UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

> Reinventa el mundo <

Computer Graphics

Class 28. Geometry Processing

Professor: Eric Biagioli

TRANSFORMATEC

Today

- General Ideas about Texture mapping.
- Mesh Parametrization.
- Topological operators and simplification
- G20 awards.
- Activity (to upload to canvas)

References for the class of today: (part of the second partial exam)

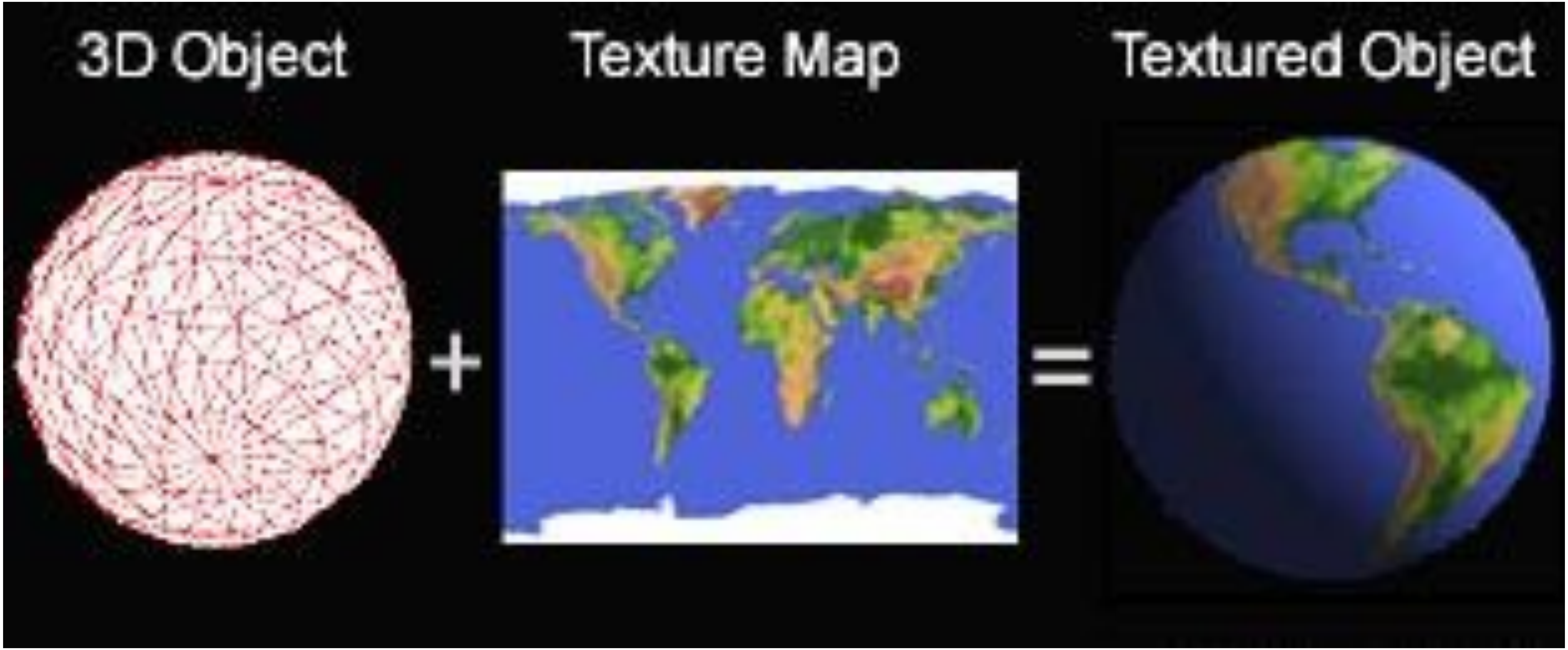
- Botsch, M., Kobbelt, L., Pauly, M., Alliez, P., and Lévy, B. *Polygon Mesh Processing*. A K Peters, 2010. → **CHAPTERS 5 (Parametrization), 6 (Remeshing) and 7 (Simplification & Approximation).**

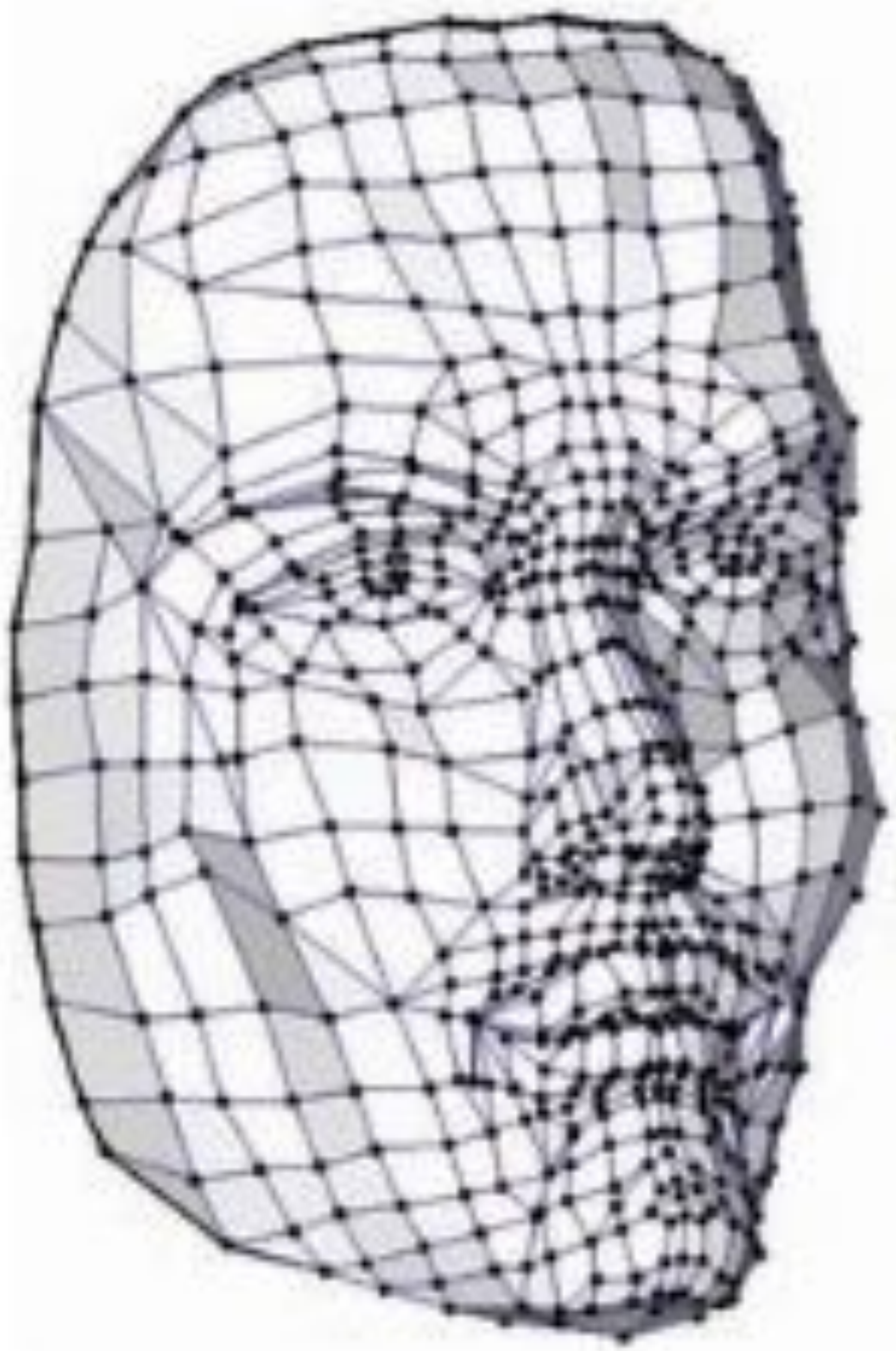
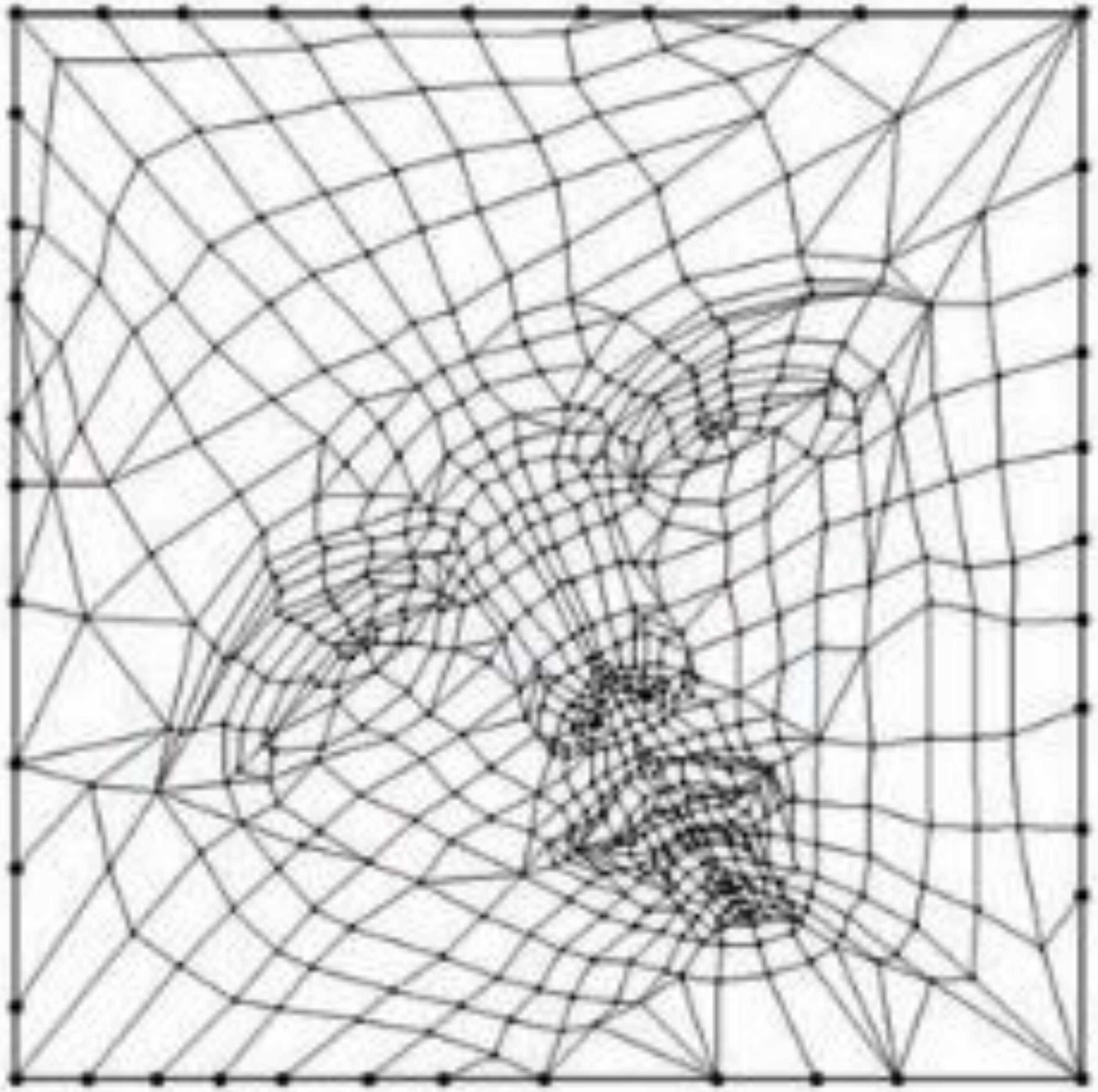
- Hughes, J. F., van Dam, A., McGuire, M., Sklar, D. F., Foley, J. D., Feiner, S., and Akeley, K. *Computer Graphics: Principles and Practice, 3 ed*. Addison-Wesley, Upper Saddle River, NJ, 2013. → **CHAPTERS 25 (Meshes), 9 (Functions on Meshes)**

