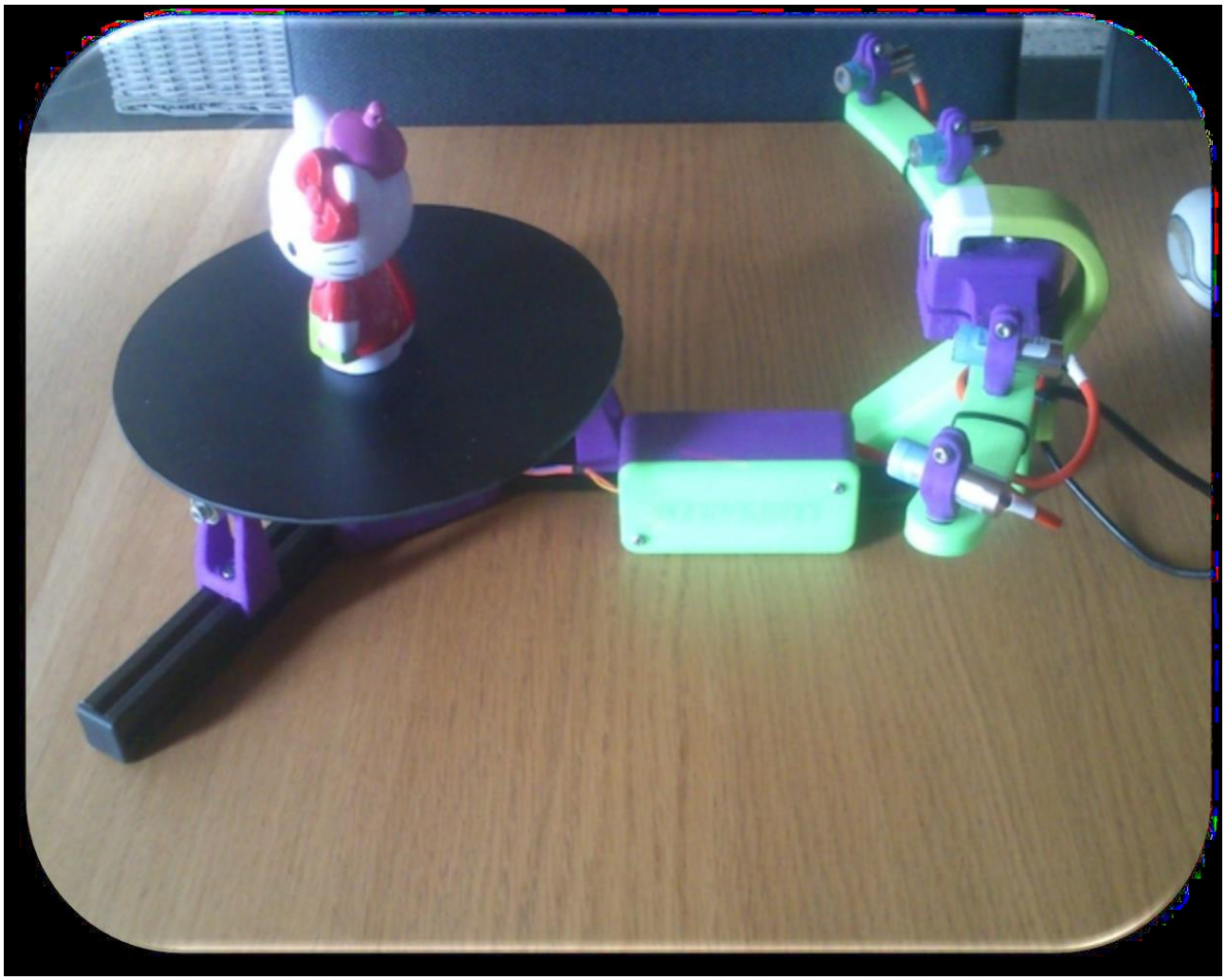
Sardauscan Tutorial



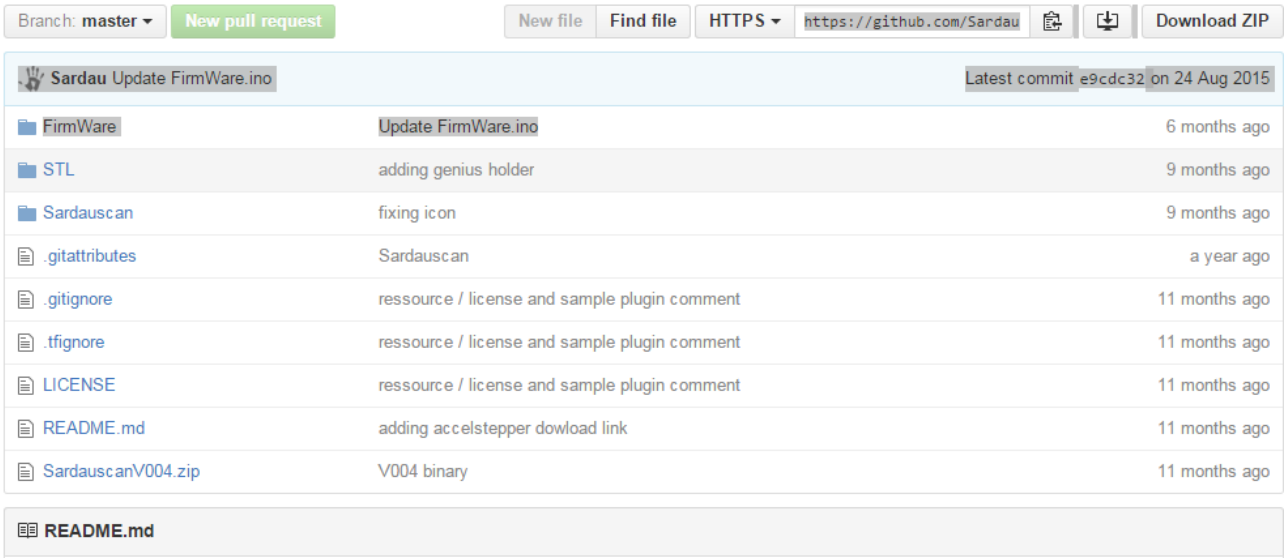
Hugo BENOIT-JANNIN

This user manual only deals with steps for calibrating and
Scanning an object in 3D with the scanner Sardauscan, the design part of the scanner
Will not be discussed in this tutorial. If you want to build your Sardauscan
Please visit: http://www.thingiverse.com/thing:702470 where everything is
(Material to be purchased, program for the Arduino, design of printed parts).
Sardauscan is a low-cost, open-source 3D scanner that can compete with
With a very good "plug and play" scanner costing more and being closed to any
Change.
The Sardauscan uses red light beams, which will be
Detected by the webcam, the webcam records the deformation of the light beam
And then translate it as an image on the software.
Provide a large space around the scanner to prevent the scanning from recording
Shapes too close, causing noise and false scanning.

