Real time point cloud comparison and capture for a 3D printing process

Team Member	Ankush Kumar Mishra
Project Type	Solid Modelling/Academia

Vision

The fourth industrial revolution or Industry 4.0 envisages a manufacturing system where the differences between the physical and the digital world is blurred. A simulation of the entire physical process is required to be done digitally. A digital twin of the physical process needs to be created for Industry 4.0. A widely accepted definition of a digital twin is "an integrated multi physics, multiscale, probabilistic simulation of a system/product that uses best available physical models, sensor updates, product history to mirror the state of its corresponding twin" [1]. Once the digital twin is created, the manufacturing process needs to be monitored by sensors. This digital twin and the product of the process is then compared to see if the manufacturing standards are met. The process has been explained in Fig. 1. We have the process monitoring of the manufacturing bed through the sensor. We simultaneously compute the digital twin on the computing bed and then we do the comparison to determine the differences between them.

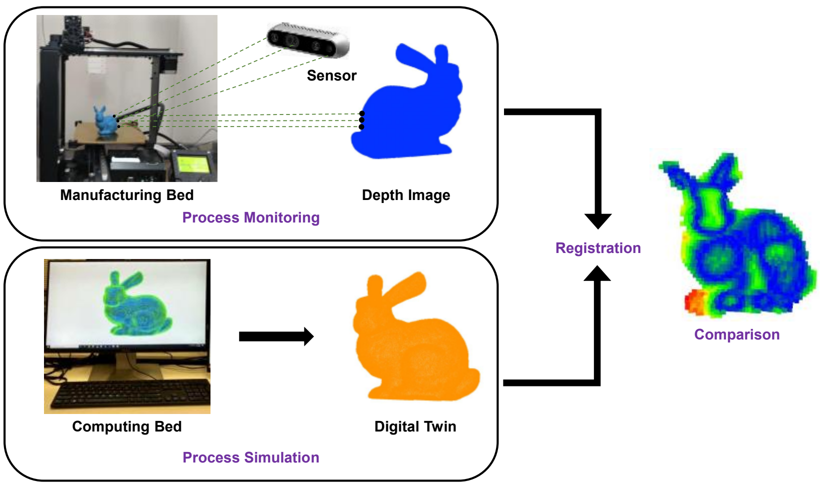

Figure 1: Real time comparison of the manufacturing process

Goal

With this larger vision in mind, we focus on characterization of the geometric aspect of a 3D printing process. We capture the point cloud information of the 3D printed object and compare it with its digital twin. The entire process can be broadly divided into the categories of i) point cloud filtering ii) point cloud registration iii) point cloud comparison. The filtering is done on the actual point cloud of the printing process while the registration and comparison require the both the actual point cloud and the theoretical point cloud (digital twin). Also, the larger goal is to do the entire process in real time.

Manufacturing Bed

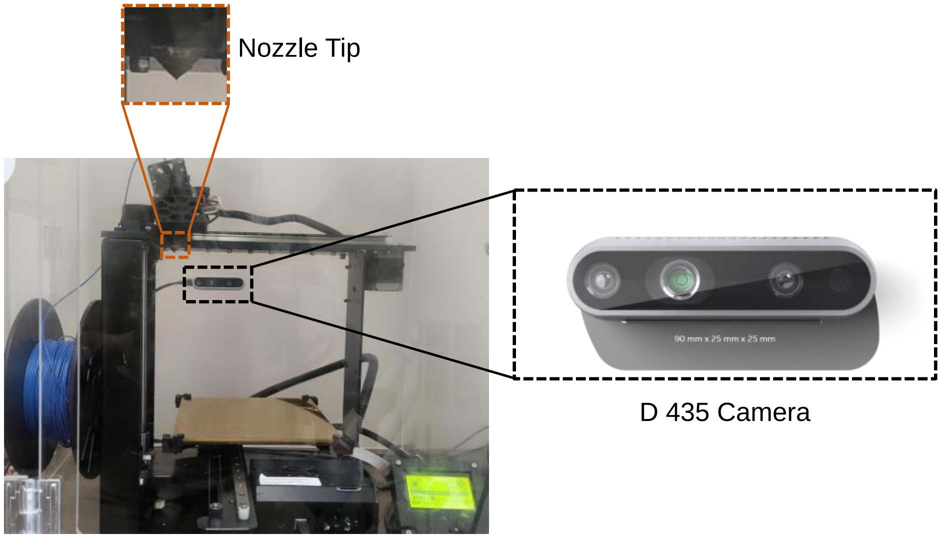

Figure 2: Manufacturing Bed

The setup for the manufacturing bed is shown in Fig. 2. The manufacturing bed consists of a MAKEGEAR M2 3D printer with dimensions of 8”x 10”x 8”. The nozzle diameter is of 0.35 mm and the printer material is ABS. For our sensor we use an Intel RealSense D435 depth camera. The camera uses stereo vision depth technology to capture the depth [2]. It consists of a left imager, a right imager, and an infrared projector. The depth is calculated by correlating the pixels of the images captured by the left and right imager. It then uses the shift of a point between the two images to capture the depth.