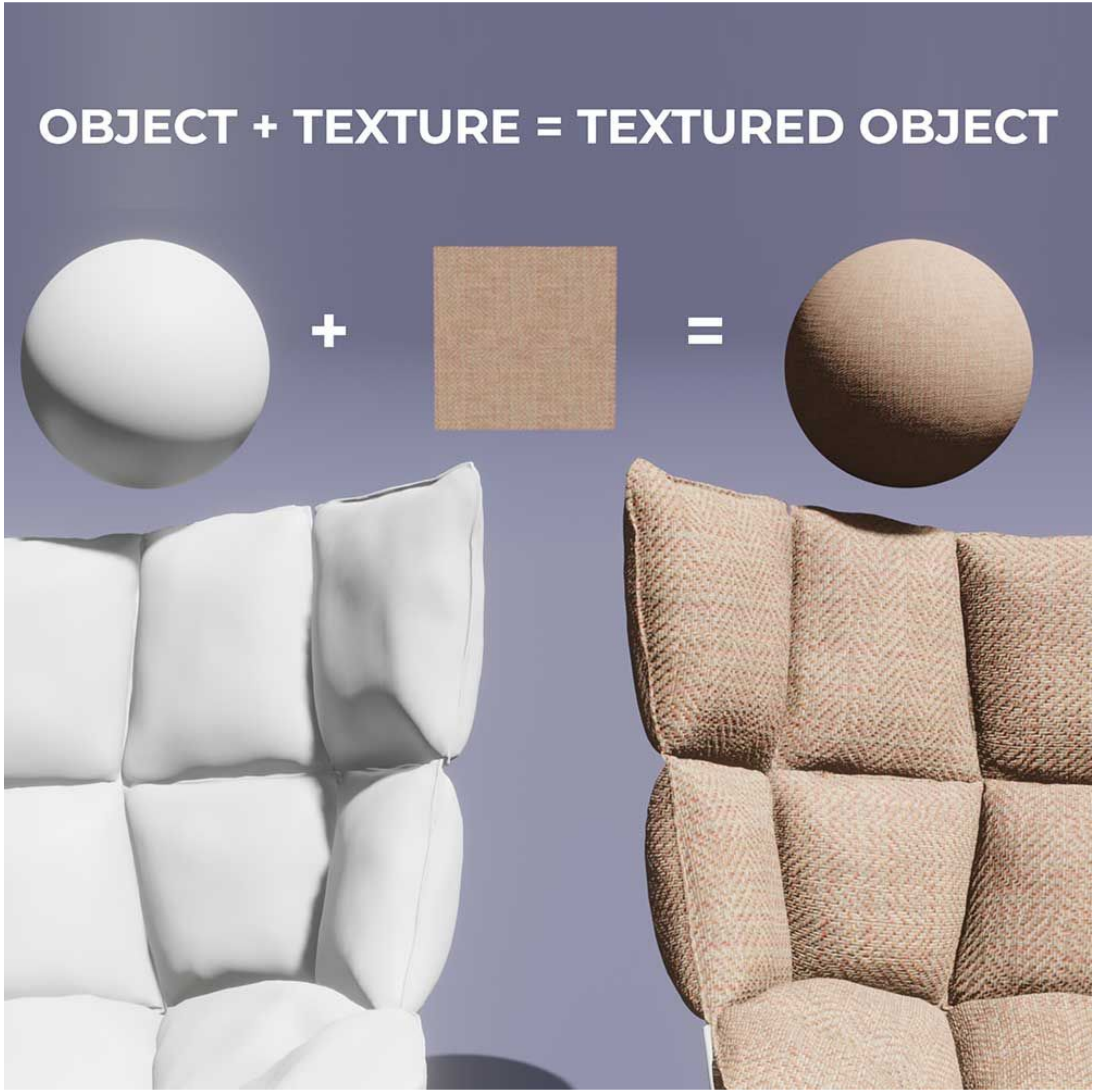
- https://en.wikipedia.org/wiki/Texture_mapping

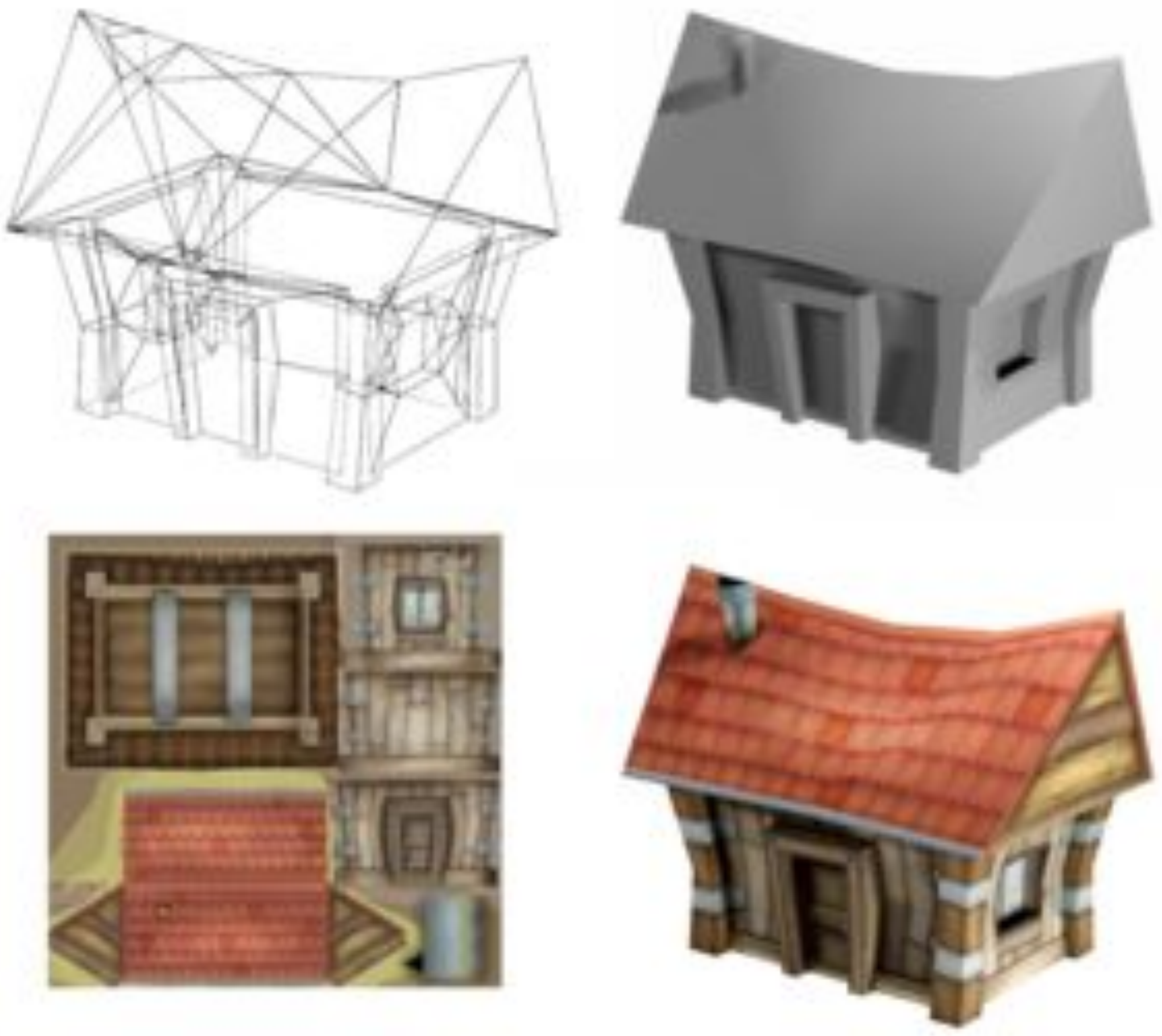
- Vieira, Antonio & Velho, Luiz & Lopes, Hélio & Tavares, Geovan & Lewiner, Thomas. (2003). Fast Stellar Mesh Simplification. Brazilian Symposium of Computer Graphic and Image Processing. 2003. 27 - 34. 10.1109/SIBGRA.2003.1240988.

