

3.6 曲面细分阶段

曲面细分阶段允许我们渲染曲面。GPU的任务是获取每个表面描述并将其转换为一组有代表性的三角形。这个阶段是一个可选的GPU功能，它首先在DirectX 11中可用（并且是必需的）它也在OpenGL 4.0和OpenGL ES 3.2中得到支持。

使用曲面细分阶段有几个优点。曲面描述通常比提供相应的三角形本身更紧凑。除了节省内存之外，此功能还可以防止CPU和GPU之间的总线成为动画角色或对象的瓶颈，其每帧的形状都在变化。通过为给定视图生成适当数量的三角形，可以有效地渲染表面。例如，如果一个球离相机很远，则只需要几个三角形。近距离观察，它可能看起来最好用数千个三角形来表示。这种控制细节级别的能力还可以让应用程序控制其性能，例如，在较弱的GPU上使用较低质量的网格以保持帧速率。通常由平面表示的模型可以转换为三角形的精细网格，然后根据需要进行扭曲[1493]，或者可以将它们细分，以便不那么频繁地执行昂贵的着色计算[225]。

曲面细分阶段总是由三个元素组成。使用DirectX的术语，它们是外壳着色器、曲面细分器和域着色器。在OpenGL中，外壳着色器是曲面细分控制着色器，域着色器是曲面细分评估着色器，它们虽然冗长，但更具描述性。固定功能曲面细分器在OpenGL中称为图元生成器，正如将要看到的确实做法。

[第17章](#)详细讨论了如何指定和细分曲线和曲面。这里我们简要总结了每个细分阶段的目的。首先，外壳着色器的输入是一个特殊的面片图元。这包括定义细分曲面、贝塞尔曲面或其他类型的弯曲元素的几个控制点。外壳着色器有两个功能。首先，它告诉曲面细分器应该生成多少个三角形，以及在什么配置中。其次，它对每个控制点进行处理。此外，可选地，外壳着色器可以修改传入的面片描述，根据需要添加或删除控制点。外壳着色器将其控制点集以及曲面细分控制数据输出到域着色器。参见图3.9。

