TABLE OF CONTENTS:

Abstract

3D Scanning is a relatively new field,that has a tremendous application properties. The 3D Scanner we make, comprises a normal 2D camera, a line laser, a stepper motor and a platform to place the object. The Output of our project consists of a .asc file, which upon processing can be converted into the printable .stl file. When the .stl file is fed into a 3d printer, the scanned object can be printed out also.

Initially the machine is “started” and the object to be scanned is placed on the platform. On pressing an Esc key, the stepper motor starts rotating, the platform and the object on it rotates too.A line laser falls on the object such that the line passes through the centre of axle of stepper. The stepper rotates for 1.8 degrees and stops and then continues again. The camera is placed facing the object and a photo is taken every time the stepper stops. In this way, around 200 photos of object and the laser line on it is taken.

Every photo of object is processed using OpenCV and the points on the object at which the laser line touched are saved as .txt file. When 200 such different .txt files are obtained, they are applied a suitable transformation with respect to angle of stepper rotation and the points from all 200 files are saved in a single .asc file.

This .asc file consists of the pointcloud of the object. The pointcloud is converted to a 3d object and can be converted to an .stl file using many opensource softwares. We used Mesh lab for this conversion.

Motivation

Ever felt the urge to scan an object and clone it? Well the 3D Scanners in market cost a bomb. What our project does is it scans an object within a short span and reconstructs it and the final output is a solid object file in 3D printable format. Being the first 3D scanner built in CFI using a laser. It's quite simple in all aspects, be it the idea, code or mechanical model. It scans an object quick enough in a single go. This document has a brief description of the ideas, mechanism, code behind this. We have drafted this document in a such a way that even a layman can get the essence of it.

Technical Details

Electrical Connections

Electrical and Electronic Components

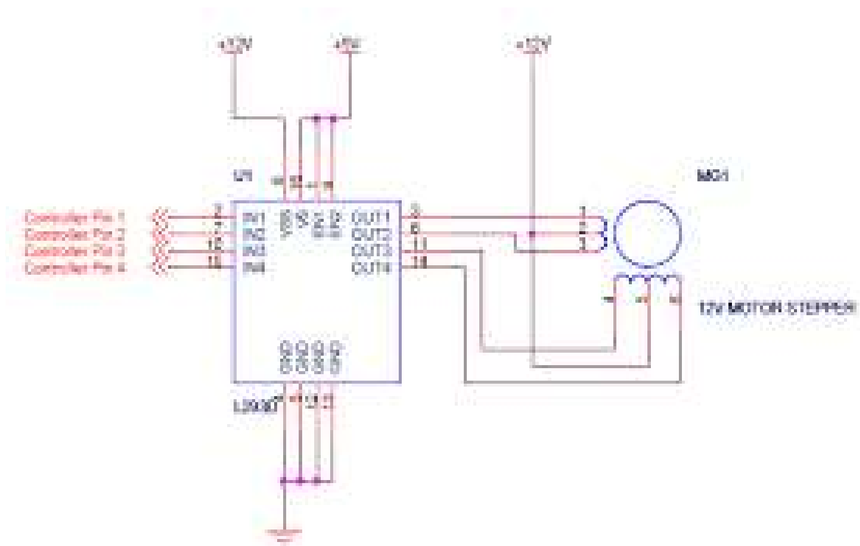
- Stepper Motor-12V
- Line Laser(5mW,650nm)
- USB web Camera
- Arduino Uno
- IC: L293D, 7805
- Soldering Rod and Lead
- Miscellaneous (male to male jumpers, 16 pin IC bases, female berg pins, General Circuit Boards)

Circuit Design

The whole use of circuit is to make stepper rotate. To rotate the stepper we used an L293D IC. L293D is a 16 pin IC, in which four pins, namely 2,7,10,15 are connected to Arduino pins of 8,9,10,11 respectively. Of the other, another 4 pins 3,6 are connected to two ends of a coil1 and 11,14 to another coil2 of a unipolar stepper motor. We are using stepper motor in bipolar condition because bipolar gives a better and accurate step angle.

