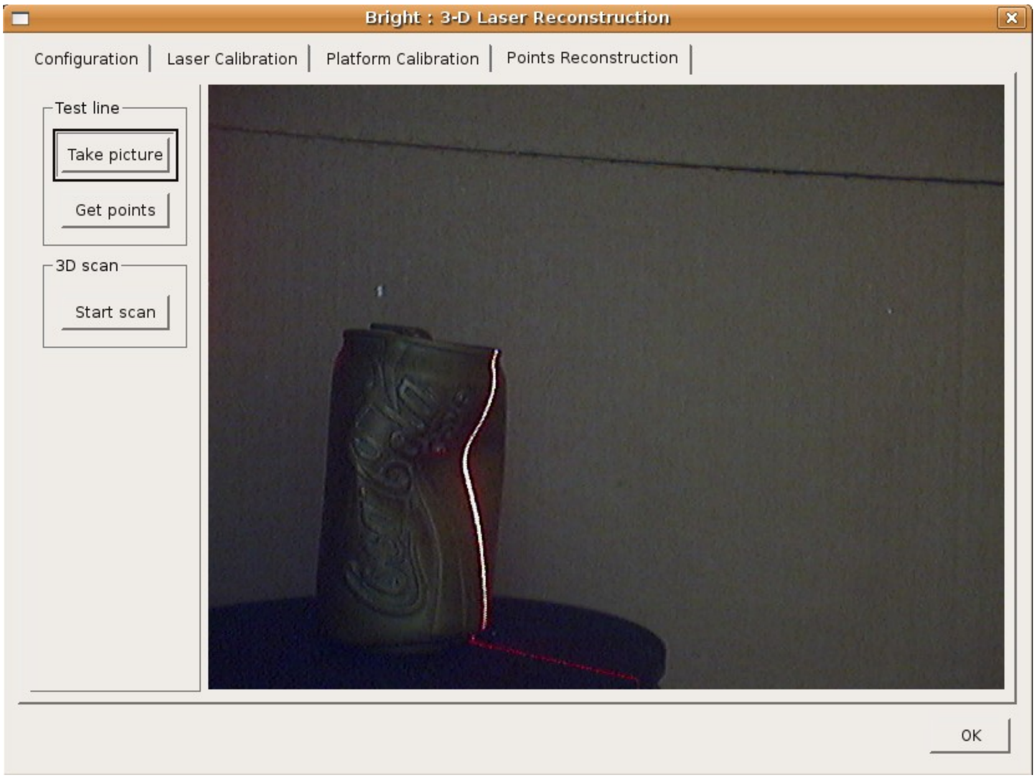


Then, take a picture with *“Take Picture”*, and provide the values of black and white as usual with the button *“Colours”*. Then click on the four extreme corners of the left chessboard, and click on *“Left corners”*. Do the same with the right corners and the button *“Right corners”*.

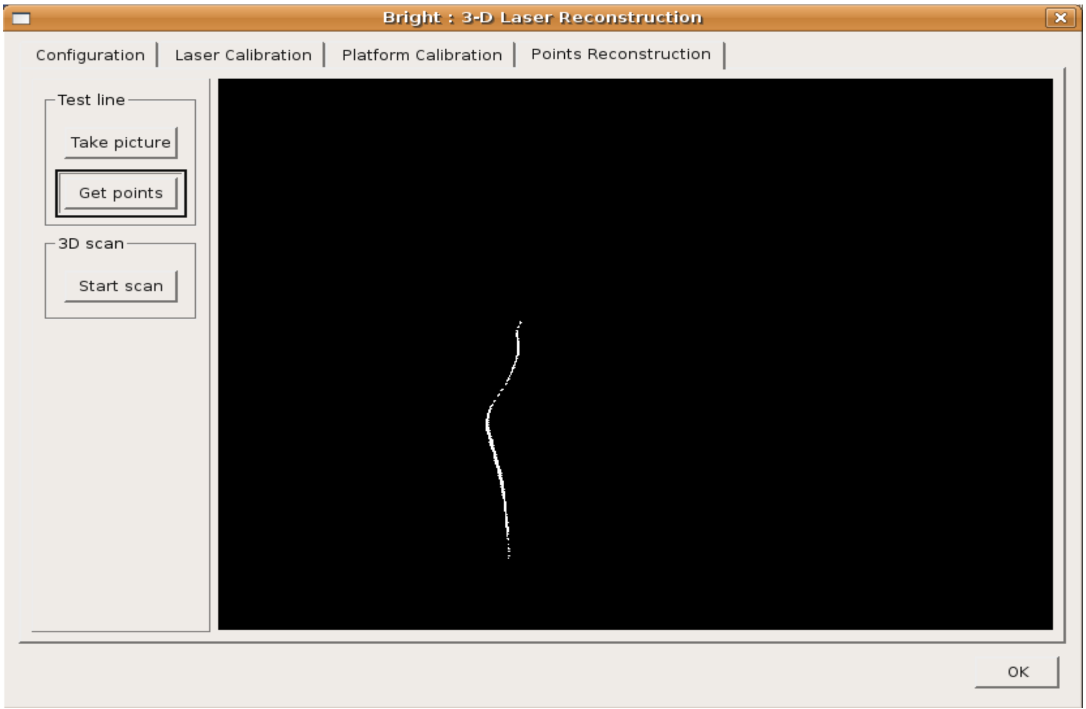
Once the 2 chessboards are found, you can compute the intersection of the 2 chessboards with the button *“Calibrate”*. The calibration result are printed in both the terminal and the Configuration tab.

