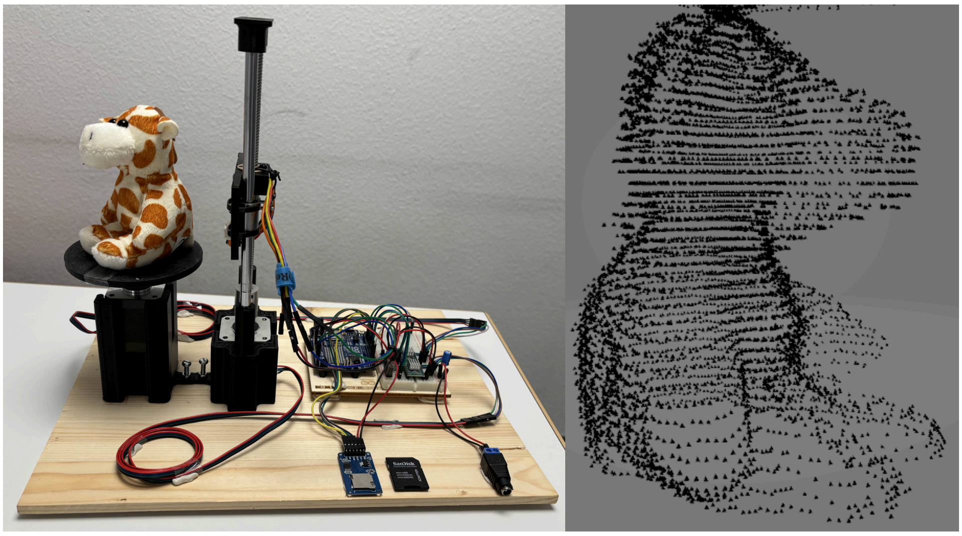
University of Basel,

62025-01 Computer Architecture HS24

Simple 3D Scanner



Dario Manser

Doga Oglakcioglu

Elagkian Rajendram

Nillan Sivarasa

21 January 2025

Abstract

This report discusses the development of a simple 3D scanner. The core idea is to approximate the subject's geometry by measuring the distance between the center of the object and its outer surface for a number of points around its perimeter and repeating this process at multiple heights to assemble the object layer by layer. To overcome memory limitations, data points were stored on an SD card (12, 13) for later processing on a computer. Due to limitations imposed by physical laws as well as hardware constraints, the final device suffers from severe restrictions regarding the subject's geometry and surface properties; nevertheless, it produces satisfactory results for subjects that adhere to these limitations, rendering the project successful as a whole.

1. Goal

The goal of this project is to develop an Arduino Uno R4-controlled (3) 3D scanner based on the VL53L1X distance sensor (15) and powered by two stepper motors (14). The scanner should be able to capture small subjects with dimensions of up to 11.5 cm in height and up to 10 cm in diameter with sufficient accuracy to recognize the subject in the resulting object file. We aim to place particular focus on the hardware side of this project and allow the software to remain relatively simple by avoiding data processing or complex algorithms, because our deeper goal behind this endeavor is to learn about programming microcontrollers, controlling hardware devices, and working with limited compute power.

2. Approach

2.1. Idea

The concept involves a rotating platform that holds the object to be scanned, while the distance sensor remains fixed at a single point. As the platform rotates, the sensor measures distances at various angles to collect data. After each full rotation, the sensor is moved upwards to capture a new horizontal slice of the object. This layer-by-layer scanning approach allows the reconstruction of the object's full 3D geometry. By keeping the sensor stationary and rotating the object, the design simplifies the mechanical setup and ensures consistent alignment, enhancing the accuracy of the captured data.

2.2. Construction

The main structure is based on an existing 3D model by "Super Make Something" (11), which we modified using the computer-aided design (CAD) software *Fusion 360* (4) to adapt it to our needs. The 3D model (fig. 1) consists of four separate components: the base, providing structural support and acting as an attachment point for further components; the rotating platform, onto which the subject is placed and which rotates around its vertical axis; the carrier, which holds the distance sensor; and the stabilizer, which caps the tower along which the carrier moves.

Figure 1: CAD model of the scanner

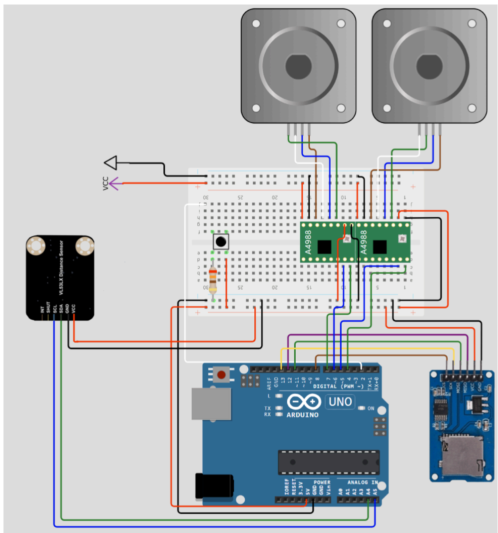


Figure 2: Circuit (1)

Going forward, we will separate our device into four distinct modules, each of which is explained in detail in the following subsections.

