3D Modeling Principles and Ideas

1 Abstract

This document lists some concepts about human 3D modeling and possible ideas to improve the deep reinforcement learning strategy for 3D object modeling proposed in [...].

2 Mesh topology

2.1 About N-gons

While the N-gon meshes presented as results in the PolyGen paper seem rather elegant, the interest in using N-gons in a deep learning setting is more about the complexity of modeling variable-length sequences (conditioned on another variable length sequence, i.e., mesh vertices) rather than quality of results.

In fact, N-gon meshes are not always the best choice: rendering engines eventually triangulate them, so there's really no **result** benefit in defining an overly complicated model just to get N-gon meshes. Instead, N-gons may introduce two notable undesired side-effects:

- Using N-gons may add a computational overhead; a small one, of course, but it is required to triangulate the mesh.
- **Artifacts**: often the introduction of n-gons causes undesired modifications of the surface's topology. These are especially problematic in a context where the obtained meshes are actually used, since light would reflect weirdly on the surface and smoothing the surface would result in even more evident deformations.

3 Modeling techniques

3.1 Edge loops

In a reinforcement learning setting, edge loops definitely help defining a finite and reasonable action space. I have also observed that, in human 3D modeling, they are actually very helpful in maintaining a consistent surface topology and symmetries.

So, to improve the approach, they are definitely a good entry point. Then, how could the methodology be improved?

1. In the paper, edge loops are the only modeling technique used and also, they're only applied to the longest dimension of the cuboid. This approach renders some surface modification **impossible**: for example, **face extrusion**. An idea, of which I still have to verify the feasibility, would be to apply edge loops to all dimensions and render all the **intersection vertices** editable.

2. **Incremental edge loops**: if the Mesh-Agent can't edit a loop to get an improvement, then it might add a loop. **Where?**

3. **Primitive substitution**: if the Prim-Agent can't edit or remove cuboids to get an improvement, it might try to substitute cuboids with another primitive. **Which? How many other primitives are there?**