**Memo**

To: Professor Pisano

From: Juan C. Vecino, Tristen Liu, Daniellia Sumigar, Ramy Attie

Team: 14

Date: 11/19/2023

Subject: Scan It! Pack It! Arrange It! - First Prototype Testing Report

1. **Setup Summary**

The scanning test required a smartphone camera and a laptop acting as a cloud computing server. The smartphone camera was used to record a 360 degree video of an object, which was then uploaded onto the laptop. This step was done in advance, due to the slow loading times of video transfer between the smartphone device and the laptop. Then, the laptop processed the video and output images to the instant-ngp executable. Finally, the NERF was generated using the processed files and the object was manually extracted from the environment to create a .obj file.

The packing algorithm test was performed using python scripts taking in manual input dimensions of the container and the objects for packing. Three total test cases were designed to observe the performance of the algorithm: (1) all objects are expected to fit in the container, (2) all objects are expected to fit in the container with no trim, (3) some objects are expected to fit in the container. The python scripts generated a 3D schematic of the optimal packaging arrangement, highlighting each object with a different color.

1. **Experimental Results**
   1. **Scanning**

Each testing video was around 30s long, and 2 frames were taken per second. Starting from a 30s video using 2 FPS, a NERF model was generated within 1 minute and 30 seconds. The rectangular prism test was conducted using a cardboard box, and a 3D model that retains its original dimensions was successfully extracted from the model.

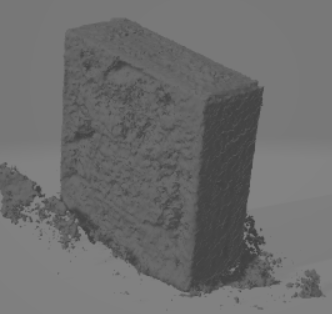


Figure 1. The cardboard box and the generated 3D model

The irregular object test was conducted using a white tub with a lid. Although this test retained the general shape of the object, some features were lost due to noise and the featureless semi-reflective material of the object.

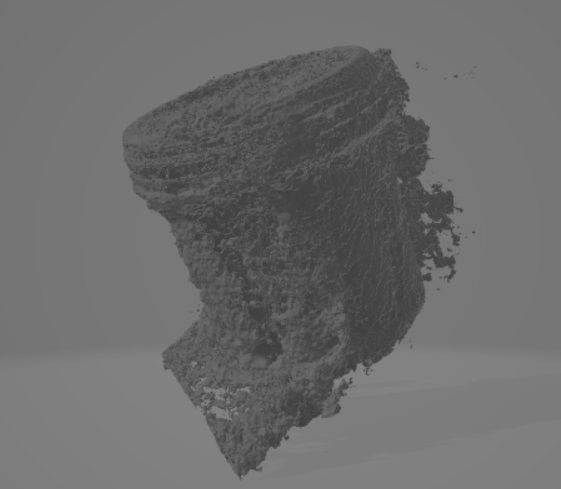


Figure 2. The white bucket and the generated 3D model

* 1. **Packing**

After running test case 1, the packing algorithm was able to produce an arrangement within 0.23 s. Depending on the order in which the objects to pack were specified, the algorithm generated a slightly different result, although not drastically different. For test 2, the algorithm produced the packing arrangement in 0.06 s. This was the scenario where all objects were expected to fit in the container with full space utilization. In test 3, the case where not all objects were expected to fit was examined. The algorithm took 0.20 s to produce an arrangement. It was observed that the object that was too large for the specified container was not included in the schematic.