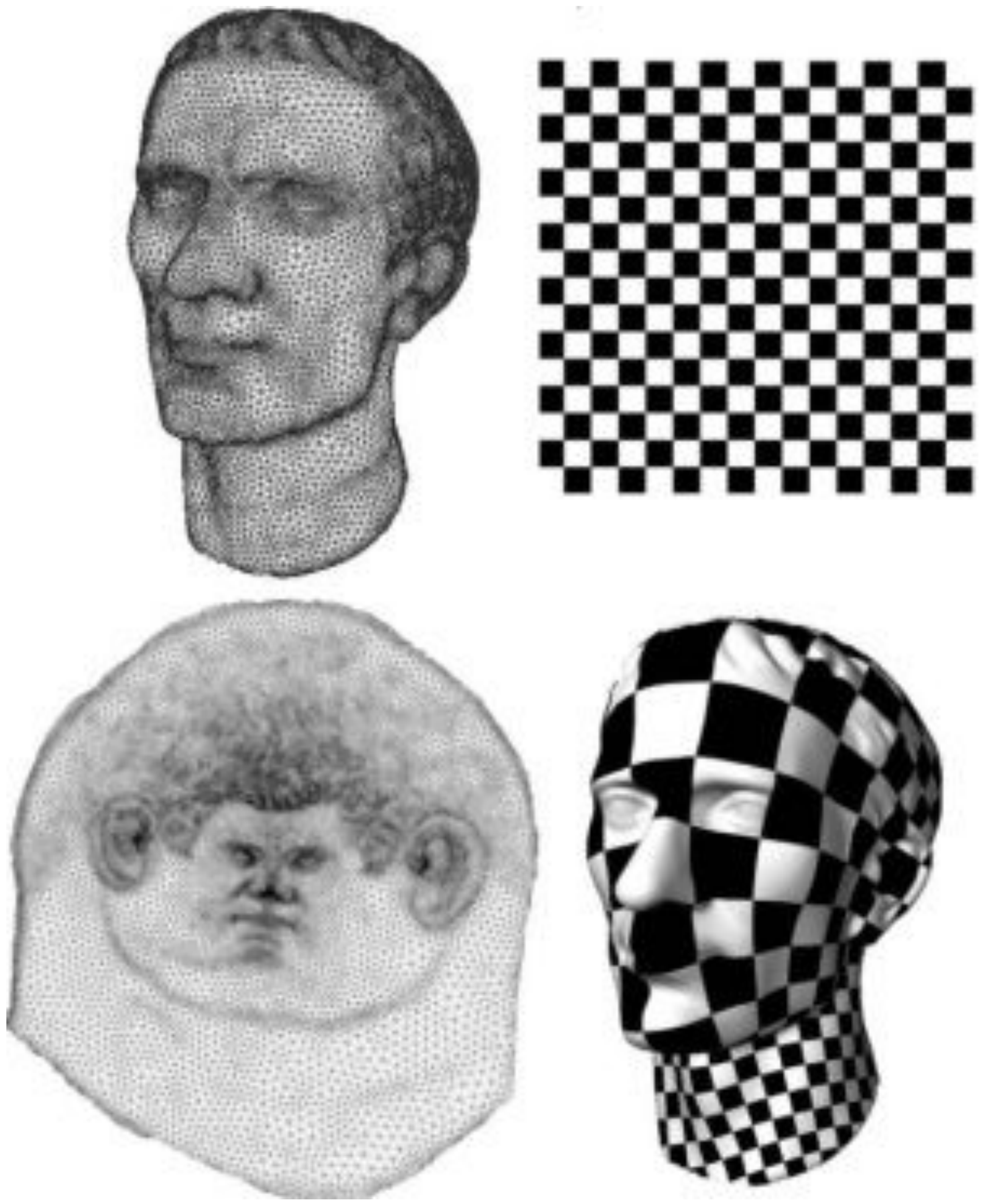
Texture mapping


3D Object
+
Texture Map
=
Textured Object

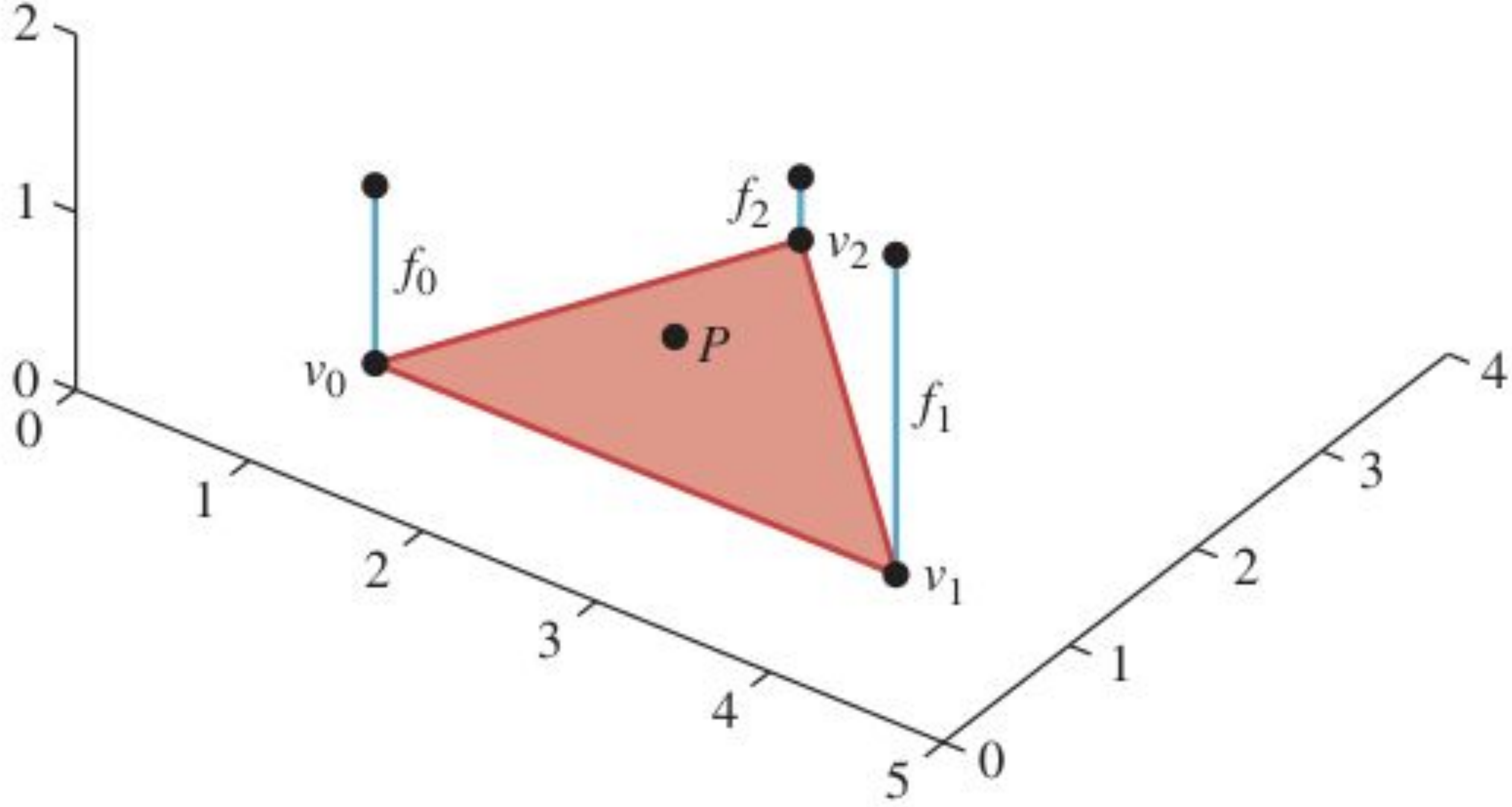




OBJECT + TEXTURE = TEXTURED OBJECT
+
=





Computing the texture coordinates



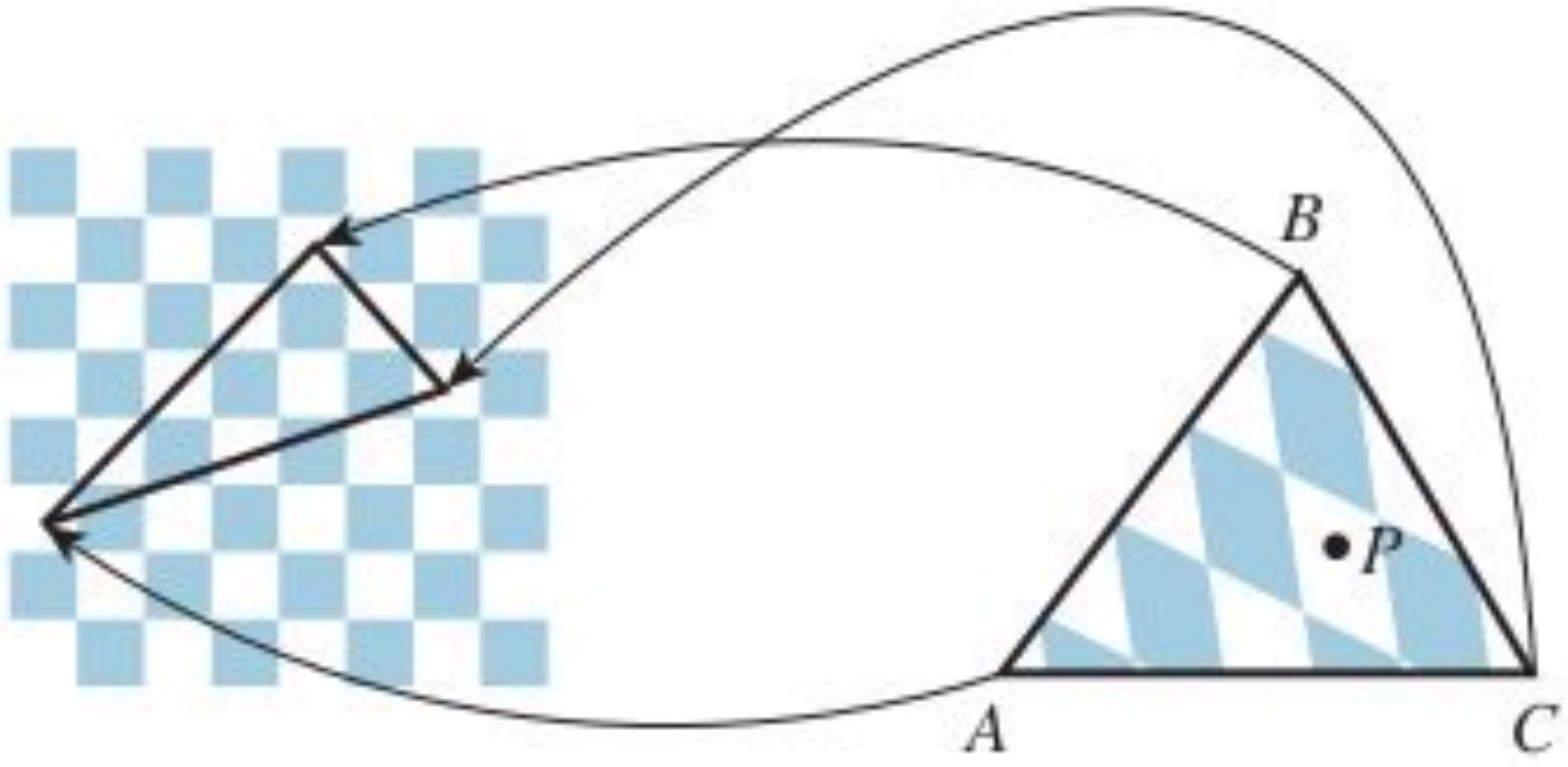


Figure 9.12: The point P of the triangle $T = \triangle ABC$ has its color determined by a texture map. The points A, B, and C have been assigned to points in the checkerboard image, as shown by the arrows; the point P corresponds to a point in a white square, so its texture color is white.