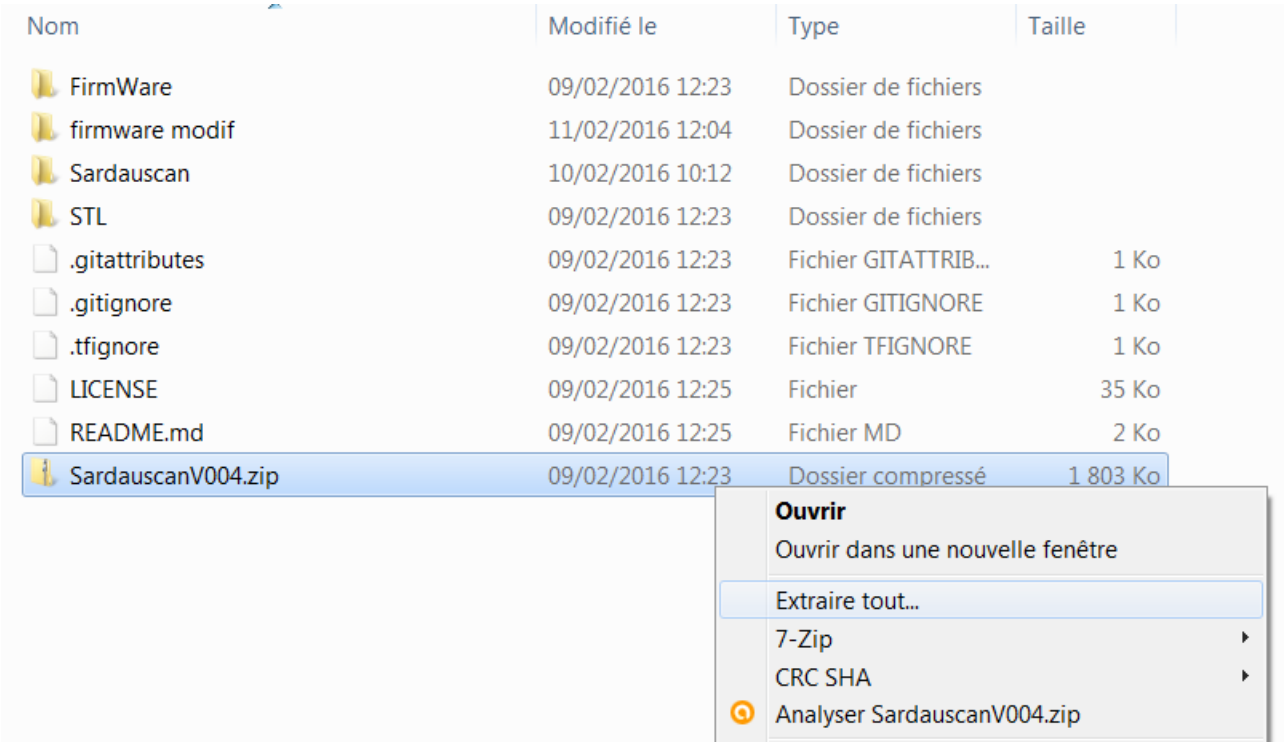
Summary

Installing the Interface Software

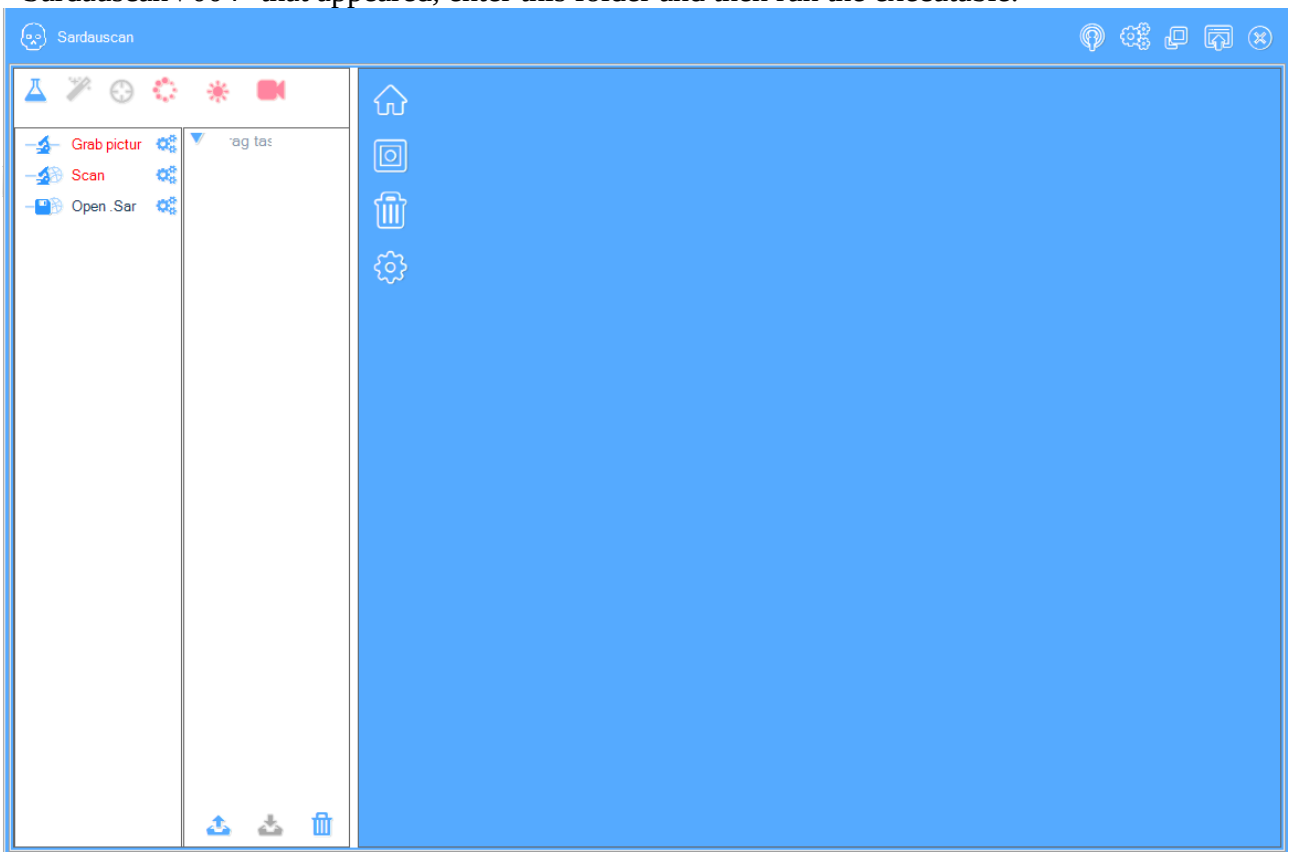
Go https://github.com/Sardau/Sardauscan to download Interface software.



Download the ZIP file, Once the zip folder is downloaded, move the file Zip in your documents (it will unzip automatically). But we still have to deziper Another file in Sardauscan-master

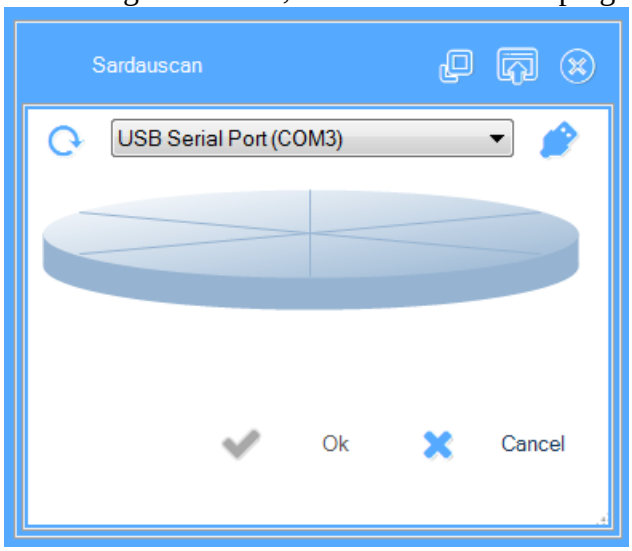


Once the file has been extracted, you have another file on behalf of
"SardauscanV004" that appeared, enter this folder and then run the executable.



