**Memo**

To: Professor Pisano

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

Team: 14

Date: 11/14/2023

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

1. **Required Materials**

Hardware:

* Phone Camera
* Laptop (acting as the cloud server)

Software:

* Python Scripts
  + Packing Algorithm
* 3D Model Rendering (NeRF)
  + NVIDIA Instant NGP

1. **Setup**

The testing setup will be split into 2 separate modules: scanning and packing.

The scanning task will be a two step process. The first step will be to record a 360 degree view of a single box. This video must be a steady recording and encompass all sides of the object. This video will be uploaded to the laptop (server), and then processed to create a series of images as well as a transforms.json file for the images. These images and the transforms.json file will then be used as input for NVIDIA’s instant NGP executable in order to create a Neural Radiance Field (NeRF) model. Finally, the object will be manually trimmed from the environment and outputted as a 3D mesh object.

The packing task will be performed using a python script that takes manual input dimensions of objects and a bin/container. Based on the input dimensions (in cm) of the objects and container, the algorithm will sort the input objects based on a combination of the “Best Fit” and “Last Fit” heuristics. The best fit heuristic involves placing each item into the bin that minimizes the remaining empty space, while the last fit heuristic places each item into the last bin that can accommodate it. For our application, we will be specifying one bin. The Python script will produce a 3D coordinate plane representation of the packaging layout, with each object marked in a distinct color. The schematic can be rotated to view the layout along the plane.

1. **Pre-testing Setup Procedure**

Scanning:

1. Ensure functionality of smartphone Camera app
2. Open the instant NGP directory using Anaconda

Packing:

1. Specify input object and container dimensions for each test example:
   * 1. Test 1: All objects fit in container
     2. Test 2: All objects fit in container with no trim
     3. Test 3: Only a portion of objects fit in container
2. **Testing Procedure**

Scanning:

1. Record a steady 360 video of an object
2. Upload the video to the laptop (server)
3. Run colmap2nerf.py in /scripts
4. Run instant-ngp.exe with the transforms.json created by the previous script
5. Isolate the object and export it as a 3D mesh

Packing:

1. Run the python scripts: test1, test2, test3.
2. **Measurable Criteria**

The criteria for successful running and output is as follows:

1. NeRF should successfully generate a 3D model of an object. This object should then be exported as a mesh that retains its original dimensions.
2. Packing algorithm should successfully create an optimal packaging schematic. In the case where all input objects are unable to fit into the specified container, it should notify the user which items may fit.
3. Packing algorithm should produce optimal packaging arrangement <1 s.

1. **Score Sheet**

| **Scanning Algorithm** | | |
| --- | --- | --- |
| Test # | Object Shape | Correct? (Y/N) |
| 1 | Rectangular Prism |  |
| 2 | Irregular |  |
| **Result** | | % |

| **Packing Algorithm** | | | |
| --- | --- | --- | --- |
| Test # | Object Size (cm) | Container Size (cm) | Correct? (Y/N) |
| 1 | Obj 1 - 15 x 39 x 22  Obj 2 - 25 x 5 x 9  Obj 3 - 53.3 x 40.6 x 10  Obj 4 - 35 x 8 x 12  Obj 5 - 30 x 40 x 10  Obj 6 - 39 x 32 x 10 | 53.3 x 40.6 x 40.4 |  |
| 2 | Obj 1 - 10 x 15 x 20  Obj 2 - 10 x 20 x 10  Obj 3 - 10 x 5 x 5  Obj 4 - 10 x 20 x 10  Obj 5 - 10 x 5 x 15 | 20 x 20 x 20 |  |
| 3 | Obj 1 - 20.3 x 20.3 x 15.4  Obj 2 - 30 x 20.3 x 15  Obj 3 - 53.3 x 40.6 x 10  Obj 4 - 50 x 30 x 30 | 53.3 x 40.6 x 40.4 |  |
| **Result** | | | % |