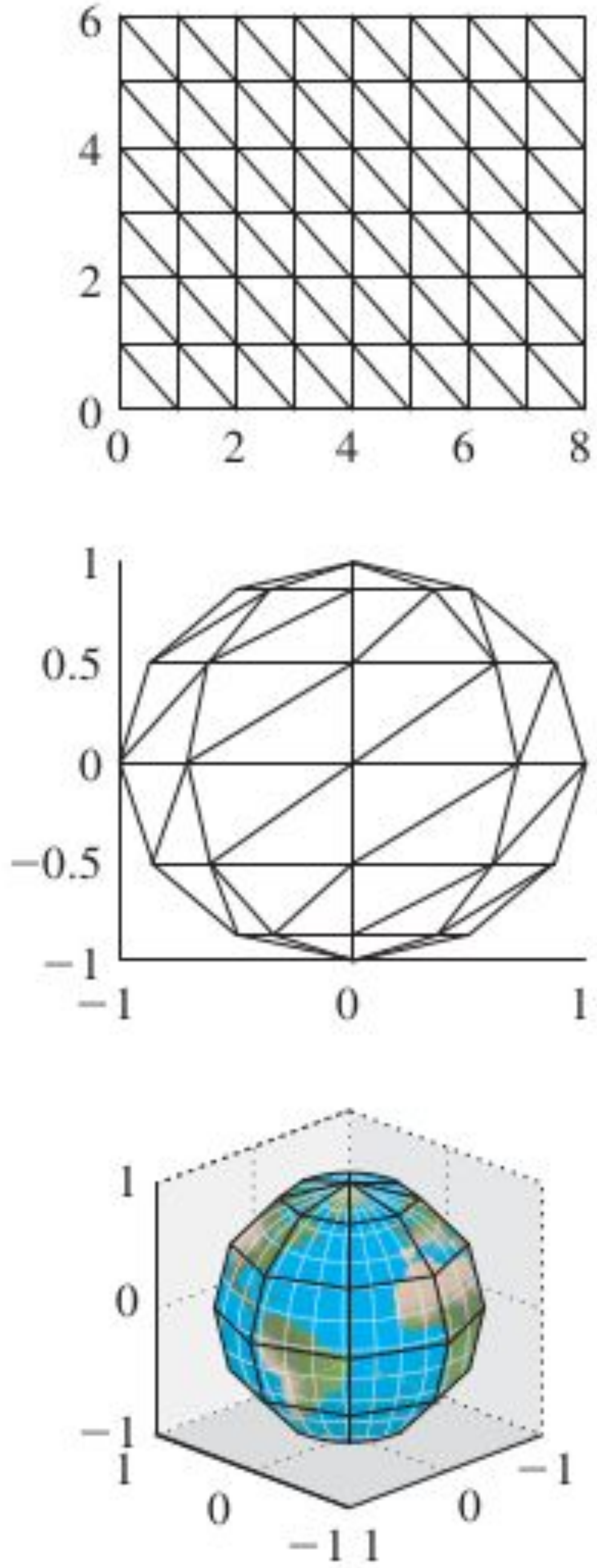


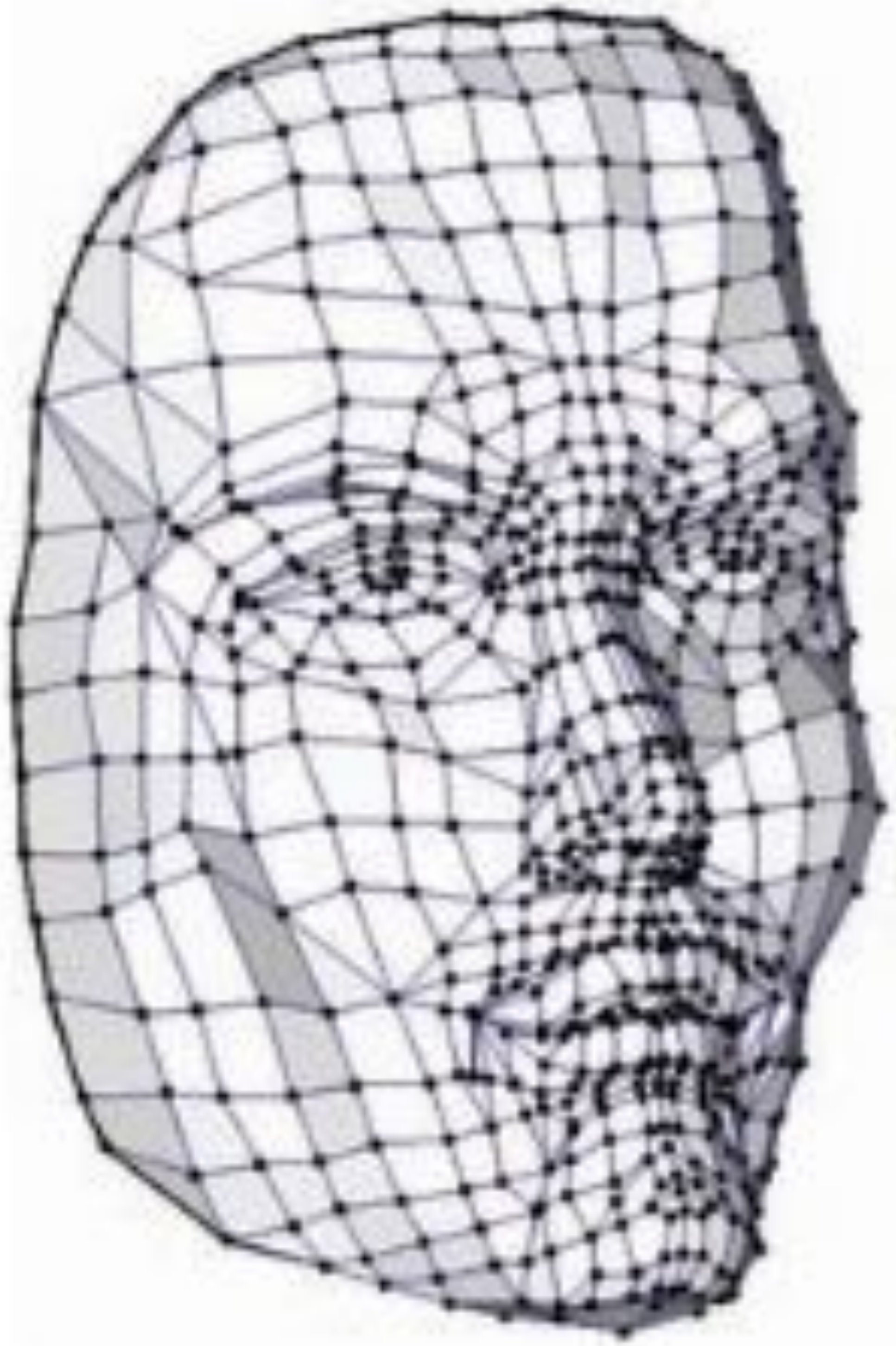
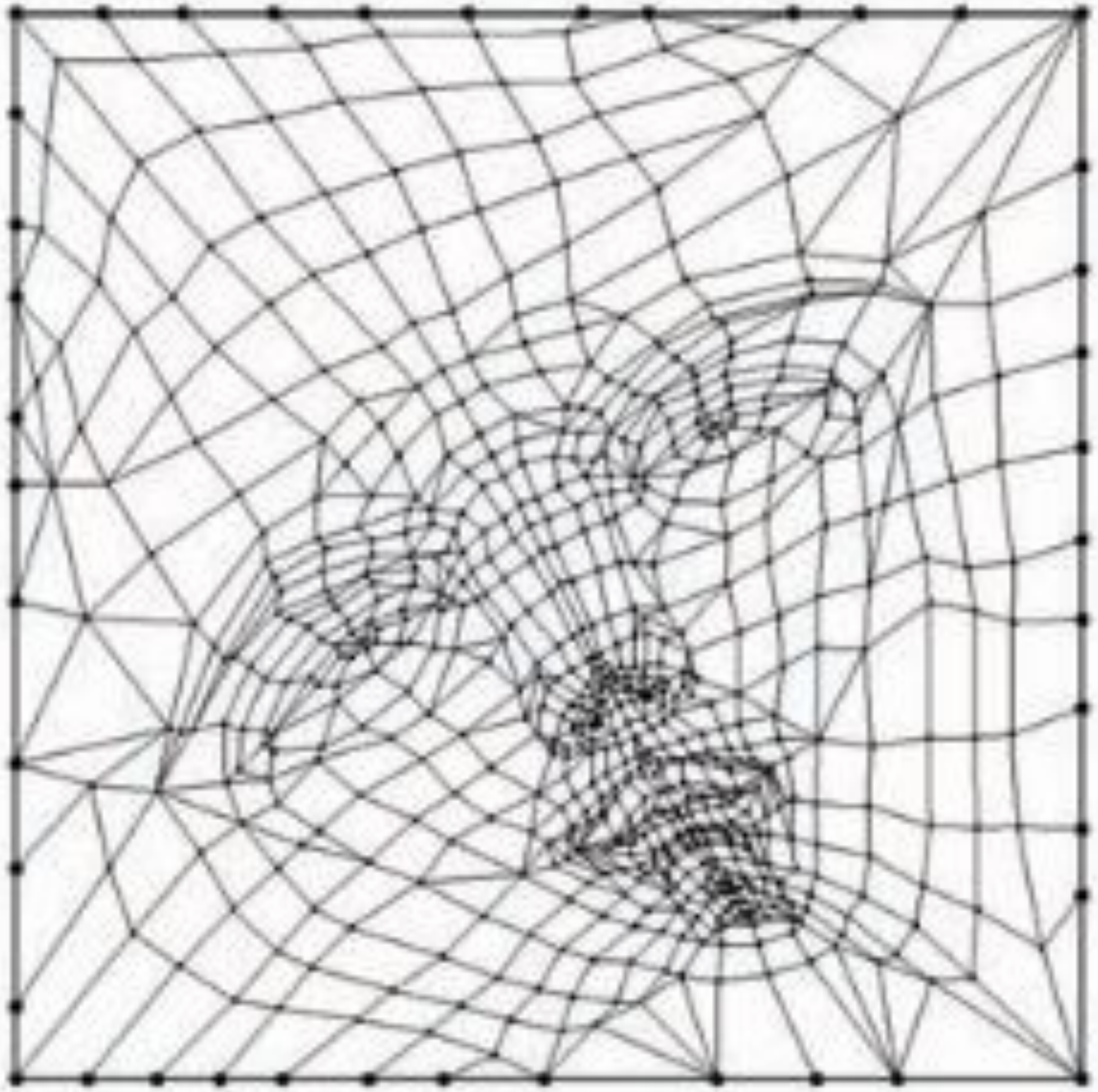
Figure 9.13: Texture-mapping a globe.

Parametrization

- We saw that we can use a texture to color the surface of a 3d model.
- How do we assign 2d coordinates to the vertices of a 3d model?
  - LSCM (Least Squares Conformal Maps)
  - ABF (Angle Based Flattening)

Challenges:

- where should we cut the mesh in order to represent it in a planar way?
- distortions!

Topological operators and simplification

Triangulated surfaces

1. Each face may occur no more than once, that is, no two faces of the mesh can share more than two vertices.
2. The degree of each vertex is at least three.