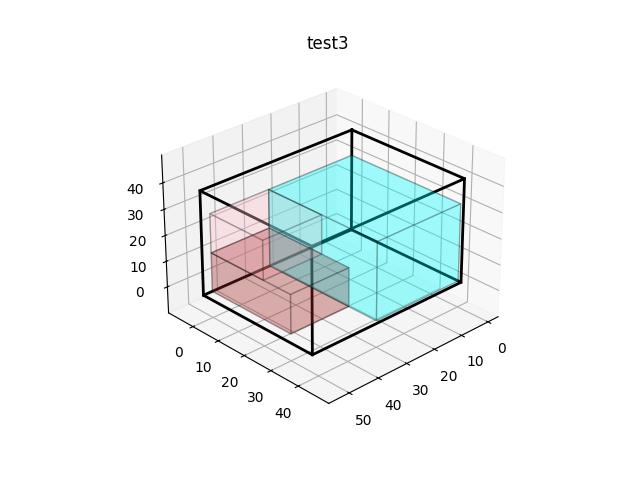
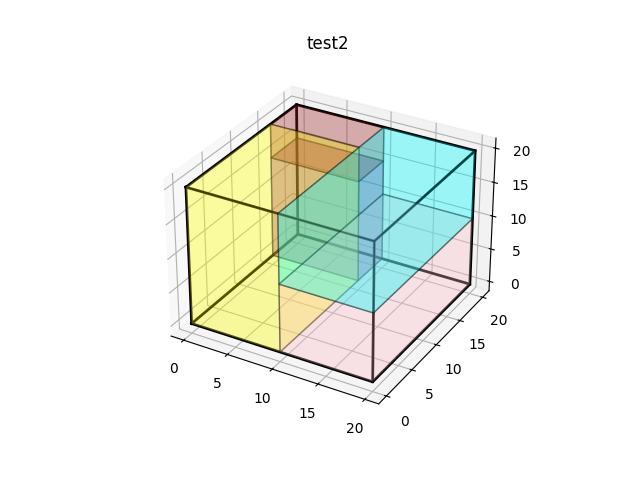
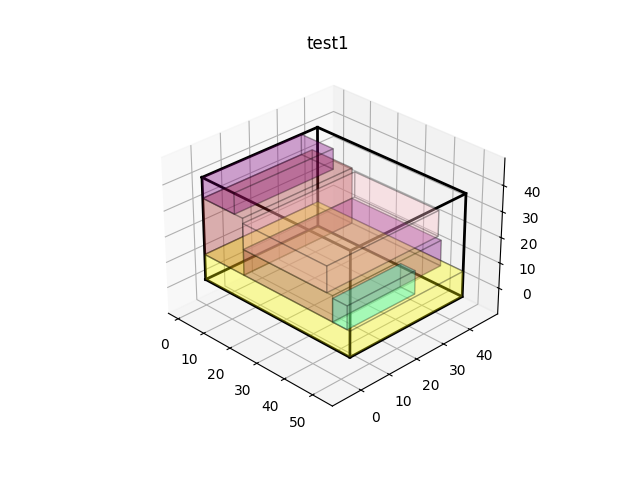


Figure 3. Packing Algorithm Test Case Results

1. **Conclusion**

Although this scanning method is able to properly retain the proportions of the object as well as create a 3D wavefront (.obj) file, it does not do so in a sufficiently quick way. The bulk of this downtime is due to the generation of the necessary transforms.json file, which uses video data in order to map each frame to a transformation matrix based on the camera’s focal length and inferred position in the environment. This downtime could potentially be minimized by generating the transforms.json file during the camera recording, rather than afterwards. This way, not only may the transforms.json dimensions be more accurate by taking the dimensions based on the phone position, but the only downtime would come during the generation of the NeRF model.

Additionally, the exact dimensions of the object were not preserved in the wavefront file. This problem could potentially be mitigated using the MobileSAM object segmenter for NeRFs. If the user can scan all of the objects at the same time, then even if the object dimensions are wrong, as long as each object is proportionally sized then they can be sorted into a container correctly. Alternatively, we can utilize Apple’s LiDAR point cloud in order to obtain a depth map as the iPhone scans the environment and obtain the correct dimensions of the objects.

Currently, the packing algorithm meets our time requirement to produce a packaging schematic in a quick manner. However, it is only limited to regular shaped objects, specifically rectangular prisms and cylinders. It is likely that the user will want to pack items of irregular shapes in which we will look into expanding the algorithm’s ability to accommodate a variety of shapes. Additionally, our testing of the packing algorithm has not considered the effect of weight distribution. To ensure a more nuanced representation of the packing process, we will also look into incorporating weight distribution analysis within the packaging algorithm.