Low-poly hard surface 3D models (like models of furniture or machinery) do not look realistic, often because of their extremely sharp edges. There are many ways to soften the edges of a model, including subsurface modifiers and normal maps, but one good way that does not affect performance or memory efficiency considerably is edge chamfering. **Edge chamfering** is usually done in modelling software with either special tools or multiple steps. To make this method more accessible for game developers not familiar with modelling software, an algorithm and a **Unity asset** were created that automatically chamfer the edges of a mesh.

### Algorithm

Given a mesh (A), vertices on hard edges are moved (B) and new faces are created between them (C) which are already smooth-shaded since no vertex normals are changed (D).

### Results

**Before** | **After**
---|---
![Image before](image1.png) | ![Image after](image2.png)

**Vertex & Triangle Counts**

<table>
<thead>
<tr>
<th>Model</th>
<th>Vertices Before</th>
<th>Vertices After</th>
<th>Triangles Before</th>
<th>Triangles After</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model 1</td>
<td>1200</td>
<td>1200</td>
<td>1800</td>
<td>1800</td>
</tr>
<tr>
<td>Model 2</td>
<td>1400</td>
<td>1400</td>
<td>2100</td>
<td>2100</td>
</tr>
<tr>
<td>Model 3</td>
<td>1600</td>
<td>1600</td>
<td>2400</td>
<td>2400</td>
</tr>
<tr>
<td>Model 4</td>
<td>1800</td>
<td>1800</td>
<td>2700</td>
<td>2700</td>
</tr>
<tr>
<td>Model 5</td>
<td>2000</td>
<td>2000</td>
<td>3000</td>
<td>3000</td>
</tr>
</tbody>
</table>

**With UVs**

![Graph with UVs](graph1.png)

**Without UVs**

![Graph without UVs](graph2.png)

[GitHub Repository](https://github.com/dianx93/EdgeChamferingDemo)

---

Diana Algma, MSc student of Computer Science  
Institute of Computer Science, University of Tartu  
Supervisor: Jaanus Jaggo