3. If V is a vertex, then the vertices that share an edge with V can be ordered $U_1, U_2, \ldots, U_n$ so that $\{V, U_1, U_2\}$, $\{V, U_2, U_3\}, \ldots, \{V, U_{n-1}, U_n\}$ are all triangles of the mesh, and
   (a) $\{V, U_n, U_1\}$ is a mesh triangle (in which case V is said to be an **interior vertex**), or
   (b) $\{V, U_n, U_1\}$ is not a mesh triangle (in which case V is said to be a **boundary vertex**),

   and there are no other triangles containing the vertex V (see Figure 25.4).

In the event that such a mesh has no boundary vertices, it's called a **closed surface;** if it has boundary vertices, it's called a **surface with boundary.**

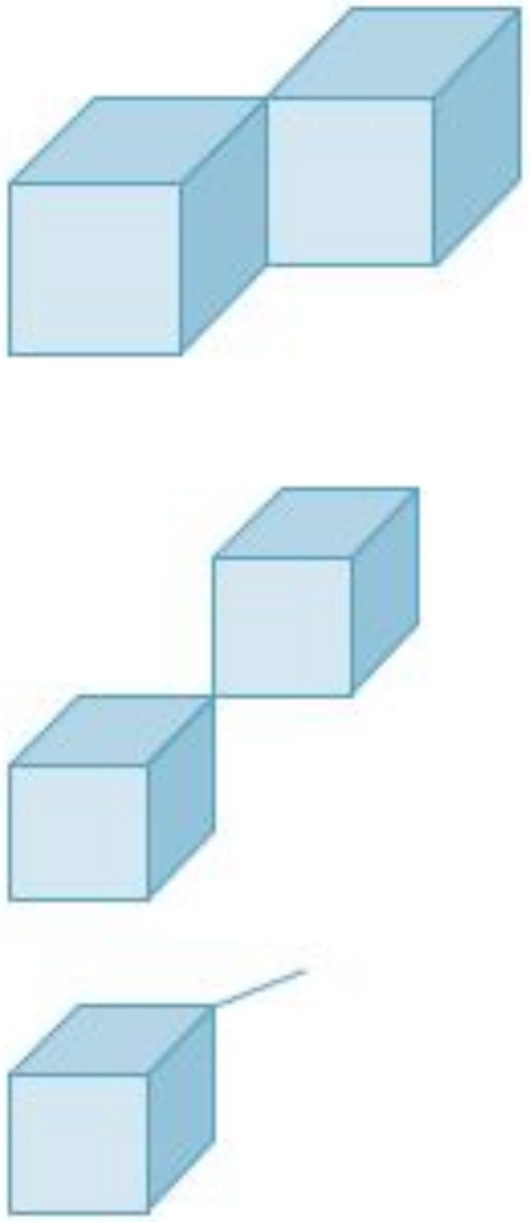


Figure 25.5: Each of these fails to be a surface mesh in some way.