The 8th pin is connected to a 12v battery while the 16th pin, 9th pin and the 1st pin are shorted and are given a 5v. The remaining four pins,4,5,12,13 are connected to ground.

Since we make use of two different voltages, a 7805 is used to obtain 5v from a 12v battery. Using the Arduino code and this circuit, the stepper can be made to rotate at a step angle of 1.8^0



Coding

General explanation of code

We use opencv libraries in c++ language for image processing. Initially we take a photo of background (i.e, without object).Then as described in introduction we obtain images of object at all angles by rotating stepper motor. All these images are converted into a black and white images. Then they are parted into four horizontal parts of equal size and to each of it we apply gaussian blur,OTSU and binary thresholding. Then image of background is subtracted from each image of object so as to eliminate the laser line which is not on the object. Now we obtain the images with laser line and some noise. We first identify the pixels where there is a change in GRAY value. These may be border points of laser line or noise. By observing the images we made an assumption that the width of noise will not will not be more than three pixels. A detailed explanation of this is provided with this assumption we can eliminate all the noise. We can now identify vertical edges of thick laser line and represent it with a line of single pixel thickness. Then the co-ordinates of these points are stored in a text file. Same method is applied for all 200images.These 200 .txt files containing the required points are then taken in by opengl code which rotates the each point set(from each file) by required angle(obtained from position of stepper) and a point cloud is generated. This is then opened in meshlab software to view the result.

Files Generation

Now that we have 200 files of points, we transform them into a point cloud. We use OpenGL frameworks for that. We use a 'for' loop to open each file and read in points from that file. We then apply angular transformation to those points with respect to y axis. This results in a point line being rotated to an angle it originally is there if stepper had not rotated. Similarly, the whole point cloud of object is created.

To facilitate this, the intake of files and Angular rotation of stepper motor is connected. For this, the files that are generated after the Image processing are named as 1.txt, 2.txt,.....200.txt. This allows to read in 200 files in a 'for' loop using an "i" from 1 to 200.This also tells us angular rotation of stepper.(i.e, 1.8*i)

An array is created and all the points from each file are appended after each iteration. After all the iterations, the array contains all the points of object.After that, the contents of array are simply printed in a .asc file. It can be observed that the contents of array can also be printed in a .txt file also. However .asc is preferred for mesh generation.

The code used for file rendering is ultraextremegl.cpp

Mesh generation

As openGL didn't provide us the freedom to look at the mesh of points from all sides, and extract the mesh to a .stl file which is our final objective.We had to look for other alternatives to view the point cloud and achieve filtering,remeshing,removal of duplicated vertices and trimming of distant point cloud noise .While going through the available open source softwares we found the MeshLab which suits our needs.It provides a wide functionality to work with the point clouds and export mesh in the format we need.

In meshlab we import a mesh from the .asc file which has all the points obtained after rotation transformation of real time coordinates obtained from cv.cpp.That part of the code is in the opengl.cpp. After importing the mesh, we apply the removal of duplicated vertices by going into Filters>Cleaning and Remeshing>Removal of Duplicated Vertices.Then we apply the necessary remeshing and reconstruction functions to get the required output.Going through the online meshlab functions tutorial would be really helpful in this regard.The most used functions by us is Voronoi Filtering and Delaunay Trinangulation.Then we finally export mesh as a .STL file.This format can be readily used to 3D print the scanned object after the dimensional modifications i.e. scaling .

Arduino and Python Codes

We used an arduino to rotate a stepper motor. The arduino is connected with the whole image processing thingy through serial communication between arduino and a python code. The python code links the all the codes that are used. The different cpp codes are compiled and executable files are prepared. These are renamed into suitable names. Now, the python code executes the executable files as the code instructs. This order of flow is executed in python. We make a serial communication between arduino and Python code. Initially python code takes the background by executing back.cpp,executable. Then, it sends a character byte to arduino. The arduino receives the byte. If byte is received in to arduino, arduino sends a signal to stepper, making it rotate 1.8 degrees. After it rotates, arduino sends in a byte to python code again. When the byte is received by python code, it completes the iteration and goes to next iteration or goes out of the loop in tandem with angle of stepper.

