Remeshing

The goal of surface remeshing is to improve the quality of a mesh and to reduce its complexity. Here, mesh quality refers to non-topological properties such as vertex sampling and face size, alignment and regularity. Remeshing algorithms often compute point locations on or near the original surface and iteratively relocate vertices to improve mesh quality. In [1], Botsch et al. used such an approach to generate isotropic triangle meshes in an efficient and robust manner.

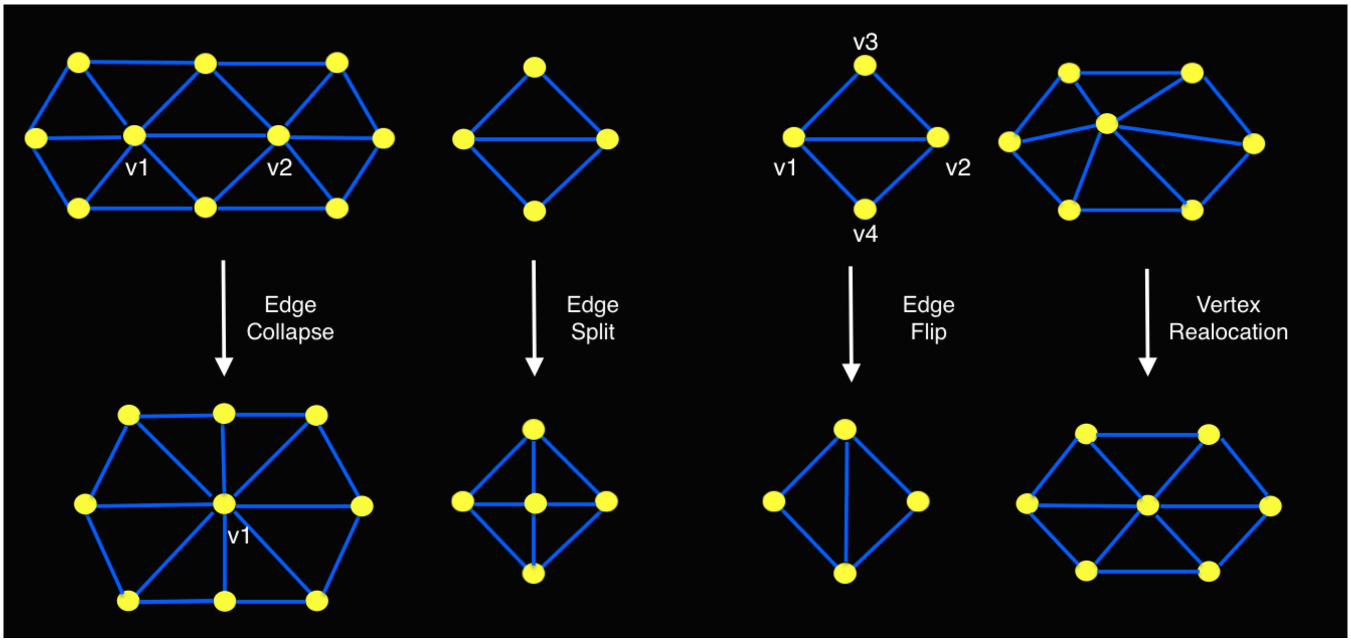


Figure 1 Figure 2 Figure 3 Figure 4

Given a target edge length, the algorithm works as follows:

1. Set **low = 4/5 x edge length** and **high = 4/3 x edge length**
2. For i = 1… n
   a. **Split long edges:** Split an edge at its midpoint if its length is greater than **high** (Figure 1).

   b. **Collapse short edges:** Remove an edge if its length is less than **low** (Figure 2). To prevent creating new edges that are longer than **high**, check if a new edge connecting the one ring neighbors of v1 (the vertex to be removed) and v2 has a length greater than **high**. No collapse takes place in such a scenario.

   c. **Equalize valences:** Flip an edge to equalize vertex valences (Figure 3). Deviation from the target valence (6 for interior vertices and 4 for boundary vertices) is computed as follows:

      dev = |val(v1)-target(v1)| + |val(v2)-target(v2)| + |val(v3)-target(v3)| + |val(v4)-target(v4)|