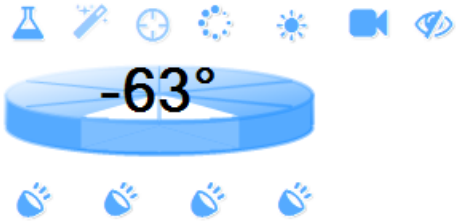
So you have your software launched, you have to recognize the different elements
Of the scanner, previously connected.

Click on the tray or lasers icon, a window will open.
Update, select the proposed USB port and click "ok".
If nothing is detected, do not hesitate to unplug / reconnect the Arduino card



For the camera to check even if the icon is blue indeed the software may have recognized the camera
Integrated if you are using a laptop. To remove the built-in camera right click
On the icon and then "remove".

Then choose "Hercules HD Twist" then click on "ok"
You can normally use each laser and the
Clicking on the laser icon, and then activating each laser independently.

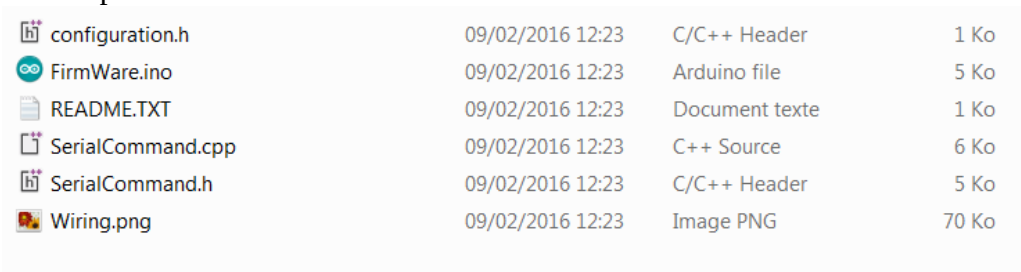


It is likely that the tray does not turn, check whether LEDs on the board
Electronics are on, if they are on, the tray is working, if not
Not the case, it is necessary to modify some parameters in the code Arduino.

In case of problem
Open the FirmWare folder

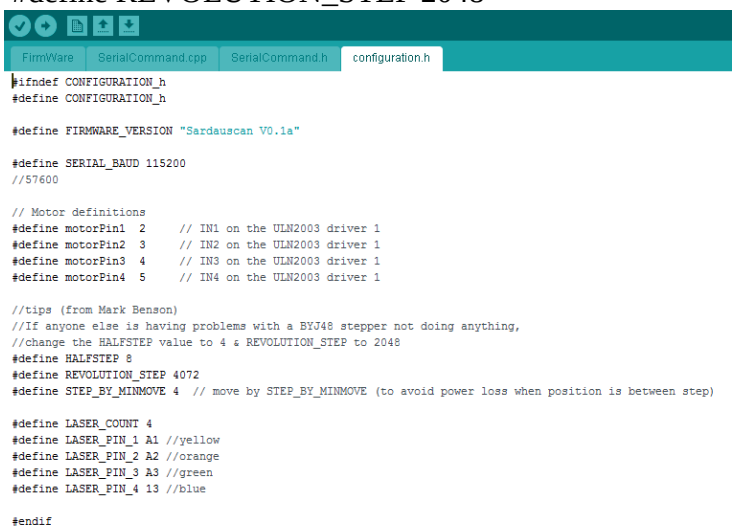
FirmWare	09/02/2016 12:23	Dossier de fichiers	
firmware modif	11/02/2016 12:04	Dossier de fichiers	
Sardauscan	10/02/2016 10:12	Dossier de fichiers	
SardauscanV004	09/02/2016 14:05	Dossier de fichiers	
STL	09/02/2016 12:23	Dossier de fichiers	
.gitattributes	09/02/2016 12:23	Fichier GITATTRIB...	1 Ko
.gitignore	09/02/2016 12:23	Fichier GITIGNORE	1 Ko
.tfignore	09/02/2016 12:23	Fichier TFIGNORE	1 Ko
LICENSE	09/02/2016 12:25	Fichier	35 Ko
README.md	09/02/2016 12:25	Fichier MD	2 Ko
SardauscanV004.zip	09/02/2016 12:23	Dossier compressé	1 803 Ko

Then open Firmware.ino



configuration.h	09/02/2016 12:23	C/C++ Header	1 Ko
FirmWare.ino	09/02/2016 12:23	Arduino file	5 Ko
README.TXT	09/02/2016 12:23	Document texte	1 Ko
SerialCommand.cpp	09/02/2016 12:23	C++ Source	6 Ko
SerialCommand.h	09/02/2016 12:23	C/C++ Header	5 Ko
Wiring.png	09/02/2016 12:23	Image PNG	70 Ko

All you have to do is change
"#define Halfstep 8
#define REVOLUTION_STEP 4072 "
By
"#define Halfstep 4
#define REVOLUTION_STEP 2048 "

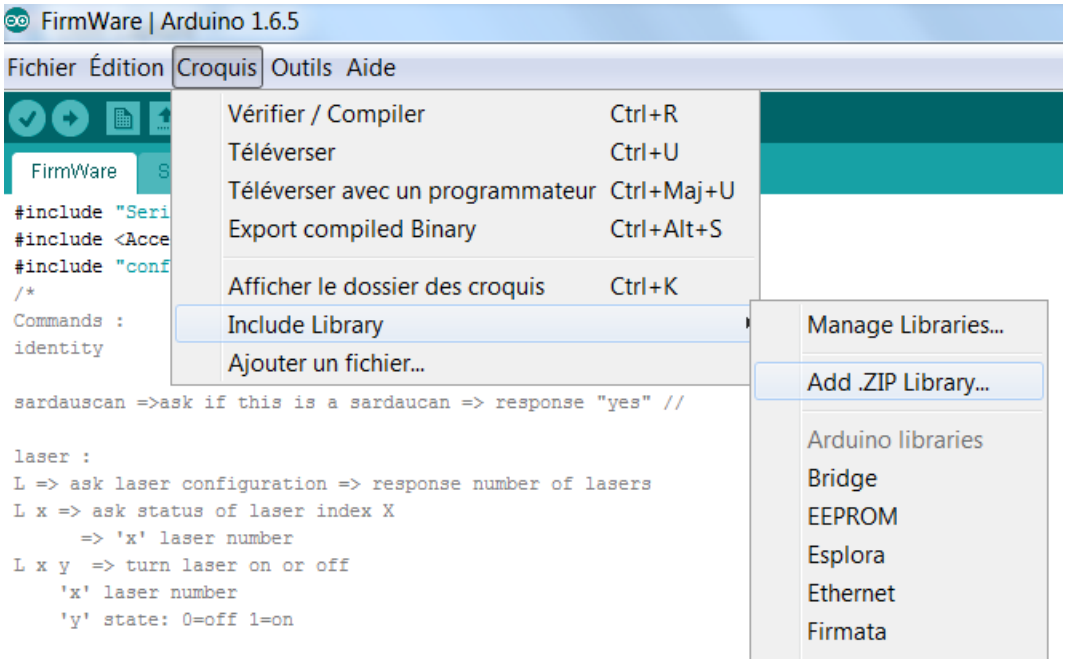


FirmWare | SerialCommand.cpp | SerialCommand.h | configuration.h

```
#ifndef CONFIGURATION_h
#define CONFIGURATION_h

#define FIRMWARE_VERSION "Sardauscan V0.1a"

#define SERIAL_BAUD 115200
//57600

// Motor definitions
#define motorPin1  2     // IN1 on the ULN2003 driver 1
#define motorPin2  3     // IN2 on the ULN2003 driver 1
#define motorPin3  4     // IN3 on the ULN2003 driver 1
#define motorPin4  5     // IN4 on the ULN2003 driver 1

//tips (from Mark Benson)
//If anyone else is having problems with a BYJ48 stepper not doing anything,
//change the HALFSTEP value to 4 & REVOLUTION_STEP to 2048
#define HALFSTEP 8
#define REVOLUTION_STEP 4072
#define STEP_BY_MINMOVE 4  // move by STEP_BY_MINMOVE (to avoid power loss when position is between step)

#define LASER_COUNT 4
#define LASER_PIN_1 A1 //yellow
#define LASER_PIN_2 A2 //orange
#define LASER_PIN_3 A3 //green
#define LASER_PIN_4 13 //blue

#endif
```