Terminology

- star of a vertex V→ the vertex V and all the edges and faces containing it.
- link of vertex V → boundary of the star of V
- 1-ring of V: vertices at distance "1 edge" from V
- 2-ring of V: vertices at distance "2 edges" from V

Figure 25.10: The star of the top vertex is drawn in brown; the 1-ring, which forms an octagon in the middle level, in red. The 2-ring, at the bottom drawn in bright green, is connected into a figure-eight shape.

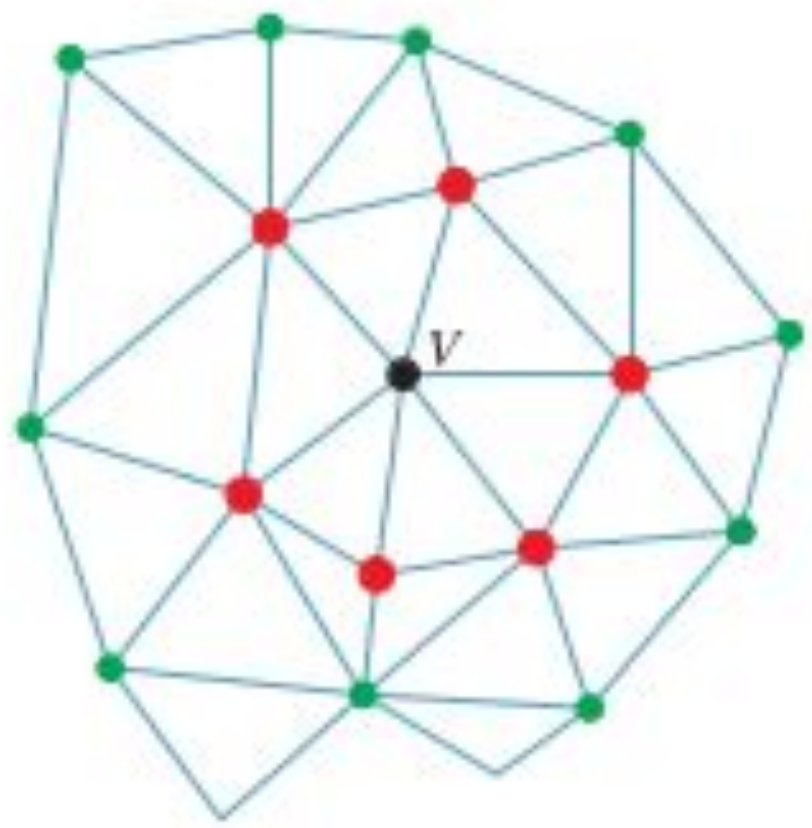


Figure 25.9: The 1-ring of V is drawn in large red dots; the 2-ring in smaller green dots.

Embedding

With this terminology, we'll define an **embedding** of a surface mesh as an assignment of distinct locations to the vertices, extended to edges and faces by linear interpolation and satisfying one property: The triangles $T_1 = C(P_i, P_j, P_k)$ and $T_2 = C(P_p, P_q, P_r)$ intersect in $\mathbf{R}^3$ only if the vertex sets $\{i, j, k\}$ and $\{p, q, r\}$ intersect in the abstract mesh. If the intersection is a single vertex index s, then $T_1 \cap T_2$ must be P_s; if the intersection has two vertices s and t, then $T_1 \cap T_2 = C(P_s, P_t)$; and if the intersection is all three vertex indices, then T_1 must be identical to T_2. (Note that we're assuming that i, j, and k are distinct and p, q, and r are distinct; otherwise, either T_1 or T_2 would not be a triangle.)

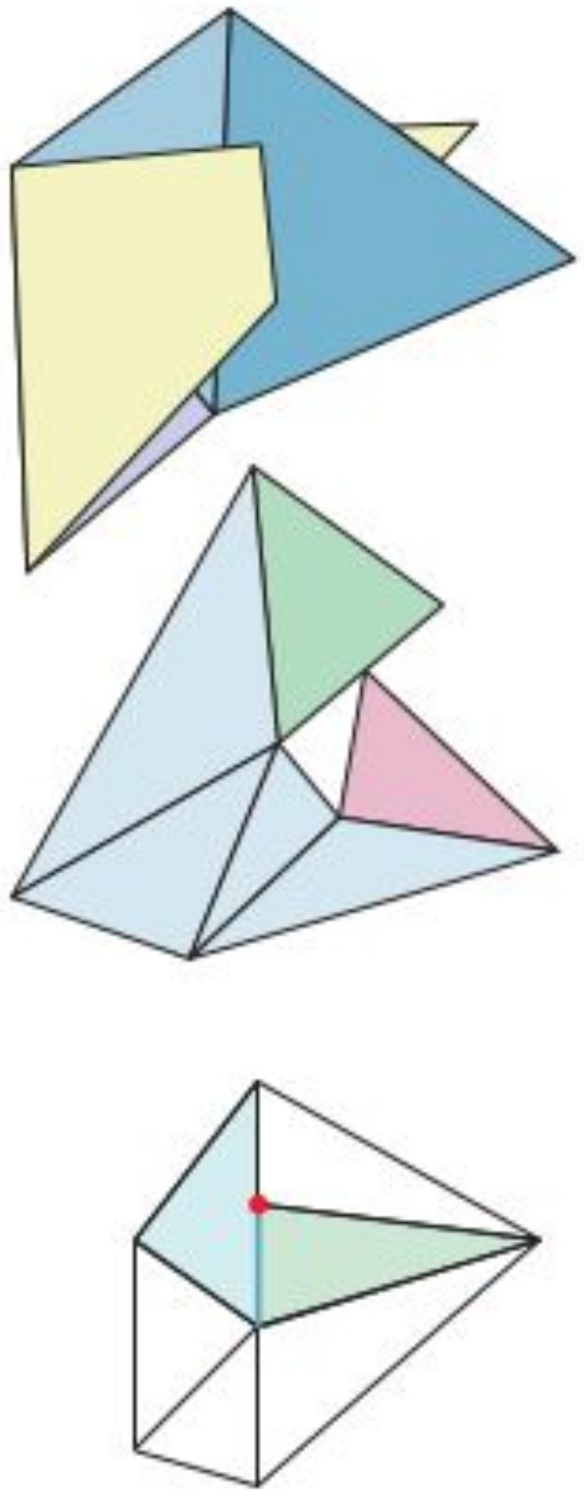


Figure 25.11: (Top) A mesh with bad self-intersections. (Middle) A mesh in which a vertex of the pink face at the right lies in the middle of an edge of the green face at the top right. (Bottom) The red vertex marked with a dot is a T-junction.