Methodology

As discussed before there are three important tasks that needs to be performed in the process. We start with filtering of point cloud data. For this we use a **software** called CloudCompare [3]. We capture the point cloud information and use this software to filter out the point cloud. The result of this filtering is shown in Fig. 3. Simultaneously, on the computing bed the digital twin of the product being manufactured needs to be created. We look at the literature where a framework exists for creating a voxel representation of the CAD file of the object using voxelization algorithm [4] and the create a G-Code and voxel printing file from the voxelized model [5]. The model can be tuned to include sparsity at different levels. The G-Code generated is used for the 3D printing process. The simulated point cloud or the digital twin is shown in Fig. 4.

Now, we have the simulated point cloud as well as the filtered point cloud. However, the filtered point cloud is just the object captured from one view. In future there is a scope to extend this work by placing multiple cameras. We need to perform the point cloud registration of point clouds captured from multiple cameras. The registration needs to be done between the filtered and simulated point clouds. Then the comparison is made between them using the CloudCompare software.

Image of the target object

Raw point cloud data with surrounding noise

Filtered point cloud data

Figure 3: Steps for extraction of filtered point cloud information from raw point cloud

Figure 4: Simulated Point Cloud

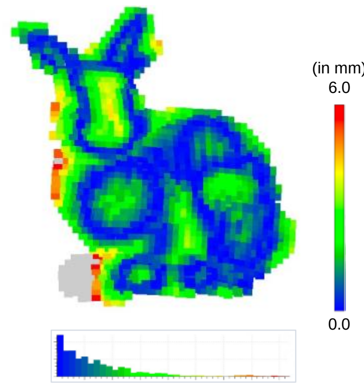

Figure 5: Results of the comparison

Results

The results are shown in Fig. 5. Most of the error is less than 1 mm. The maximum error observed was 6 mm. We see a grey patch near the foot region. This is because the error observed there was more because of a registration issue rather than an actual geometric aspect error and thus has been removed.

Continuing Work

We have continued work on this to make the entire process real-time. We have started with the filtering process and analyze the filtered data to come up with depth and rgb masks on the raw data. The results are shown in Fig. 6. Other than this, we also have setup a benchmark for comparison. We used a TLS (Terrestrial Laser Scanning), which has at least 10x less error at same distance, to scan the object from multiple view and compare it with the digital twin. Multiple views and the comparison results is shown in Fig. 7 and Fig. 8.

Relationship to Course Content

During the course, we came across the concepts of voxelization. The model used for 3D printing is essentially a voxelized model. The resolution can be setup to induce sparsity across the height of the model. The model is dense at top and bottom and less dense in the central region. We applied the concepts learned on voxelization in class to apply this. Moreover, the paper [5] that we referred to generated g-code and voxel printing file uses marching cube algorithm to identify the boundaries of the voxelized model.

Raw Point Cloud | Manually Filtered Point Cloud | Filtered using Distance Mask and average RGB mask

Figure 6: Ongoing work on real-time filtering of data

Front View | Back View | Left Side View | Right Side View

Figure 7: Multiple view captured from TLS camera

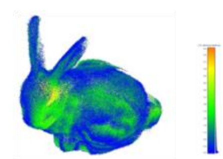

Figure 8: Comparison results using TLS camera

<u>Summary</u>

The success of the project would be to clearly develop the methodology for comparison between real manufacturing product and its digital twin. We were able to use the information in literature and advance it to come up with this framework. The comparison is done only of final product, but inline monitoring can also be done to get a feedback as we go. The registration process in the software was a challenge as the point clouds needed to be arranged very closely for correct comparison. However, it was mitigated by repeating the process multiple times.

The problem will stack up as we try to do the process in real-time. Our focus is now on filtering, and we plan to use ML algorithms of clustering and connected components for enhanced filtering. The challenge is going to be on the registration part of the problem which would require some efficient thinking.

If I were to do this project again, I would probably focus on making it real time right from the start. The cool part of it would be to use various ML algorithms to achieve things in real-time. Nevertheless, I have learned a lot while working on the project. I was completely new to the concepts of voxelization before learning it. The 3D printer, the depth cameras as well as the CloudCompare software was all new to me when I started the project and I have developed a good understanding of these stuff now.

References

1. E. Glaessgen, D. Stargel, The digital twin paradigm for future nasa and us air force vehicles, in: 53rd AIAA/ASME/ASCE/AHS/ASC structures, structural dynamics and materials conference 20th AIAA/ASME/AHS adaptive structures conference 14th AIAA, 2012, p. 1818
2. A. Grunnet-Jepsen, J. N. Sweetser, P. Winer, A. Takagi, J. Woodfill, Projectors for intel® realsense™ depth cameras d4xx, Intel Support, Intel Corporation: Santa Clara, CA, USA (2018).
3. E. R&D, T. ParisTech, Cloudcompare (version 2.6. 1)[gpl software], Telecom ParisTech (retrieved from http://www. cloudcompare. org/) (2015).
4. G. Young, A. Krishnamurthy, Gpu-accelerated generation and rendering of multi-level voxel representations of solid models, Computers & Graphics 75 (2018) 11–24.
5. S. Ghadai, A. Jignasu, A. Krishnamurthy, Direct 3d printing of multi-level voxel models, Additive Manufacturing 40 (2021) 101929.