Then click on "upload" the
New program is in the
Arduino map

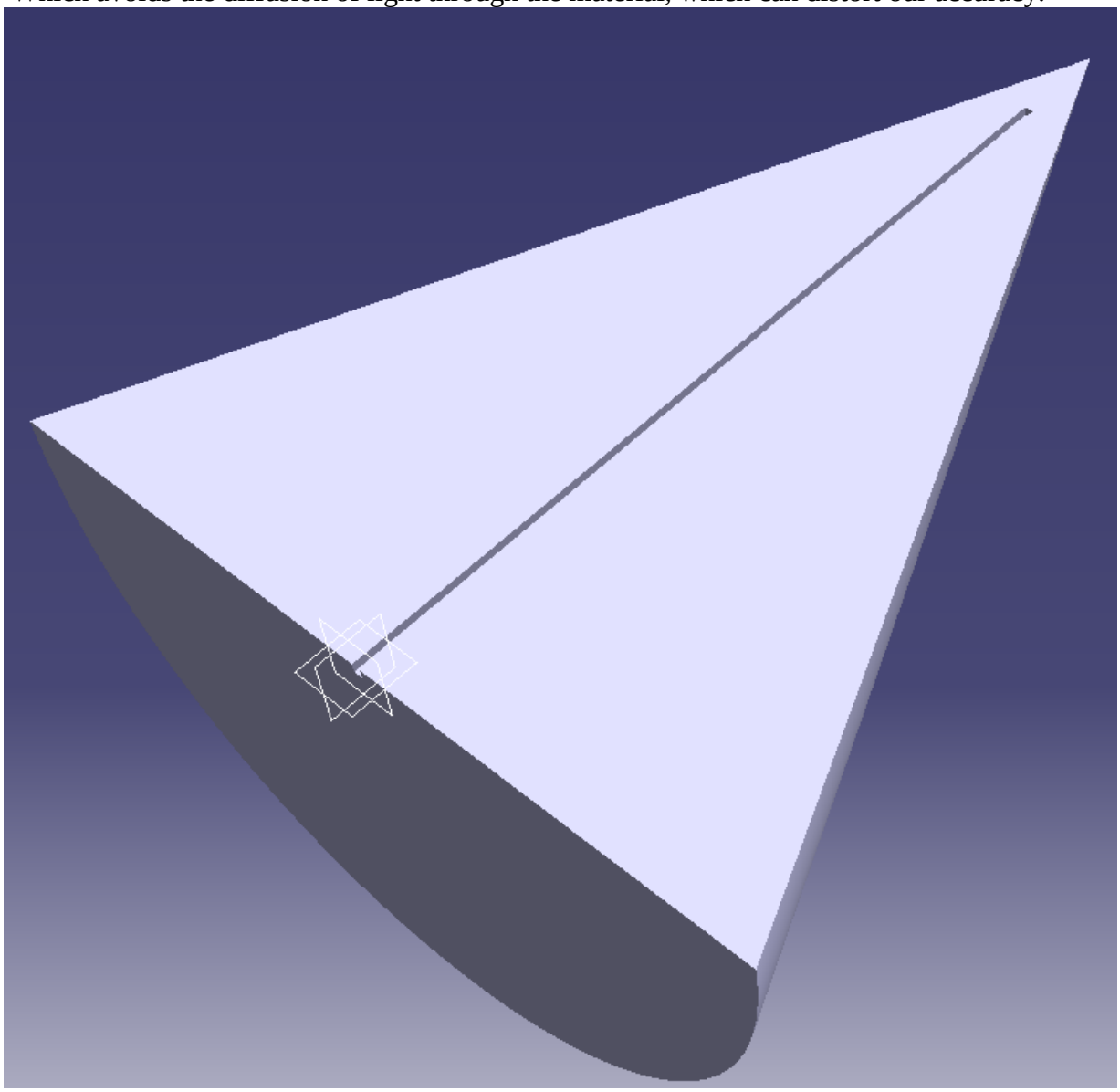
An error message may appear, in fact it may miss a library in your Arduino software. For this go on https://github.com/adafruit/AccelStepper and Download the folder, and rename the "AccelStepper", respecting the Lowercase / uppercase.
Back in Arduino software



Add the AccelStepper library, then upload the new program, the latter Should then settle into your Arduino without worries.
Your interface is now ready for use, but now you have to calibrate the to scan.

Hardware calibration

We will now calibrate the scanner, this task is the longest and the most
Complicated, but it is necessary that the calibration is done with care to obtain a result
Of good quality, so be patient and accurate.
To calibrate I advise you to print in 3D a half-cone (cut in height)
And make a groove in its height (in the direction of the lasers), make the color black, this
Which avoids the diffusion of light through the material, which can distort our accuracy.



You have previously found the center of your turntable if it is not the
You have to try to determine it, you can help yourself with your half-cone and
Your lasers. Place your half cone on the board and pass a laser through its top,
Then turn the tray, if the laser still remains at its top, you have found the
Center of the tray (mark its center for the next uses of the scanner).

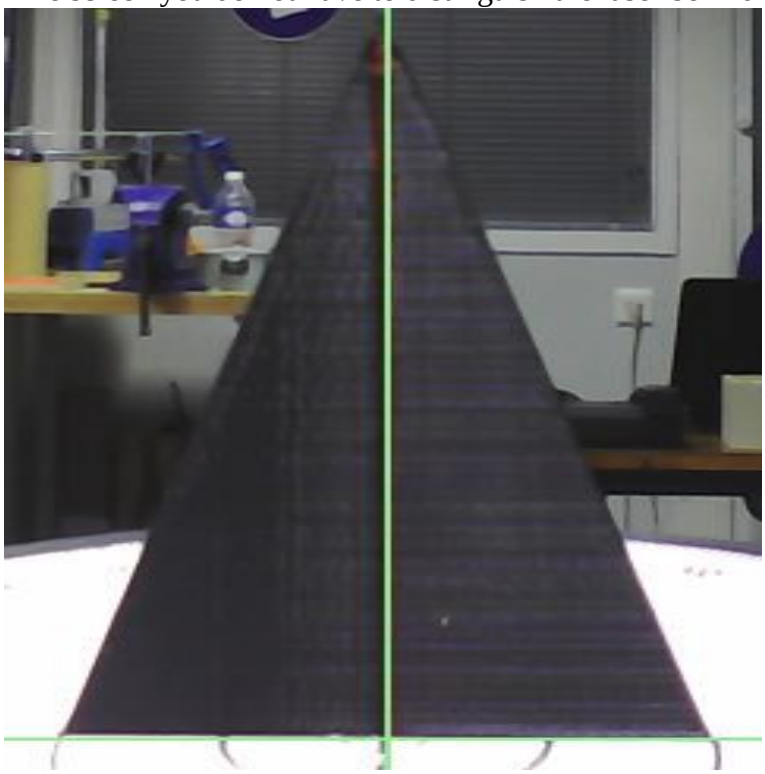
Calibration camera

To calibrate the camera nothing more simple, go to the camera tab, then
In the "Physical calibration" section

