

PACSSR-301007

Final Project

Tsukumo Jiang, Wency Wei, Justin Wu, Eric Yin, Austin Zhang

3/20/2025

1 Project Overview

This project creates an interactive 3D visualization of two procedurally modeled objects: a snowman and a cake. The implementation combines:

- Procedural modeling using OpenSCAD
- 3D rendering with Three.js
- Web deployment via GitHub Pages
- VR display

The final deliverable is an interactive webpage featuring:

- Turntable animation of both models
- Real-time lighting and materials
- Responsive camera controls

2 Implementation Approach

2.1 Modeling Through OpenSCAD

The Cookie uses several different functions to generate different parts of the cookie:

- **Cylinder()**: Utilize the `cylinder()` function to generate the primary thickness of the cookie, with the height representing the thickness and the radius denoted as r .
- **The upper portion of the "arch" structure**: Form a sphere($r=r$) and then stretch or compress it lengthwise with `scale([1,1,domeHeight/r])`, so that the sphere becomes a hemispherical structure of `domeHeight`. By intersecting `intersection()` and a `cube()`, only the upper part of the sphere will be retained, resulting in a smooth cookie "vault". `union()` combines the cylinder and arched parts into a complete cookie base.
- **ChocolateChips(...) module**: Iterate through the coordinates in the positions array (`pos[0]`, `pos[1]`) and determine if it is less than the radius r of the cookie (if (`hypot(...)`). $j = r$) to determine whether the chocolate bean should fall on the surface of the cookie. Simulate each chocolate bean using `sphere(r=3)`, and `translate()` places them on the corresponding coordinates and sits them on the surface of the cookie (`thickness + 0.8*domeHeight`).
- **Function $\text{hypot}(x, y) = \sqrt{x^2 + y^2}$** : This is a simple function used to calculate the Euclidean distance (that is, $\sqrt{x^2 + y^2}$). Used to determine the circumference of the chocolate bean's position relative to the center of the cookie.

The Snowman Candy uses **combination of multiple geometries**:

- **Hierarchical positioning**: Translation vectors (x, y, z) position components relative to parent coordinate system

- **Primitive combination:** Union operation merges 7 distinct geometries (2 large spheres, 1 cylinder, 4 small spheres)

The cake uses **parametric equations** for the heart-shape upper layer:

$$\begin{cases} x = 16 \sin^3(t) \\ y = 13 \cos(t) - 5 \cos(2t) - 2 \cos(3t) - \cos(4t) \end{cases}$$

2.2 Three.js Integration

The visualization employs several key animation methodologies within the Three.js framework:

- **Turntable Rotation System:** A continuous rotational animation is implemented using angular displacement about the vertical axis. The models undergo incremental y-axis rotation at a fixed angular velocity of 0.005 radians per frame, creating perpetual circular motion while maintaining visual smoothness.
- **Frame Synchronization:** The animation loop utilizes the browser's native requestAnimationFrame API, ensuring optimal frame pacing synchronized with the display refresh rate. This technique guarantees consistent 60Hz rendering performance across different hardware configurations.
- **Interactive Camera Control:** OrbitControls integration provides six degrees of freedom camera manipulation through:
 - Drag-based azimuthal and polar angle adjustment
 - Mouse wheel zoom functionality
 - Touch-optimized mobile interaction
- **Hierarchical Scene Updates:** The scene graph undergoes recursive traversal each frame, applying transformation matrices to all child nodes. This enables simultaneous animation of multiple independent meshes while preserving spatial relationships.

2.3 VR Realization

3 Results

4 Teamwork Split-up

Chengzi Jiang:

Wency Wei: Cake modeling, Powerpoint design

Zhuoyuan Wu: Shapes modeling, VR production

Eric Yin: Cookie modeling

Austin Zhang: Snowman modeling, Spin animation, Report maintainance

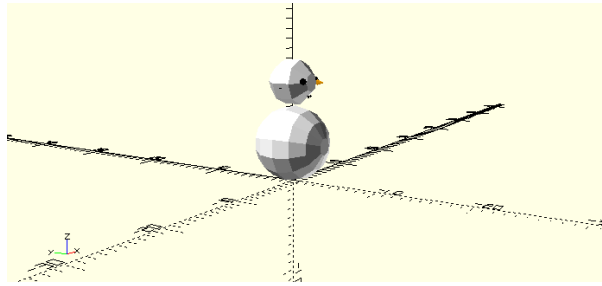


Figure 1: Snowman Candy

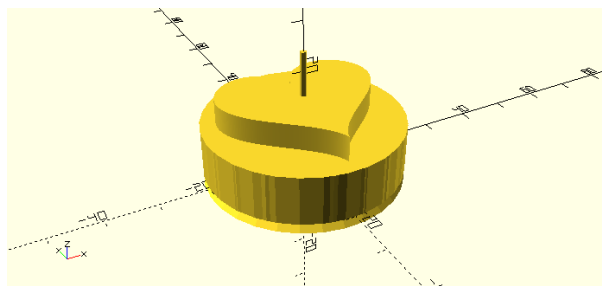


Figure 2: Cake

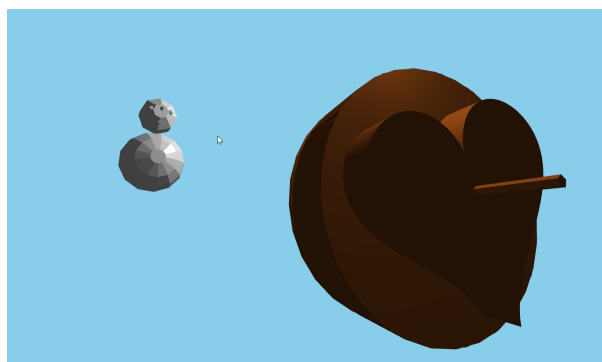


Figure 3: Animation