Tesselation

Englisch: Tessellation

VFX

GPU-beschleunigte Geometrie-Unterteilung in 3D-Rendering — erhöht Poly-Count dynamisch für Details ohne Asset-Speicher zu sprengen. Standard in Real-Time-VFX und Game-Engines.

Tessellation — oder Tesselation — ist einer der wichtigsten GPU-Tricks, um komplexe Geometrien in Echtzeit zu rendern, ohne die Speicher- und Rechenlast zu sprengen. Statt bereits am Set oder in der Asset-Pipeline millionen von Polygonen zu modellieren, startet man mit einer dünneren Base-Mesh und lässt die GPU sie dynamisch unterteilen. Der Hardware-Tessellator sitzt zwischen Vertex- und Fragment-Shader und kann Flächen in wenigen Millisekunden zerlegen — je nach Entfernung zur Kamera, je nach Oberflächenkrümmung, je nach Displacement-Map. In der Praxis nutzen wir Tessellation vor allem für drei Szenarien: erstens für High-Poly-Details auf Charakteren und Umgebungen — Falten in Tüchern, Unebenheiten in Beton, Poren in Haut — ohne die Base-Mesh mit 10 Millionen Vertices zu ersticken. Zweitens für adaptive Geometrie-LOD (Level of Detail), die bei Kamera-Nähe automatisch detaillierter wird und bei Distanz wieder ausdünnt. Das spart gigantisch Bandbreite. Drittens für Wasser-, Stoff- und Haut-Simulation im Real-Time-VFX, wo man ein grobes Cloth-Mesh tesselliert, um dann per Displacement-Map oder Normal-Map feinere Bewegungen reinzulegen. Der Vorteil gegenüber Pre-Computed High-Poly-Assets: dynamisch, speichereffizient, und die Detailmenge passt sich dem aktuellen Frame-Budget an. Wichtig beim Workflow: Tessellation braucht verlässliche UV-Layouts und korrektes Normal-Baking. Wenn die Base-Mesh geknickte Normals hat, sieht das Tessellat-Ergebnis furchtbar aus — wie ein zerknittertes Papier statt glatter Oberfläche. Displacement-Maps müssen sauber in 16-Bit oder höher sein, sonst entstehen Artefakte. Und: Tessellation kostet GPU-Power — nicht endlos skalierbar. Ein 4K-Shot mit überaggressivem Tess-Factor kann einen RTX-Render auch bremsen. Man braucht Augenmaß: genug Detail für die Close-Ups, aber nicht mehr als nötig. In Echtzeit-Engines wie Unreal Engine oder Unity ist Tessellation Standard in modernen Viewport-Renderern. Für Film-VFX (Arnold, RenderMan, V-Ray) kommt es weniger zum Einsatz — dort precomputed man oft lieber dichte Meshes voraus. Aber in Interactive VFX, Game-Cinema und Live-Action-Compositing mit Real-Time-3D (Unreal Engine mit Neon oder ähnlich) ist Tessellation das Rückgrat für Performance und visuellen Reichtum zugleich.