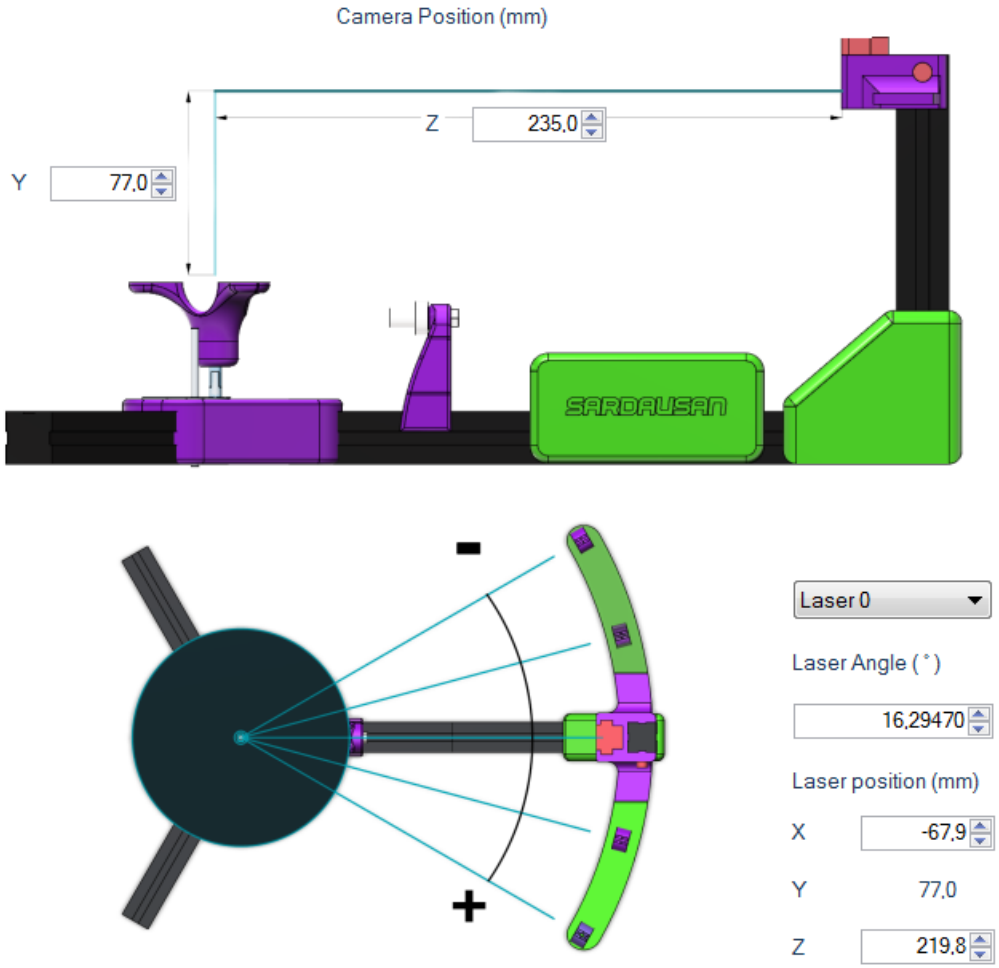
Simply click on the bar
Horizontal and place it on the
Center of your tray, so your
Camera will be calibrated for scans
Future.

Calibration lasers

Laser calibration is one of the most complicated tasks to perform
By its precision. You will need to perform the operation for **each laser independently.** You need your half cone that you will place
Center of the tray, the flat side of the camera side.
Adjust the light beam to make it as thin as possible by turning the
Knob directly onto the laser. Then direct the light beam into the groove of the
Half-cone (the groove is in the center of the tray). On your image you have to
The screen you do not have to distinguish the laser behind the green bar (see below).



Calibration Position
If you have not already done so, you have to calibrate the dimensions of your scanner, that is to say Positions of your camera and your different lasers, for that go in
Tab "calibrate", then in "build calibrate" and enter the coordinates of the camera.

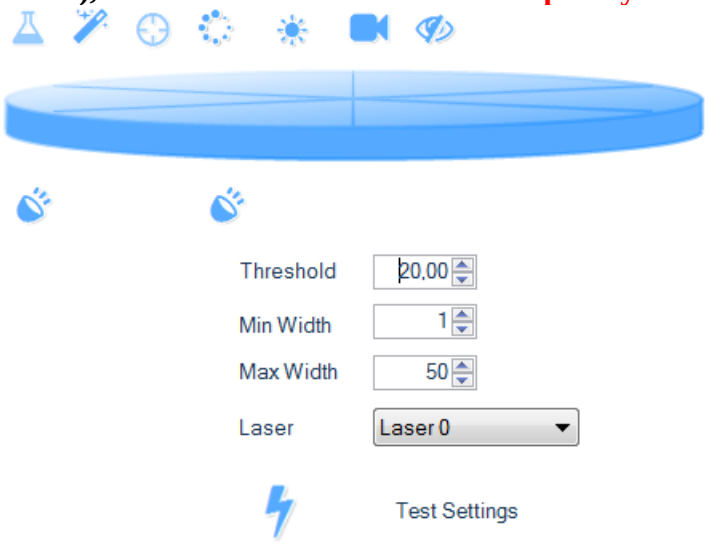


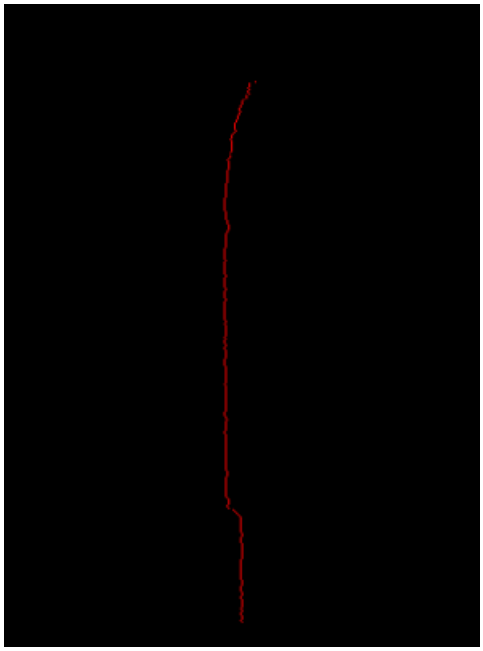
Tip: Do not switch the angle of the laser in accordance with the sign, contact information
Will adjust themselves. Check all laser angles that can move by adjusting
The other lasers.
Tip: print a circular protractor that you'll stick on the shelf

Calibration of the laser image

We will now adjust the image of the laser, ie eliminate possible noise
That the camera records. To do this, go to the "tune" icon and set for the parameters
So as to have a continuous line, and eliminate all isolated points.

The setting is different for each piece of scanned material (material reflection, color), it is essential to do so with THE piece you want to scan.



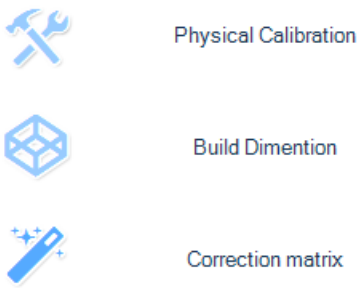


Correction of matrix

Matrix Correction is the final step in calibrating the scanner and the Important. To do this take a piece with a particular shape, a tube with a bar Crossing it in its radius see next photo

Your calibration object must have a simple but irregular shape (eg no Single tube).