Topological operators in meshes

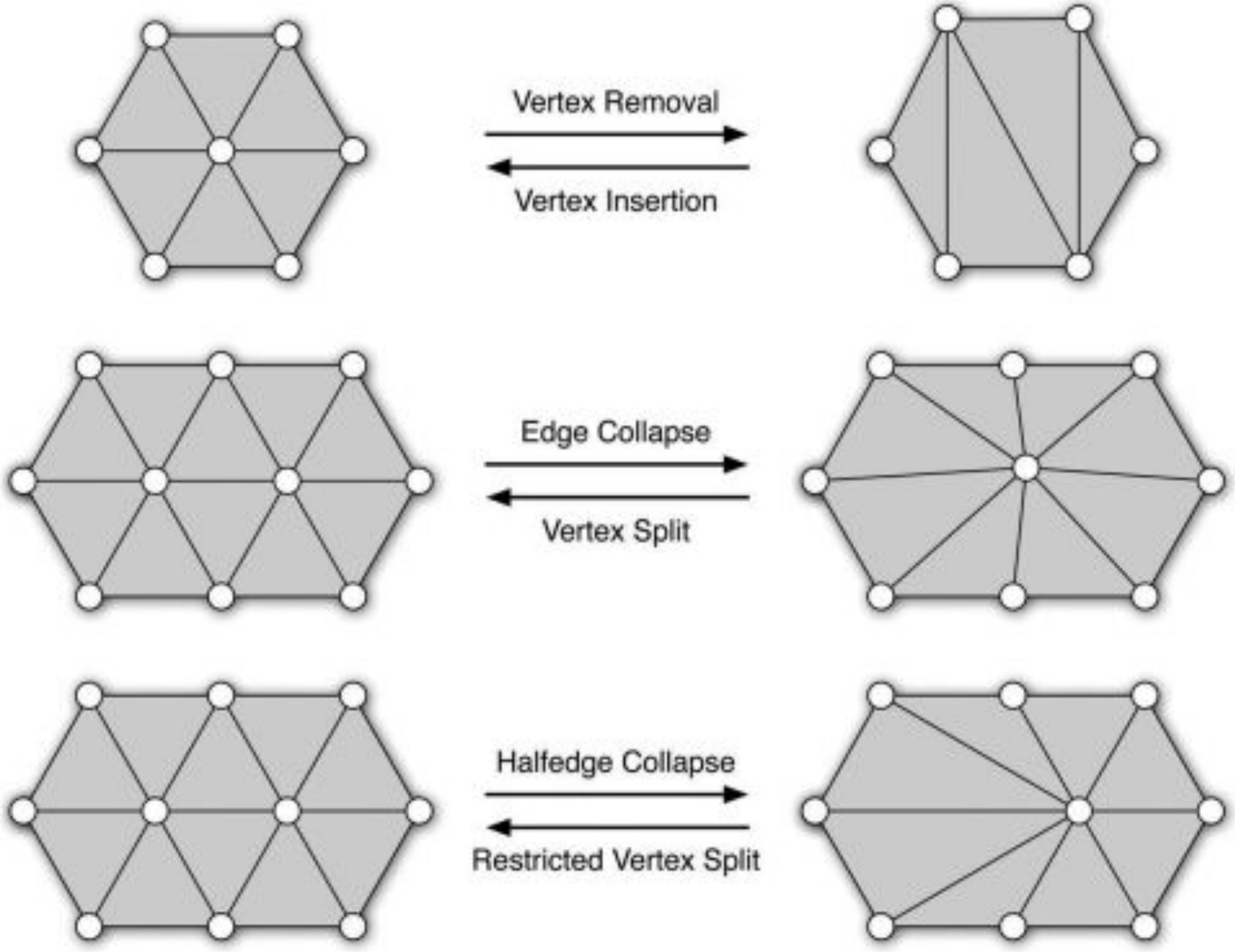


Figure 7.3. Euler operations and inverses for incremental mesh decimation: vertex removal (top), general edge collapse (middle), and halfedge collapse (bottom).

Topological operators in meshes

- If both $\mathbf{p}$ and $\mathbf{q}$ are boundary vertices, then the edge $(\mathbf{p}, \mathbf{q})$ has to be a boundary edge.
- For all vertices $\mathbf{r}$ incident to both $\mathbf{p}$ and $\mathbf{q}$ there has to be a triangle $(\mathbf{p}, \mathbf{q}, \mathbf{r})$. In other words, the intersection of the one-rings of $\mathbf{p}$ and $\mathbf{q}$ consists of vertices opposite the edge $(\mathbf{p}, \mathbf{q})$ only.

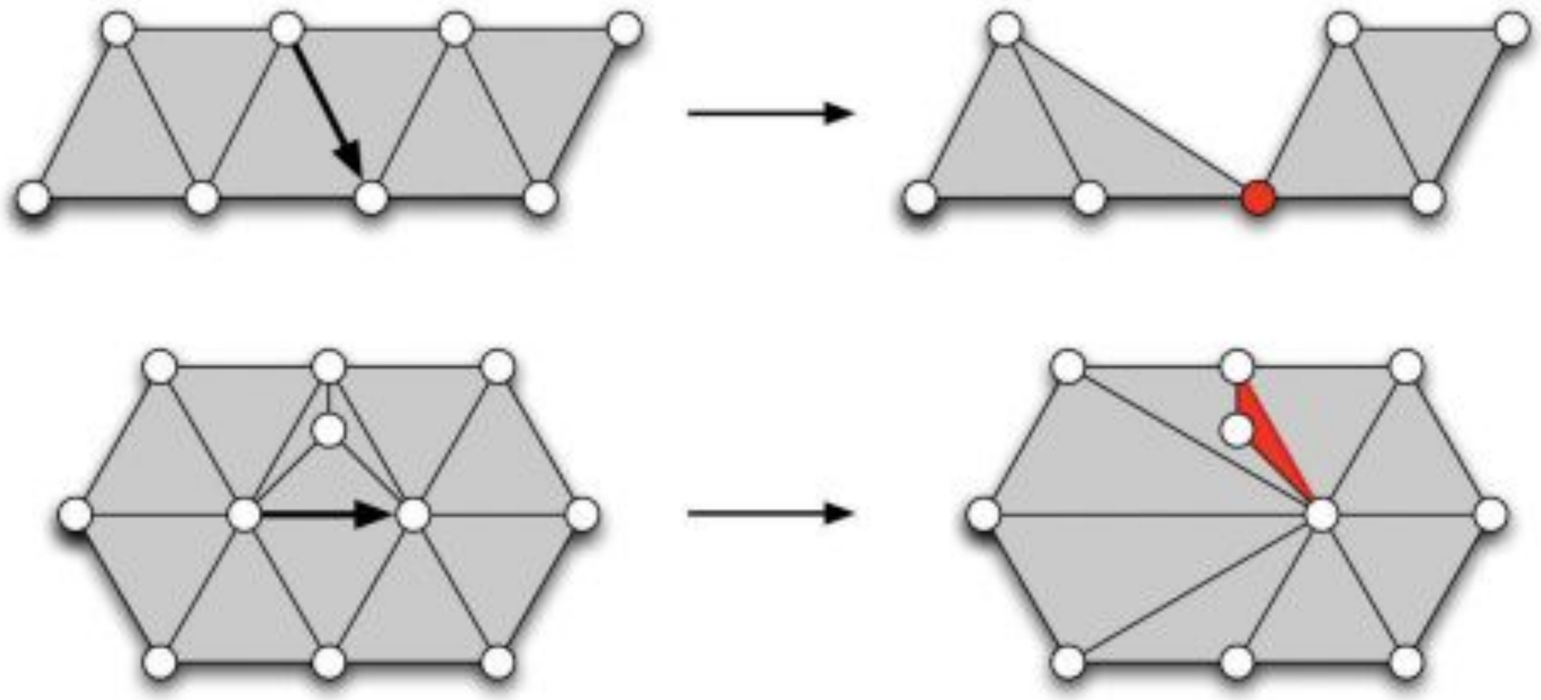


Figure 7.4. Two examples for topologically illegal (half-)edge collapses $\mathbf{p} \to \mathbf{q}$. Collapsing two boundary vertices through the interior leads to a non-manifold pinched vertex (top). The one-rings of $\mathbf{p}$ and $\mathbf{q}$ intersect in more than two vertices, which after collapsing results in a duplicate fold-over triangle and a non-manifold edge (bottom).

Simplification: what is?

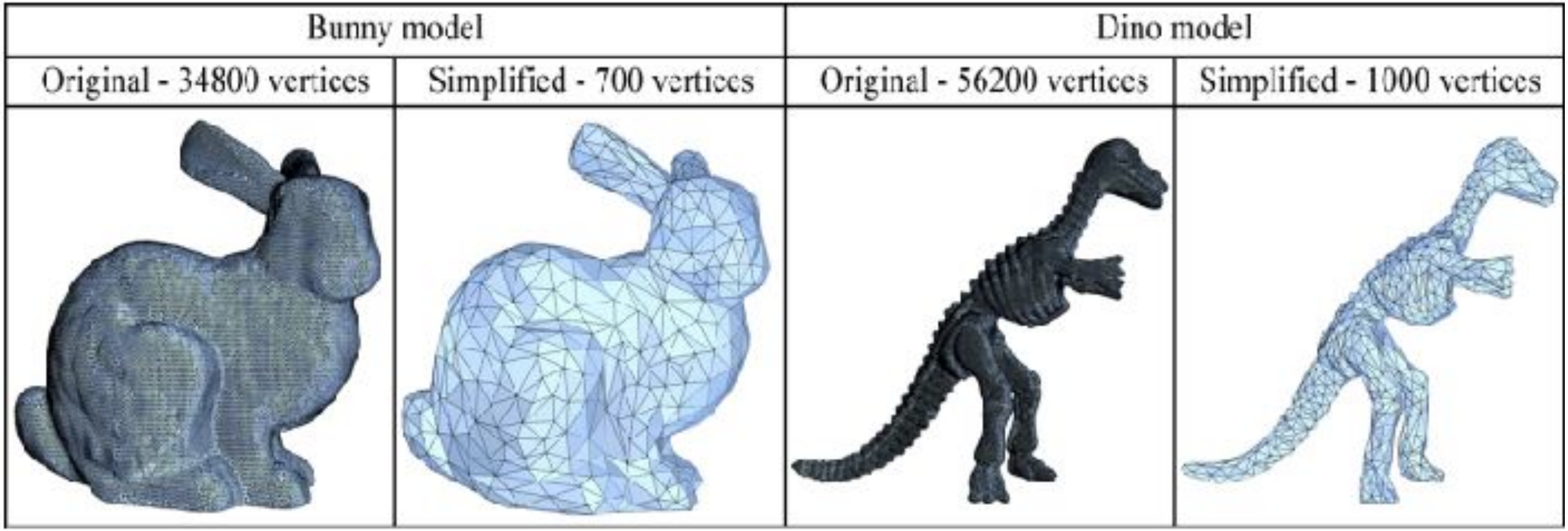


Fig. 2. Two examples of mesh simplification.