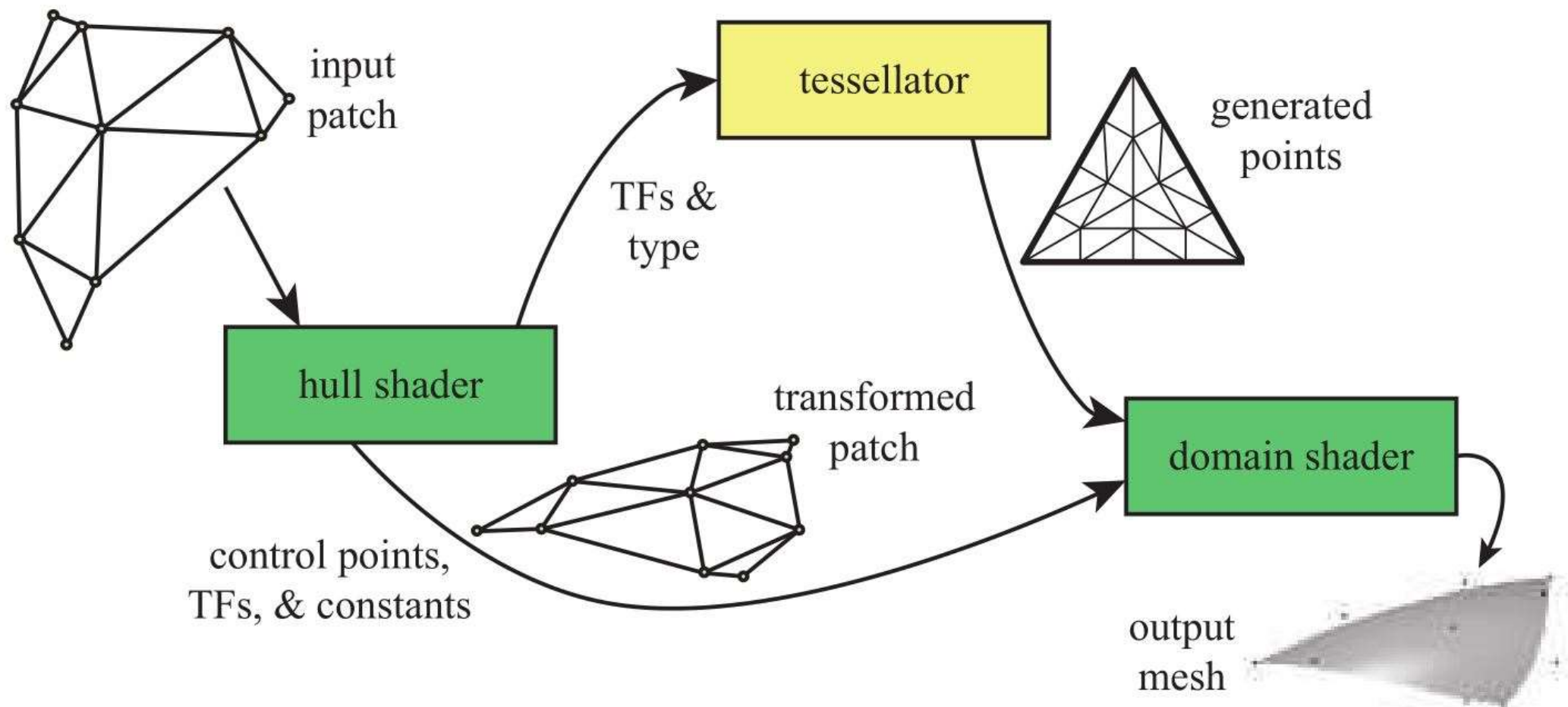


图3.9. 曲面细分阶段。外壳着色器接收由控制点定义的面片。它将曲面细分因子(TF)和类型发送到固定功能曲面细分器。控制点集由外壳着色器根据需要进行转换，并与TF和相关面片常量一起发送到域着色器。曲面细分器创建一组顶点及其重心坐标。然后这些由域着色器处理，生成三角形网格（显示控制点以供参考）。

曲面细分器是管线中的固定功能阶段，仅与曲面细分着色器一起使用。它的任务是为域着色器添加几个新的顶点进行处理。外壳着色器向曲面细分器发送有关所需曲面细分类型的信息：三角形、四边形或等值线。等值线是一组线带，有时用于头发渲染[1954]。外壳着色器发送的其他重要值是曲面细分因子（OpenGL中的曲面细分级别）。它们有两种类型：内边缘和外边缘。这两个内部因素决定了三角形或四边形内部出现多少细分。外部因素决定了每个外部边缘的分裂程度（第17.6节）。图3.10显示了增加曲面细分因子的示例。通过允许单独的控件，我们可以让相邻曲面的边缘在细分中

匹配，而不管内部是如何细分的。匹配边缘可避免在面片相交处出现裂缝或其他阴影伪影。顶点被分配了重心坐标（第22.8节），这些值指定了所需表面上每个点的相对位置。

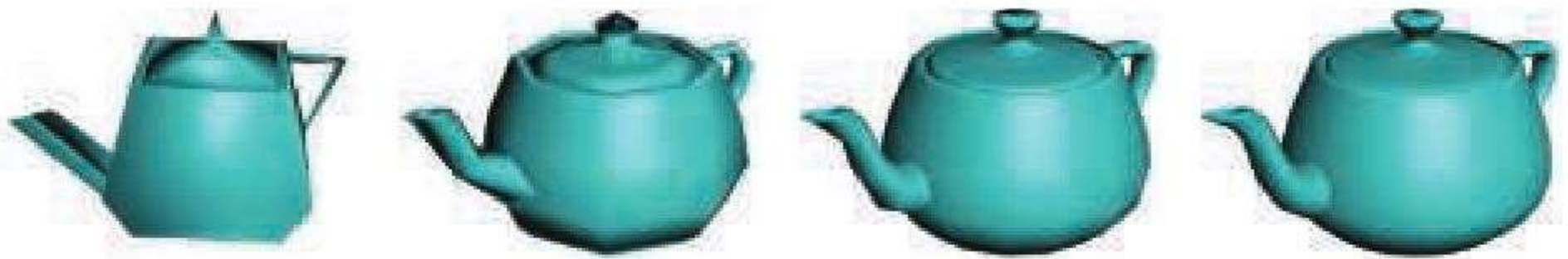


图3.10. 改变曲面细分因子的影响。犹他茶壶由32个面片组成。内部和外部曲面细分因子，从左到右，分别为 1、2、4 和 8。（由Rideout和Van Gelder的演示生成的图像[1493]）