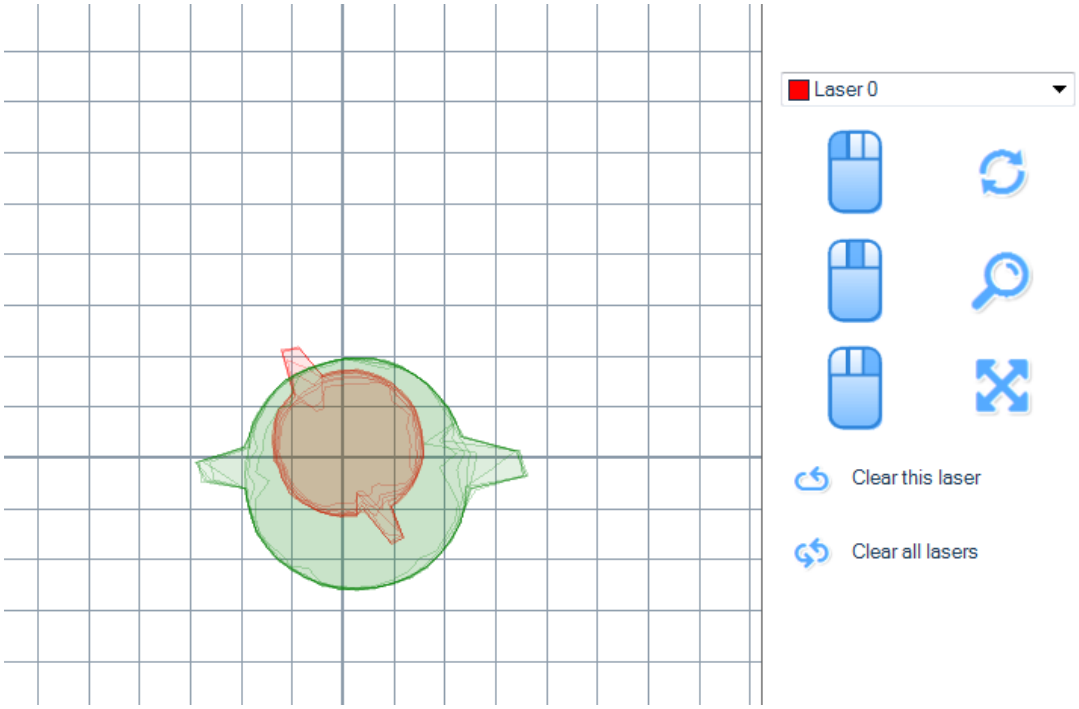
Now go to the tab "calibrate" then in "correction matrix"



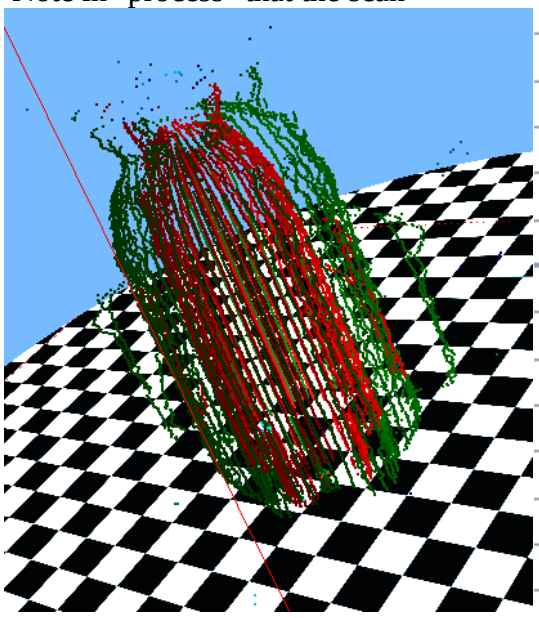
Each setting is specific to your Environment, it's up to you to find it

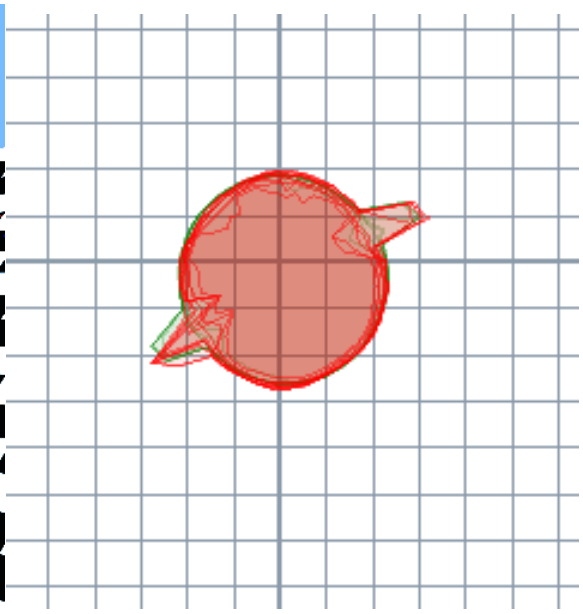
A continuous line without an isolated point.

Place your calibration object in the center of the tray, then launch a quick scan. You
Will get shapes in your window (top view) so you just have to align
Your scans of different colors to make one.
Tip: Go see the "Process" tab and look what color scan respects
Scale the scanned object, do not touch the scan respecting the scale and align the others. Yes
The scans are not to scale but are aligned, a processing function allows
To correct this error.



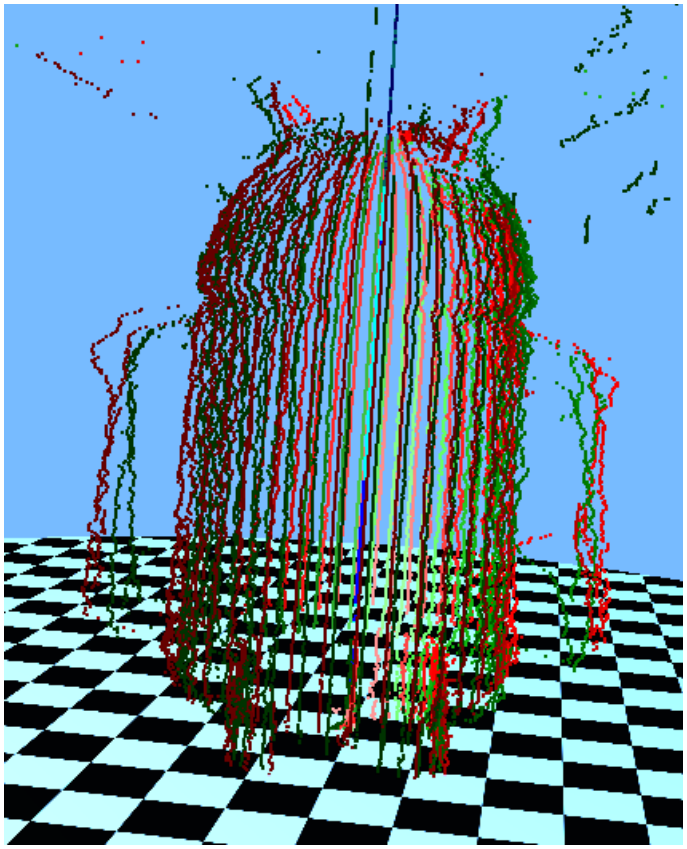
Here the green respects the dimensions, so we must align the red scanner on the green,
Before any manipulation click on "clear this scan" and do it for each scan.
Note in "process" that the scan





Red is more "lean" than the green scan,
And the green scan is on a good scale, it must therefore align with the green scan.