2.2.1. Base

The entire project is mounted on a wooden base, which holds the Arduino Uno R4, a breadboard for wiring, and the remainder of the device.

2.2.2. Table

The table holds and rotates the subject during the scan. Structurally, it consists of a hollow cuboid into which we fit a Nema 17 stepper motor such that its shaft rises above the cuboid. We then mount a disk to the shaft onto which we place the subject. The motor is connected to the Arduino via the motor driver (2) and powered by an external power supply (fig. 2). To control the motor, the firmware (9) exposes a collection of preprocessor defines in the motors.h header.

2.2.3. Tower

The tower consists of three parts (fig. 1); the threaded spindle (7) with lead screw, that moves the carriage vertically; metal rods (6) with ball bearings (8), stabilizing the carriage; carriage, onto which the distance sensor is placed.

The distance sensor is fixed onto a carriage that moves vertically along smooth guide rods and it is directly connected to the arduino (fig. 2). We used bearings on these rods to make sure the vertical motion is smooth and precise, reducing mechanical play and improving the overall scanning accuracy. The vertical movement is powered by a lead screw that is connected by a shaft coupling (16) to the stepper motor called Nema 17 in the same way as the other stepper motor for the table (fig. 2).

2.2.4. Storage

Since the Arduino Uno R4 has only 32 kB of SRAM, which is not enough to store all the points, we store the object on an SD card via an SD card module connected to the Arduino. To minimize constant IO overhead, we use a 12 kB buffer to temporarily store data before writing it to the SD card

2.3. Process

While assembly of the hardware went without major problems, the initial results produced by the scanner were often inaccurate, hardly recognizable, and sometimes, for lack of a better word, pure chaos (fig. 3).

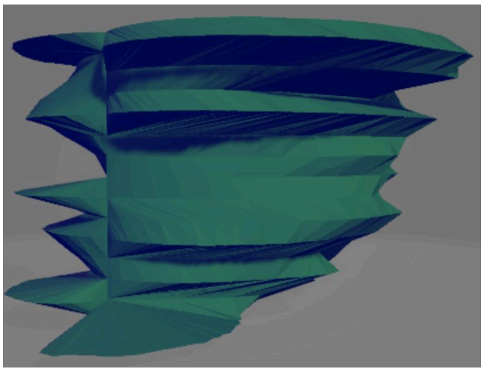


Figure 3: first 3D image of a "can"

Some of the underlying problems were quickly identified and addressed, such as the subject not being centered on the table or the distance between the scanner and the table using a wrong value. This allowed us to capture simple cylindrical shapes, but scans of more complex geometry still produced poor results. At this point we suspected our algorithm for determining the Cartesian coordinates of measurements to be the problem:

$$x = r\cos(\theta), \text{ and}$$
$$y = r\sin(\theta),$$

where r is the radius, determined by subtracting the distance between the sensor and the center of the table from the measured distance, and θ is the angle of the table. However, we then determined that the real reason for so many disappointing results was two-fold: first, subjects whose shapes are dominated by convex features are impossible to capture using a single point of view, since the convex geometry inevitably obstructs parts of the subject's surface. Second, the distance sensor's default field of vision spans 27°, leading to a "blurry vision" that makes small details impossible to capture. We tried decreasing the sensor's field of vision, but were unable to do so, so instead we resorted to decreasing the distance between the sensor and the subject by moving the table closer to the tower. This finally allowed us to accurately capture non-reflective subjects, so long as their shape wasn't dominated by convex geometry.

2.4. Results

The scanner was tested on various small objects, such as figurines and geometric shapes. The resulting 3D point clouds demonstrated enough accuracy and resolution.

The use of a fixed distance sensor and a rotating platform proved effective for certain geometries. However, minor discrepancies were observed, primarily due to mechanical tolerances and environmental factors. Therefore the scanner is well-suited for educational and hobbyist purposes, with potential for further refinement to enhance performance. To get a better image, we could use a more advanced sensor that measures distances with greater precision and captures finer angles. Additionally, if higher quality is desired, complex algorithms in software like MeshLab (10) can be used to filter and refine the point cloud, producing a smoother and more detailed final model.

When the geometry met our requirements, we were able to produce a good image (fig. 4) that satisfied our goal.