The 3-D reconstruction

Now put your object on the platform, and switch on your laser.



The button *"Take picture"* is just to do a test, to make sure the object is well positioned, and the laser line visible. You can also try the laser detection with the button *"Get points"*. The laser line should appear in white, so you can test the line detection.

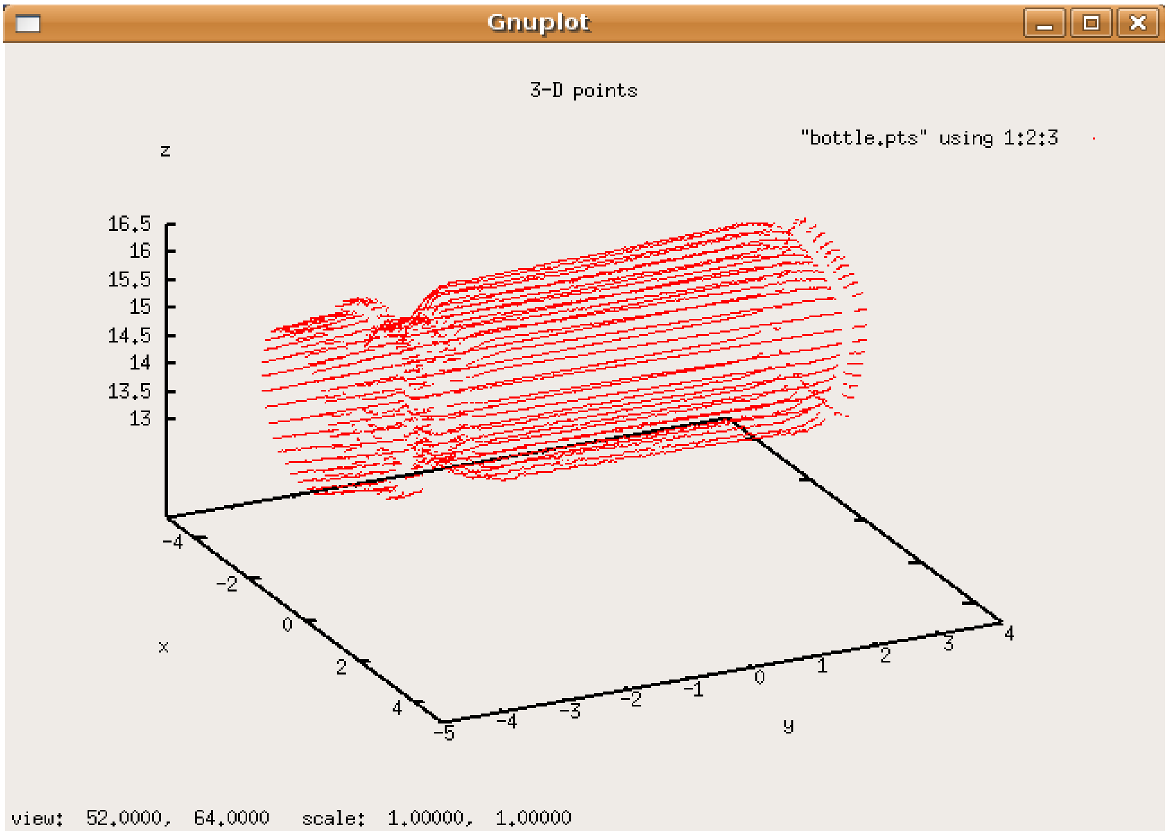


Make also sure you have specified in the "Configuration" tab the step you want. To do a first draft, try 10 or 20 degrees, and for more precision you can try something like 1 degree.

Once everything is ready, click on *"Start scan"* to proceed. Once the scan is over, the 3-D points should be available in the file output.gnuplot.

If you have gnuplot installed, you can see the result with

```
$ gnuplot plotme
```



Note : plotme is a default configuration file for gnuplot setting up the axis and reading the file output.gnuplot.

You can also try the PowerCrust algorithm. If you have PowerCrust and Geomview installed on you computer :

```
$ powercrust -m 1000000 -i output.gnuplot
$ geomview pc.off
```