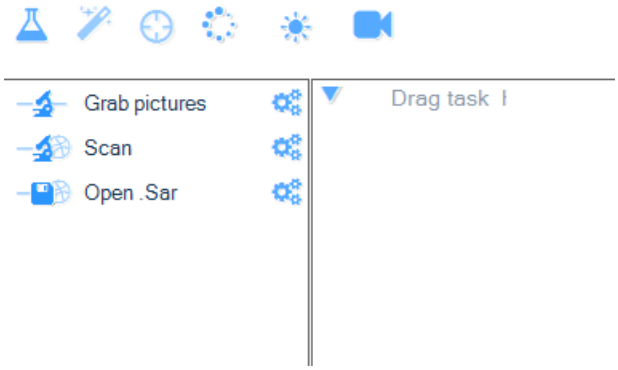
Your scan is now calibrated, we will be able to scan
Treatment to eliminate noise and smooth scans.



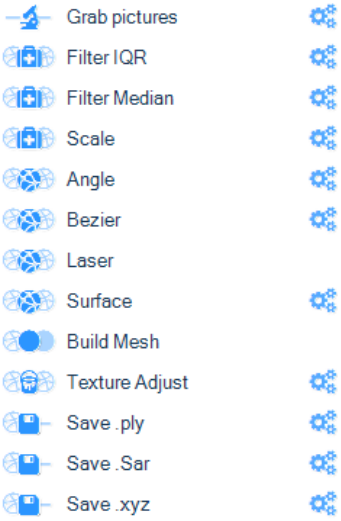
Scanning and processing

Since the calibration of the scanner is done we can now do our first scans.

Go to the "process" tab and then drag the scan icon in the "drag task here" window.



A treatment list will appear on the left side of window, where each icon has a specific function.



 Represents the filter icon and transformation

 Represents the dot smoothing icon

 Represents the icon of build

 Represents the color icon

 Represents the backup icon

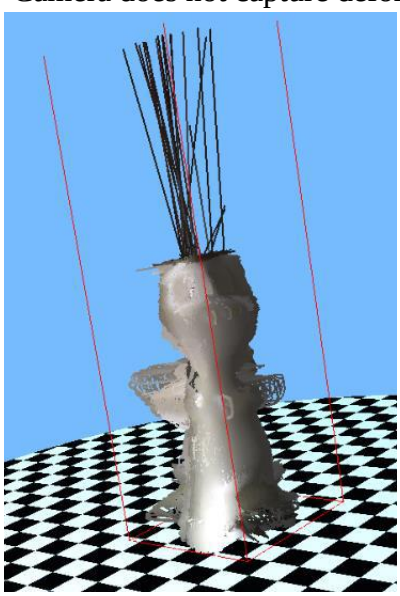
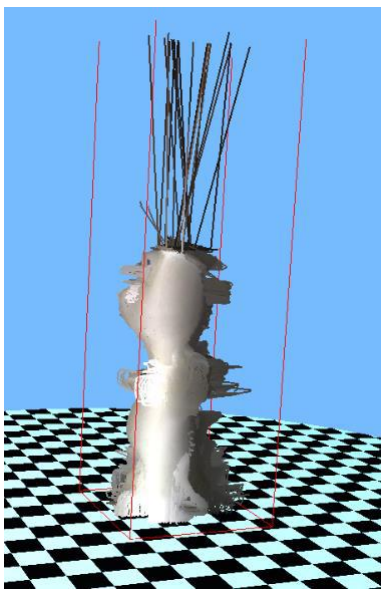
 Represents the parameter icon

The detail of each function is described below.

Filter:
Filter IQR: Removes scan errors with an IQR filter (standard filter in samples
Of probabilities).
Filter Median: Removes scan errors with a "Median filter" (standard filter in
The imagery).
Transformation:
Scale: Changes the scale to (x, y, z).
Smoothing:
Angle: Makes an average approximation of all scanlines based on an angle.
Laser: Makes an average approximation of all scanlines based on an angle.
Bézier: Makes an approximation of Bézier on each of the scanlines.
Surface: Makes surface smoothing with a simplified Laplace.
Color:
Texture Adjust: Adjusts the brightness, contrast and gamma of the texture.
Backup:
Load Sar: Opens a ".sar" file.
Save xxx: Save a file xxx. (Xxx, is for ply, xyz, ect).
The .ply and .xyz files are there if you prefer to use a
Surface reconstruction as "Meshlab" or other.
The .stl and .obj files are there for slicers (.obj files can be imported
In 3D modelers like "3ds Max" or "Blender".
Tip: It is advisable to make the raw scan and save it for later only
Work on scatter cloud processing. You must therefore drag the scan icon
The icon "IO: save.sar" in the right window.
Then import your .sar file into the software to apply the processing

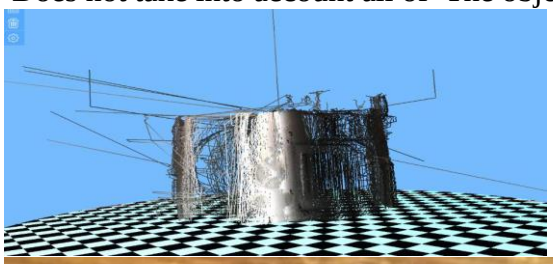
Some scans

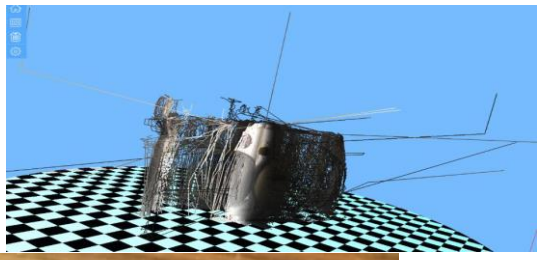
Here are some examples of scans with different objects and treatments
I use only two lasers for the moment, which are placed in position 1 and
2, are the positions closest to the camera.
Scans are very sensitive to the external environment and therefore to the parameters
We must apply to have a correct rendering. So I advise you to be in a room
Not too bright and not to put the scan in direct light and under a lamp, indeed
Too much luminosity decreases the precision of the laser and the outside light can vary and therefore
Change the quality of the scan.
Objects that fit on "legs" or "legs" are not ideal for scanning
Since the ground will then be scanned and can be misinterpreted when processing the objects.
Shiny objects are also not recommended because the material diffuses the laser and
Camera does not capture deformation.

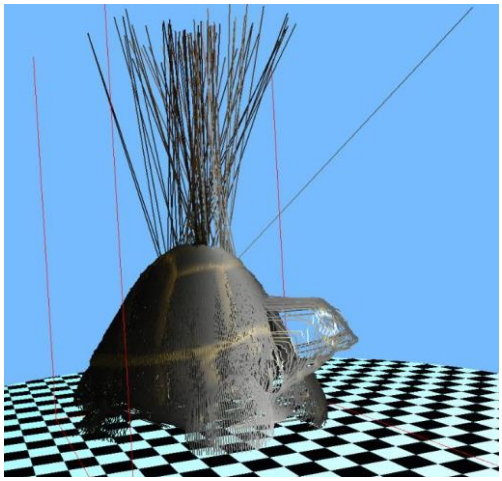
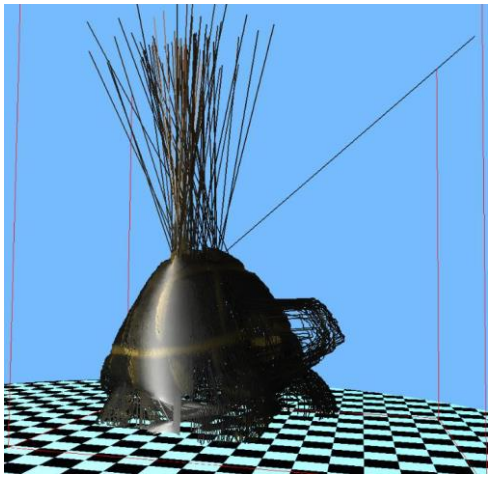




Here it is noted that the dragonnet which is itself printed as a printer material
3D lack of details at the back, hands and tail is missing. We notice
Of pikes above his skull because the scan did not take the points of the skull (no
Visible) and connected directly with spurious points.