The python code is serial123.py

This is the Arduino code

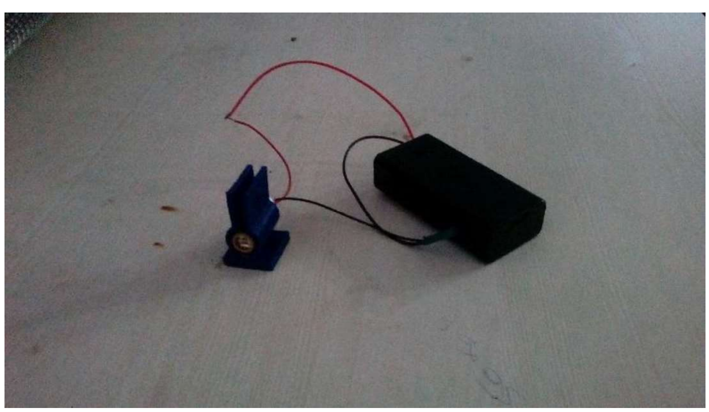
Mechanical Requirements

Mechanical Components and Wooden Box

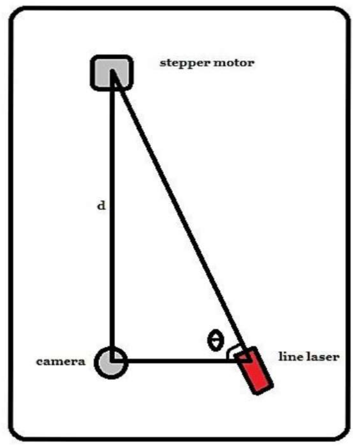
Initially we started of with a mechanical model which would include a servo to turn the platform. But later we realized that it would be better to go with a high torque stepper motor which would provide us accuracy. So we bought a stepper motor,but didn't know the industrial rating of that. We had spent a lot of time figuring out how to run the stepper motor. This would be discussed in detail in the problems section. Finally we had to use an L293D IC to get the one step at a time rotation of the stepper. We needed a model to avoid external light entering the scan area. So we had to build a rectangular box with a lid all covered in black chart paper to avoid noise in our scans. We ehave used standard plywood available in CFI to build the wooden rectangular model. We also erected a T shape wooden platform to hold the laser and camera. The stepper motor is mounted on the base of the box with the help of 4mm thick-4 inch long bolt and nuts. A 3D printed Platform is mounted on the axle of the stepper. The camera is positioned in such a way that it's optic centre(lens centre) is horizontally opposite to the stepper's axle. Also we have provide a 50 cm distance.

Laser Holder

To hold the laser we had to 3D print a laser holder. We used a square base to the laser holder to mount it on the T shape wooden platform Positioned opposite to the platform area.



Overview of mechanical model



d=50cm distance between camera and line laser=d/tan(theta)

Problems

Lasers

We were using a local non -industrial line laser bought from chennai. It had a voltage rating of about 2.5-4 volts. Initially we used two AA cells to provide the required voltage. But due to high current drawn from from the cells they were not sufficient for the purpose. We were also not aware of the precautions of usage of a laser. Laser diodes are semi conductor diodes which get damaged due to several reasons like spike in voltages or currents. Using Zener diodes or Optocouplers to keep the laser safe from voltage surges. We also mishandled the laser by carelessly throwing it down on thee ground many a times. So we managed to screw up a laser because of that. Also the second laser got damaged because of shorting of the positive and negative terminals of laser. So we need to be careful with the free ends of Line Laser.

The Stepper motor

The stepper motor we bought is unrated. It didn't have a voltage rating or the model number. The stepper motor should always be bought with voltage ratings and model number. We bought a 6 wire unipolar stepper motor. We faced many problems in determining the wires of same coil due to discrepancies of multimeter and also due to other unknown problems. Finally we gave in to the pressure of finding the centre tap wires and completely abandoned the unipolar arrangement. We used Bipolar due to ease and reliability.