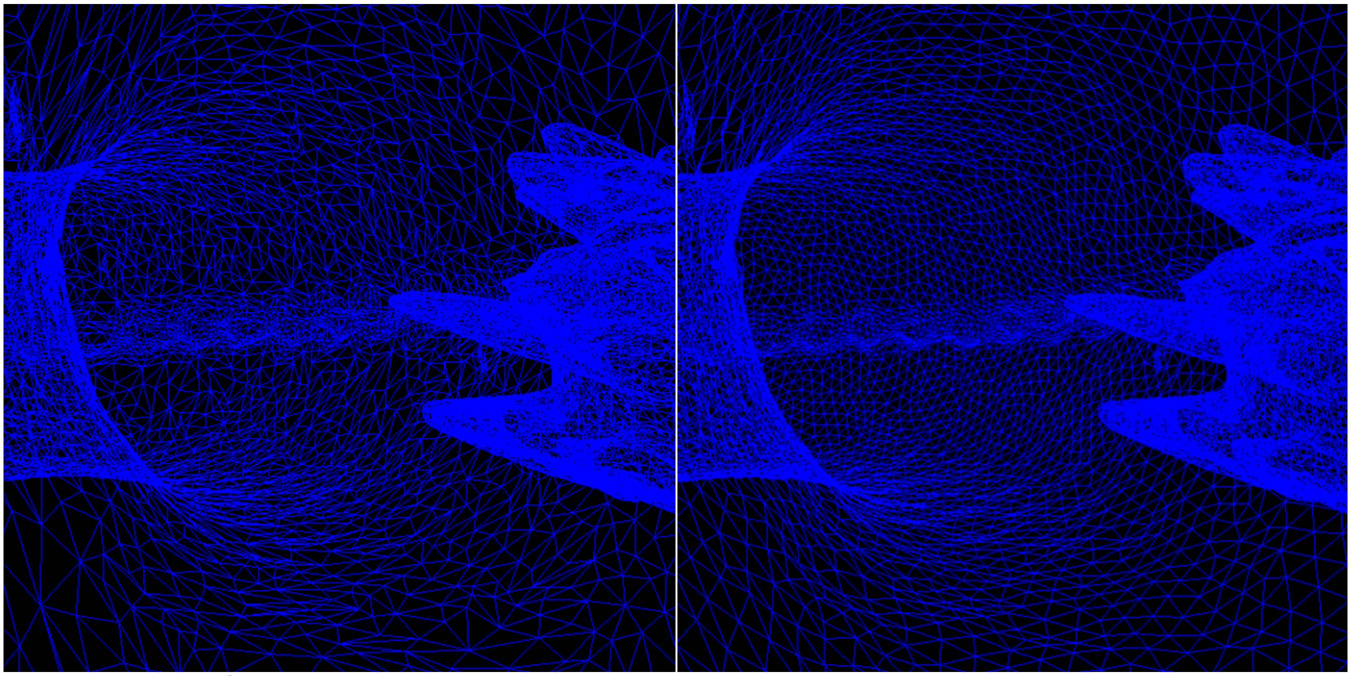
      If the deviation before the flip is less than the deviation after the flip, then the edge is flipped back.

   d. **Tangential relaxation:** Apply an iterative smoothing filter to the mesh by constraining vertex movement to its tangent plane (Figure 4). If p is an arbitrary vertex position, n the vertex normal and q the barycenter of the one ring neighbors of p, then the new position p_{new} is computed by projecting q onto p's tangent plane:

      $$\mathbf{p_{new} = q + n^T n\ (p - q)}$$

   e. **Project to surface:** Map the vertices back to the original mesh.

The algorithm as stated produces a mesh with uniform triangle size. The algorithm can be extended to achieve adaptive remeshing that produces finer face elements in regions with high curvature.



Original Mesh Mesh after Isotropic Remeshing

Implementation: https://github.com/rohan-sawhney/remesh

[1] Botsch et al. A Remeshing Approach to Multiresolution Modeling