For the camera on Notes that it lacks Of detail, this is due to the shape Of the object, in fact when the Turns the lasers Change or little position On the face of the object. And so we Does not take into account all of The object.

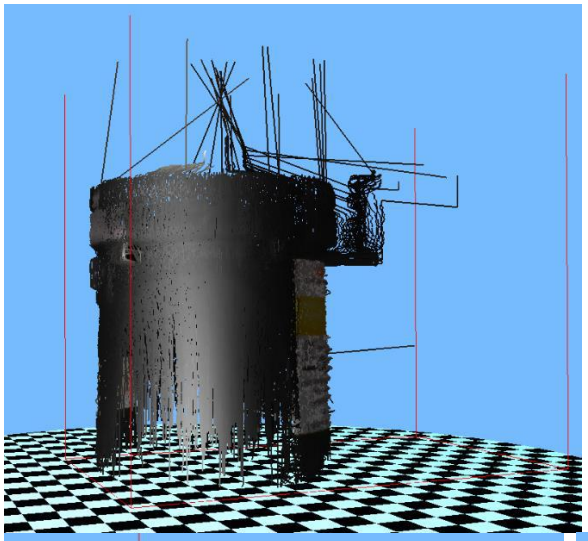


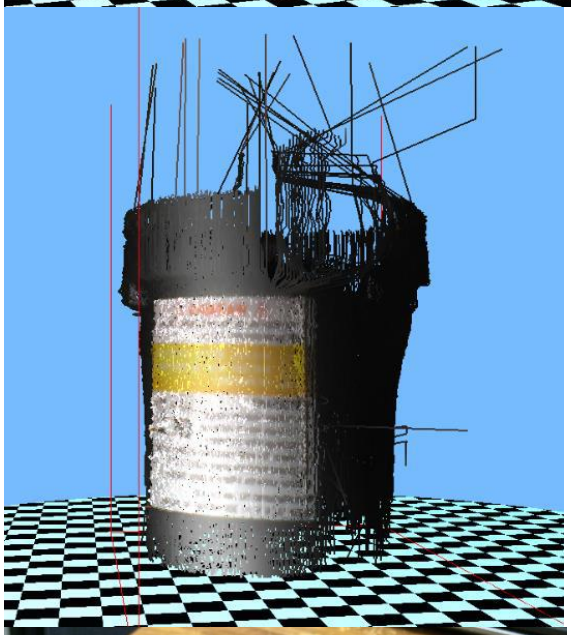


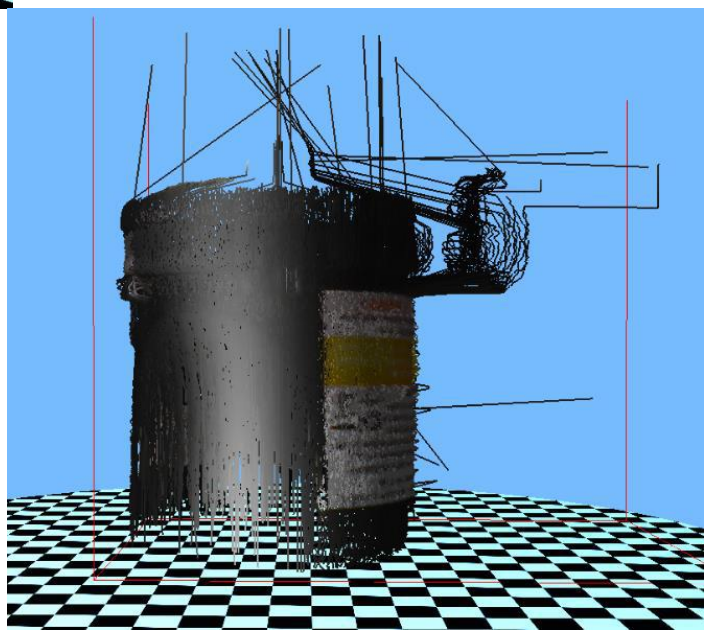



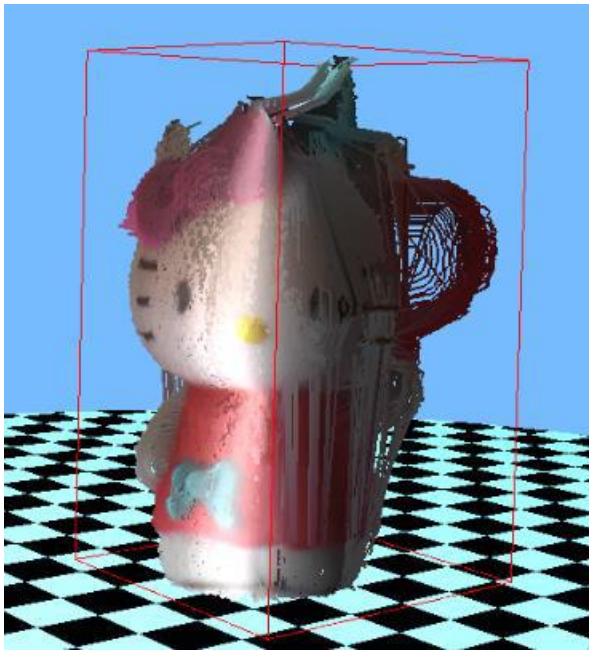
The tortoise is found in the case of the dragon with the top which is not scanned and
Whose treatment links the points of the carapace with parasites. For the head one finds himself
In a particular case where the lasers are too inconsistent with the camera and therefore
Scan not the shape of the head well.

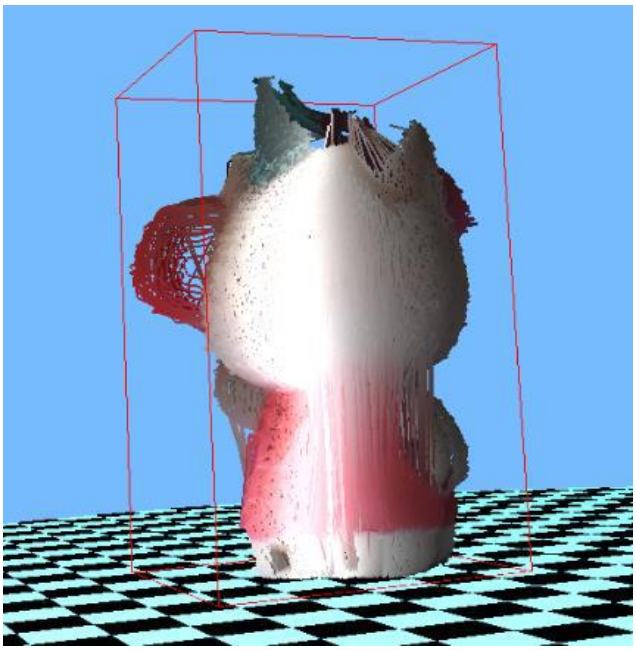
Balloon inflator is a rich object In particular at the level of the grid Thermal evacuation of the object, which did not Could be scanned correctly, due to the Large size of the object, the top of the object Is not scanned.











The Hello Kitty remains an ideal object for this scanner one notices that the details are of Good quality, however here a sudden drop in brightness (a cloud
The sun) modified some points of the scan (white lines on the side of the balloon) and thus Rendering the scanner rendered.