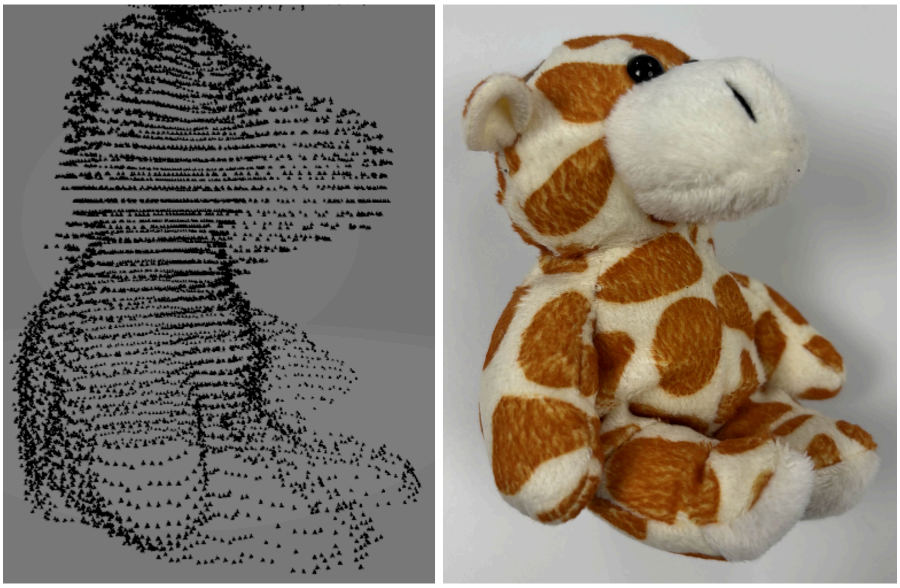


Figure 4: 3D Scan of a plush toy and the scanned object for the comparison

3. Closing

Right from the start, we made it a priority to buy the components early. Doga and Elagkian took charge of researching each part, carefully checking specifications, and handling the orders. While waiting for everything to arrive, we jumped into design work, modifying an existing model in Fusion 360 to fit our project's needs and running simulations to refine it. Meanwhile, Dario and Nillan tried to implement the algorithm to convert the raw sensor data into a 3D point cloud. Once the parts had arrived, we mostly worked together on everything. We encountered various challenges during the development process. There were moments when motors stalled or moved erratically, and sensors returned inconsistent readings, leading to a cycle of adjustments and testing. Arguably our biggest mistake was assuming we knew the source of our problems and waiting to address until two days before the deadline, only to discover that the problem was much bigger than anticipated, resulting in a stressful sprint before the finish line.

References

1. (n.d.). https://wokwi.com/projects/420619003261818881
2. *A4988 Stepper Motor Driver*. (n.d.). https://www.amazon.de/dp/B07C2V9GWC?ref=ppx_yo2ov_dt_b_fed_asin_title&th=1
3. *Arduino Uno R4*. (n.d.). https://www.amazon.de/dp/B0C8V88Z9D?ref=ppx_yo2ov_dt_b_fed_asin_title
4. *CAD Software Fusion*. (n.d.). https://www.autodesk.com/ch-de/products/fusion-360/overview?term=1-YEAR&tab=subscription
5. "Chenqi". (2021, June 18). *VL53L1X Distance Ranging Sensor*. CQRobot-Wiki. http://www.cqrobot.wiki/index.php/VL53L1X_Distance_Ranging_Sensor_SKU:_CQRWX00744US
6. *Metal rods* (n.d.). https://www.amazon.de/dp/B08XYLG9J4?ref=ppx_yo2ov_dt_b_fed_asin_title&th=1
7. *Threaded spindle* . (n.d.). https://www.amazon.de/dp/B0C5M5P2LT?ref=ppx_yo2ov_dt_b_fed_asin_title&th=1
8. *Ball bearings*. (n.d.). sdhttps://www.amazon.de/dp/B0D92378F2?ref=ppx_yo2ov_dt_b_fed_asin_title
9. Manser, D., Oglakcioglu, D., Rajendram, E., & Sivarasa, N. (2025). *rl3ds* [Firmware for a simple Arduino-controlled 3D scanner]. GitHub. https://github.com/thechnet/rl3ds
10. *MeshLab*. (n.d.). https://www.meshlab.net

11. Schepelmann, A. ". M. S. (2016, March 14). *DIY 3D Scanner* [STL files for the 3D scanner]. Thingiverse. https://www.thingiverse.com/thing:1413891/files
12. *SD Card*. (n.d.). https://www.fust.ch/handy-pc-tablet/speichermedien/speicherkarte-sd-karte-microsd-karte/sandisk-ultra-sdhc-32gb-/p/10668079
13. *SD TF Card Reader*. (n.d.). https://www.amazon.de/dp/B06X1DX5WS?ref=ppx_yo2ov_dt_b_fed_asin_title
14. *Stepper Motor Nema 17*. (n.d.). https://www.amazon.de/dp/B0B93HTR87?ref=ppx_yo2ov_dt_b_fed_asin_title&th=1
15. *VL53L1X Distance Sensor*. (n.d.). https://www.amazon.de/dp/B07F3TV3G4?ref=ppx_yo2ov_dt_b_fed_asin_title
16. *Shaft coupling*. (n.d.). https://www.amazon.de/dp/B0CP76451B?ref=ppx_yo2ov_dt_b_fed_asin_title&th=1

Appendix