The Stepper Driver

The problems of stepper motor and driver came in hand to hand. Using a ULN2003, we made a unipolar stepper driver. It didn't work. It could be due to wrong wire selection of the stepper or wrong connections of driver or due to insufficient or high voltage input to ULN2003. We noticed that the ULN2003 gets so heated up. In additional to this, the ATMEGA328P in arduino gets heated up too. We might have fried around 2 to 3 arduinos in this process. The reason for +this, as we figured out is due to unregulated voltage. The limit for current in one of a pin in ULN2003 is 500mA only. In our circuit, it might have been more than that so the ULN2003 and the ATMEGA328P are heating up badly.

We then used a mini ULN 2003 stepper driver which we ordered from online. It worked decent enough, till we realised the driver gets heated up so badly. The good thing is ATMEGA328P works just fine without any issues. We tried resolving the issue with a slightly variant circuit which involved not giving the 12v power to the stepper through the driver but directly. It solved the problem.

But that circuit is unreliable. It doesn't always work properly. Soon enough we discovered the driver had been fried.

So we shifted to using a Bipolar connection using L293D. That solved the problem and the circuit worked. But we still had problems.

Electronics and Soldering problems

After the ideas of using ULN2003 and a stepper driver both cupped, we finalised a circuit using L293D.We first tried out the circuit on breadboard. After becoming sure of its working we decided to go ahead and make a GCB. But apparently due to some problem in soldering the same circuit which worked on breadboard did not work on GCB. We repeatedly tried to get the problem sorted. In fact we soldered the GCB 5 times but not a single time did it work. We are yet to resolve this problem and hence are still using a circuit on breadboard.

External light, reflecting surfaces problems

The problem with our 3D scanner is that it loses its efficiency in the presence of external lighting. We have tried to reduce this problem by covering the insides of the entire structure in black chart. Also, if the object has reflecting surfaces then the scan won't be correct.

Problems with the cv codes

In writing codes for image processing we mainly faced problems in detection of laser light accurately, removing noise after laser detection. For the laser detection we first tried to use the (red)colour detection technique but later dropped it because it is not that accurate and the object may also be red in colour. Later we tried to use hsv color space which too didn't work because we couldn't get a perfect boundary of laser light. After that we tried applying gaussian blur and binary thresholding to whole image(black and white). The results were better but not that good. So we finally we tried applying same method for horizontal slices of image as explained in the codes part. For the background subtraction we tried subtraction of images directly which gave good results with some disturbances which we removed using proper algorithm.

Problems with mechanical model and positioning of camera,laser and platform.

The major problem here is to properly fix the laser holder and camera at assigned positions and at assigned angles. Each time we fix the laser holder we have to make sure that the laser line falls on the stepper axle and that the laser line is perpendicular(vertical) and we have to measure the angle of laser holder. To avoid this finally we made a slotted wooden holder to hold them in fixed positions. One more problem was to keep all the wires and components away from the laser light because they may cause some noise in the scans. We made a small box at a corner to place all the circuitry to solve this.

codes

Arduino code

To get the background image

To adjust the light conditions

To get the point set

python code

Acknowledgements

A lot of people have helped us enroute the completion of this project. We would like to thank all those people. Firstly, we would like to thank the entire team of CFI for giving us this opportunity. We would like to thank our mentor Aboo backer for guiding us and constantly motivating us. We would also like to thank K.Rajiv,G.Harivenkatakiran and K.Kishore for their constant encouragement and support. We would also like to express our gratitude to the CVG team especially R.S.Nikhil Krishna and Jeevan for helping us with the image processing and fixing the bugs in our code.

Overall, it was an amazing experience working in this project. It instilled in us enormous potential for perseverance and helped us explore the spirit of teamwork.

References

1)OpenCV Documentation by Abid Rahman

2)Open CV documentation

3)StackOverflow

4)FAQ portal of CFI

5)Instructables