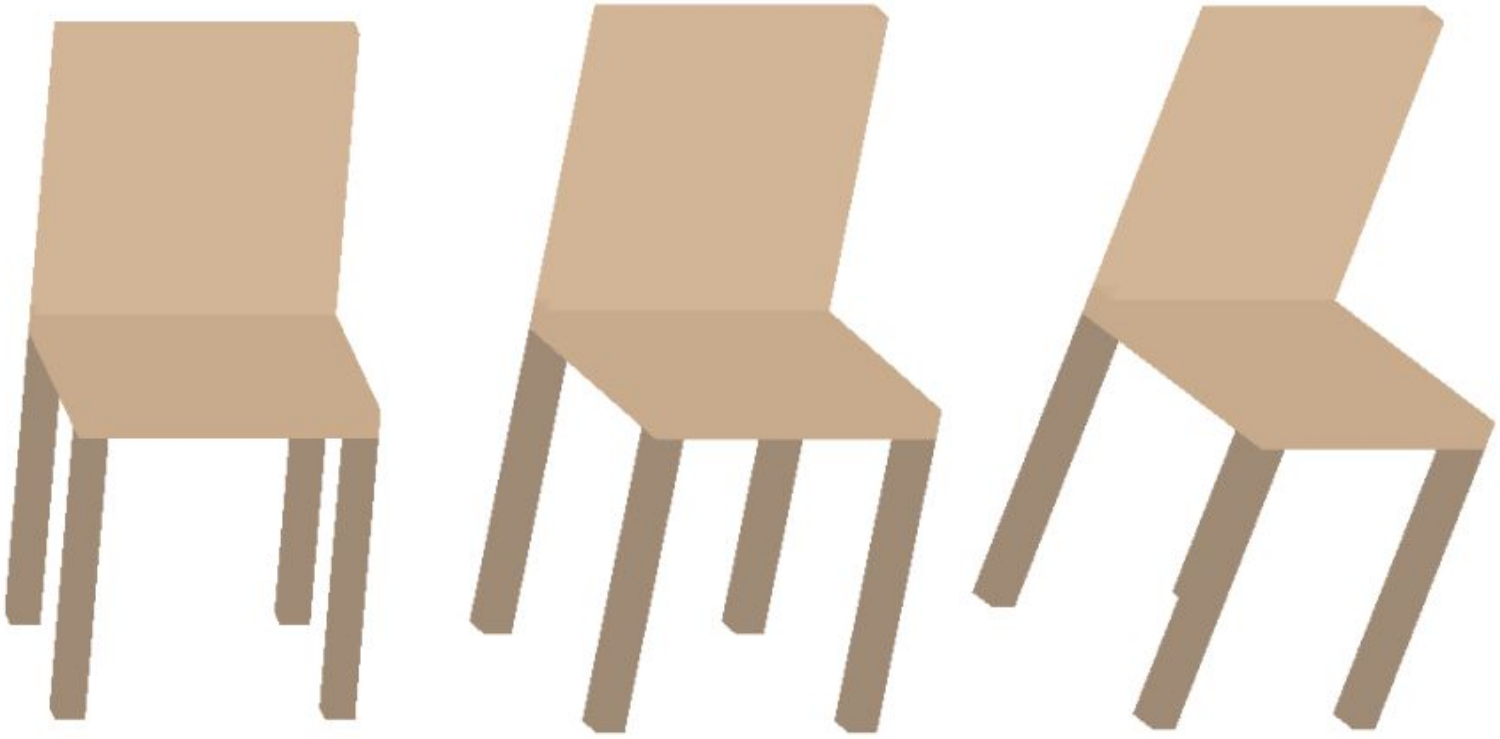
3D Graphics: Simple Rendering Engine

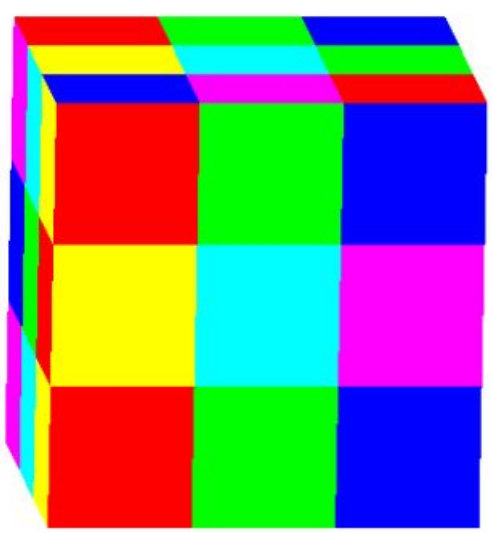
Jonathan Giha, University of Michigan Student

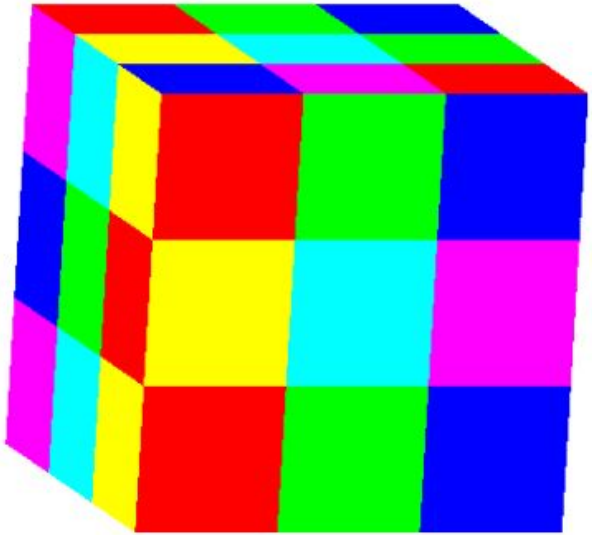
Table of Contents

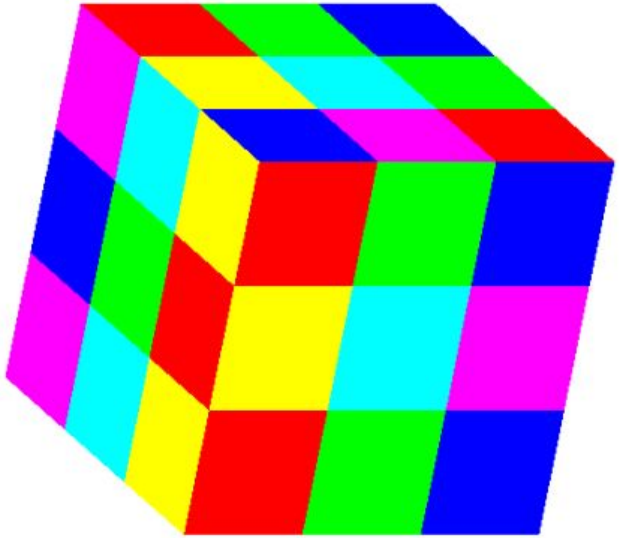
Sample Renders - Chair



Sample Renders - Rubik’s Cube







Rendering Process

- Take in camera position, angles, and a list of surfaces (defined by a list of coplanar vertices, and a color)
- Linear Algebra
  - Create transformation matrices
  - Translate vertices according to the camera position and rotate vertices according to the angles
  - Project vertices on to the 2D image plane
- Go through each pixel of the result, checking what surface, if any, should be drawn there

Limitations

- Light sources were omitted for this project
- Surfaces can only be defined with one color, no multicolored or gradients
- Making objects is very time consuming

Libraries used

- C++ Standard Libraries
  - Basic functionality such as file reading/writing.
- Eigen3
  - Making and performing operations on matrices.
  - Does NOT provide built in rendering functionality.