Fast stellar mesh simplification

Operations:

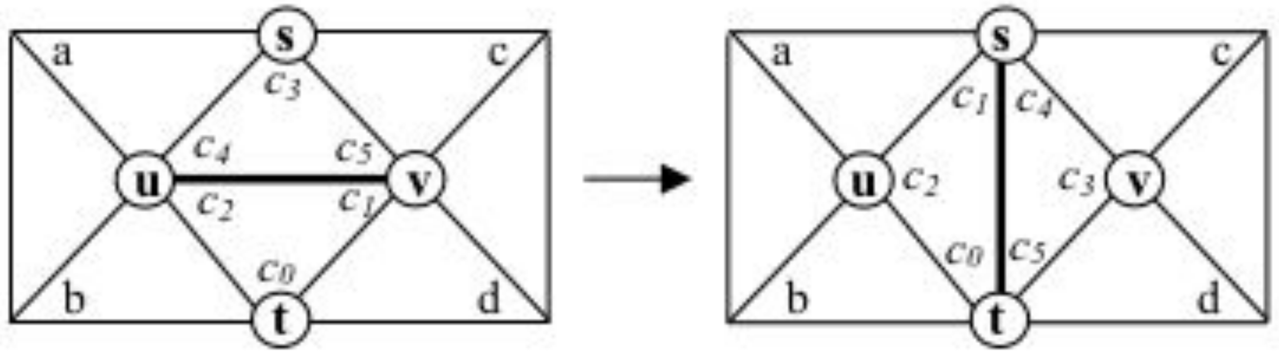


Figure 3. Edge–Flip.

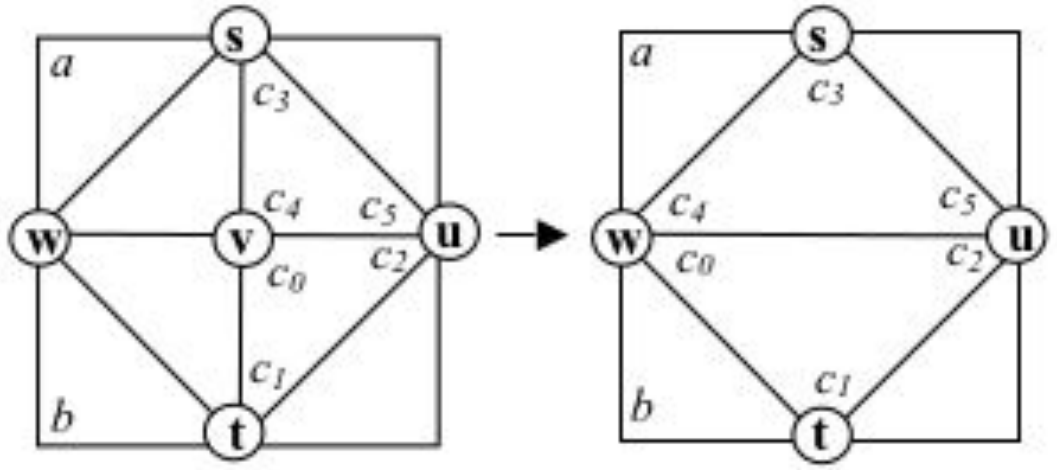


Figure 4. Edge–Weld.

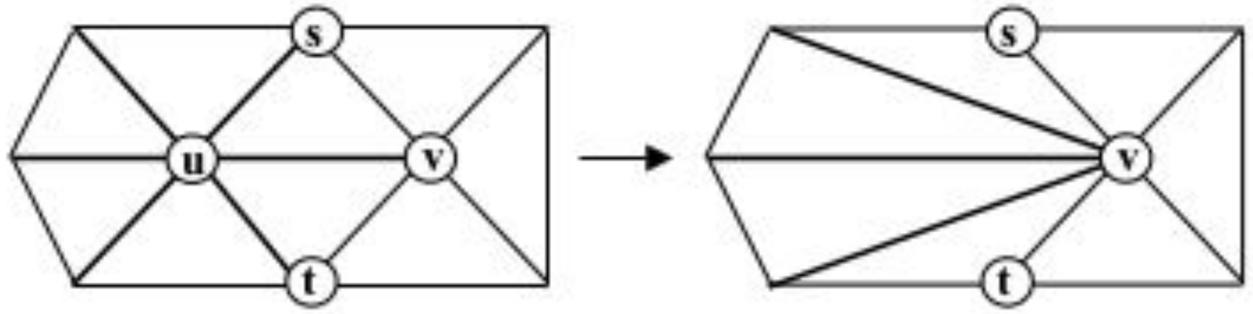


Figure 2. Edge–Collapse.

Fast stellar mesh simplification

Operations:

Algorithm 1: Edge–Flip(c_0)

// Label incident corners

$c_2 = prev(c_0)$; $c_1 = next(c_0)$;

$c_3 = O[c_0]$; $c_4 = next(c_3)$; $c_5 = prev(c_3)$;

$a = O[c_5]$; $b = O[c_2]$; $c = O[c_4]$; $d = O[c_1]$;

// Label incident vertices

$t = V[c_0]$; $v = V[c_1]$; $s = V[c_3]$;

// Perform swap

$V[c_1] = s$; $V[c_3] = v$; $V[c_4] = s$; $V[c_5] = t$;

// Reset opposite corners

$O[c_2] = c_3$; $O[c_0] = a$; $O[c_3] = c_2$; $O[c_4] = d$;

$O[c_5] = c$; $O[a] = c_0$; $O[c] = c_5$; $O[d] = c_4$;

Algorithm 2: Edge–Weld(c_0)

// Assign incidences

$c_1 = next(c_0)$; $c_2 = prev(c_0)$;

$c_4 = next(O[c_1])$; $c_5 = prev(O[c_1])$;

$a = O[next(O[c_5])]$; $b = O[prev(O[c_2])]$;

// Perform vertex removal

$V[c_0] = V[O[c_2]]$; $V[c_4] = V[c_0]$;

// Reset opposite corners

$O[c_5] = a$; $O[a] = c_5$; $O[b] = c_2$; $O[c_2] = b$;

Luiz Velho proved in [11] that the *Edge–Collapse* operation can be decomposed into a sequence of *Edge–Flips* operations, followed by one *Edge–Weld* operation. The Figure 5 illustrates this process.

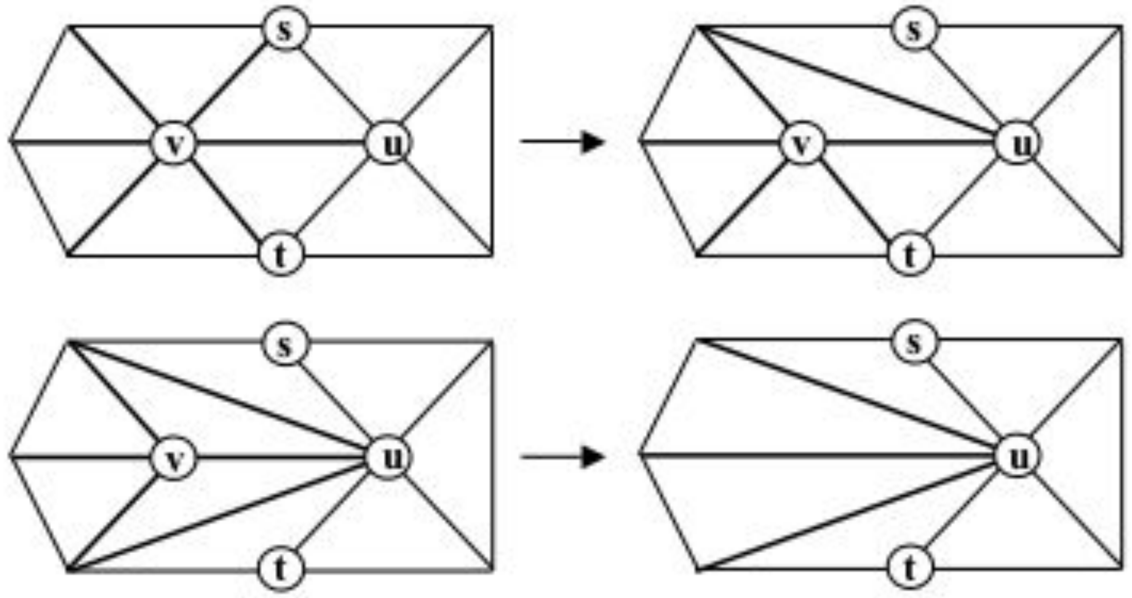


Figure 5. Edge–Collapse decomposition

Fast stellar mesh simplification

Algorithm 4: SimplifyFS(M, n)

---

assign quadrics;
for $(j = 1 \text{ to } n)$ **do**
  mark vertices as valid for removal
  while (exist a valid vertex) **do**
    for $(i = 1 \text{ to } 8)$ **do**
      v_i =valid random vertex
      if $E(v_i) < E(v)$ **then** $v = v_i$
    perform edge swaps to bring v to valence 4
    unmark the vertices $w \in link(v)$
    remove vertex (v)
    re–compute quadrics Q_u and Q_w

> Reinventa el mundo <

G20 AWARDS

TRANSFORMATEC

UTEC
UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

> Reinventa el mundo <

ACTIVITY (TO UPLOAD IN CANVAS)

TRANSFORMATEC

Activity (in groups)

- Implement a program to model a sphere, a cylinder and a cube. Find different textures and apply each of them to all the 3 models created.
- Your code should generate OFF files that show a sphere with texture, a cylinder with texture, etc.
- Example:

Summary of today

- Texture mapping.
- Mesh parametrization.
- Mesh simplification.

Thank you