外壳着色器总是输出一个面片，一组控制点位置。但是，它可以通过向曲面细分器发送零或更低（或非数字，NaN）的外部曲面细分级别来表示要丢弃面片。否则，曲面细分器会生成一个网格并将其发送到域着色器。来自外壳着色器的曲面控制点，被域着色器的每次调用用于计算每个顶点的输出值。域着色器具有类似于顶点着色器的数据流模式，来自曲面细分器的每个输入顶点都被处理并生成相应的输出顶点。形成的三角形然后沿管线向下传递。

虽然这个系统听起来很复杂，但它的结构是为了提高效率，而且每个着色器都可以相当简单。传递到外壳着色器的面片通常很少或没有修改。该着色器还可以使用面片的估计距离或屏幕大小来动态计算曲面细分因子，如地形渲染[466]。或者，外壳着色器可以简单地传递一组固定值，用于应用程序计算和提供的所有面片。曲面细分器执行一个复杂但功能固定的过程，即生成顶点、给它们位置并指定它们形成的三角形或线。该数据放大步骤在着色器之外执行以提高计算效率 [530]。域着色器采用为每个点生成的重心坐标，并在面片的评估方程中使用这些坐标来生成位置、法线、纹理坐标和其他所需的顶点信息。有关示例，请参见图3.11。

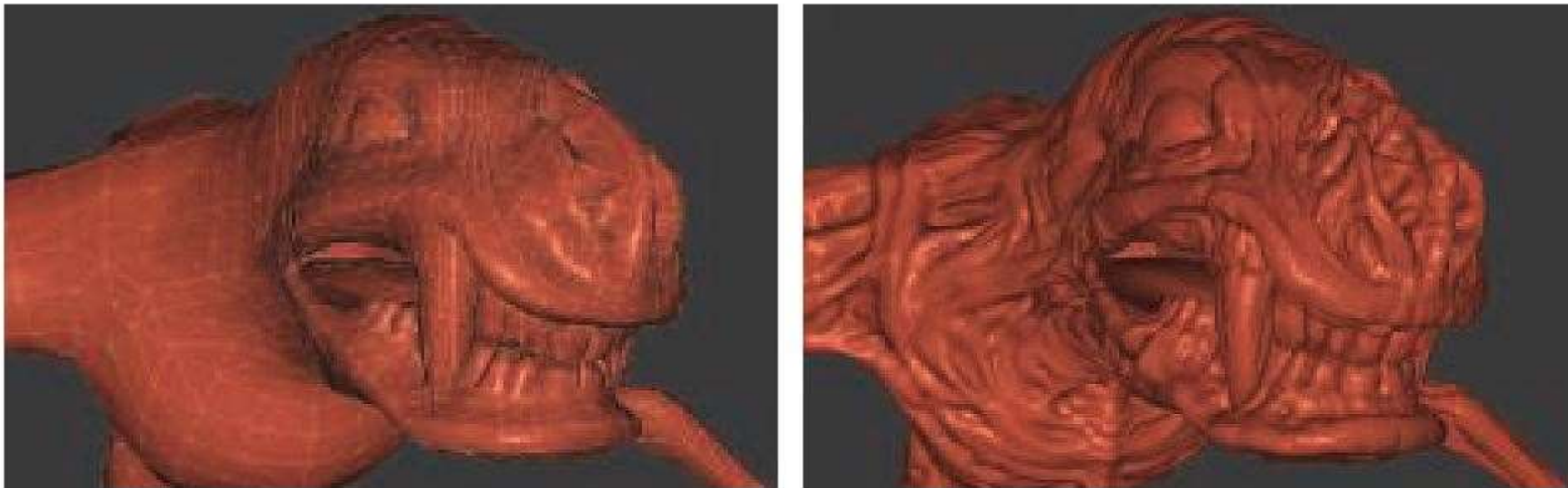


图3.11. 左边是大约 6000个三角形的底层网格。在右侧，每个三角形都使用PN三角形细分进行细分和置换。（来自NVIDIA SDK 11[1301]样本的图像，由NVIDIA Corporation提供，模型来自4A Games工作室的游戏《地铁2033》。）