

虚平面映射:深度图像内部视点的绘制技术*

盛斌¹⁺, 吴恩华^{1,2}

¹(澳门大学 科技学院 计算机与信息科学系,澳门)

²(中国科学院 软件研究所 计算机科学国家重点实验室,北京 100190)

Virtue Plane Mapping: A Method of Rendering into Depth Images

SHENG Bin¹⁺, WU En-Hua^{1,2}

¹(Department of Computer and Information Science, Faculty of Science and Technology, University of Macau, Macao, China)

²(State Key Laboratory of Computer Science, Institute of Software, The Chinese Academy of Sciences, Beijing 100190, China)

+ Corresponding author: E-mail: shengbinben@hotmail.com

Sheng B, Wu EH. Virtue plane mapping: A method of rendering into depth images. *Journal of Software*, 2008,19(7):1806–1816. <http://www.jos.org.cn/1000-9825/19/1806.htm>

Abstract: An effective method of depth image based rendering is proposed by applying texture mapping onto virtual but pre-defined multiple planes in the scene, called virtual planes. The method allows the viewpoint either static or moving around, including crossing the plane of the source images. By the approach, the relief textures from depth images are mapped onto the virtual planes, and through the pre-warping process, the virtual planes are converted into standard polygonal textures. After the virtual plane mosaics, the resultant image that supports 3D objects and immersive scenes can be generated by polygonal texture mapping. In addition, both hardware and software implementation of the method can increase the power of conventional texture mapping in image based rendering. In particular, the scope of the viewpoint can be extended into the inner space of depth images, and as a result, a novel framework is provided for constructing real-time walkthrough systems as well as for panoramic modeling from an arbitrary viewpoint in the depth image space.

Key words: image-based modeling and rendering; relief texture mapping; depth image; virtual plane

摘要: 首先推导与归纳了图像三维变换中像素深度场的变换规律,同时提出了基于深度场和极线原则的像素可见性方法,根据上述理论和方法,提出一种基于深度图像的建模与绘制(image-based modeling and rendering,简称IBMR)技术,称为虚平面映射.该技术可以基于图像空间内任意视点,对场景进行绘制.绘制时,先在场景中根据视线建立若干虚拟平面,将源深度图像中的像素转换到虚平面上,然后通过对虚平面上像素的中间变换,将虚平面转换成平面纹理,再利用虚平面的相互拼接,将视点的成像以平面纹理映射的方式完成.新方法还能在深度图像内侧,基于当前视点快速获得该视点的全景图,从而实现视点的实时漫游.新方法视点运动空间大、存储需求小,且可以发挥图形硬件的纹理映射功能,并能表现物体表面的三维凹凸细节和成像视差效果,克服了此前类似算法的局限和不足.

* Supported by the National Natural Science Foundation of China under Grant No.60473105 (国家自然科学基金); the National Basic Research Program of China under Grant No.2002CB312102 (国家重点基础研究发展计划(973)); the Studentship Grant of University of Macau (澳门大学研究生奖学金)

Received 2006-08-13; Accepted 2007-01-23

关键词: 基于图像的建模与绘制;浮雕纹理映射;深度图像;虚平面

中图法分类号: TP391 文献标识码: A

近年来,计算机图形硬件,特别是图形处理器(graphics processing unit,简称 GPU)在性能上的飞速发展,使得传统的基于多边形网格的图形建模与绘制技术的能力获得了极大的释放.同时,基于图像的建模和绘制(image-based modeling and rendering,简称 IBMR)作为另一种流行的图形技术,也已广泛应用于虚拟现实、计算机动画、电影和游戏等领域中.该技术的突出特点在于,图像数据可以从预计算后的场景或者真实场景中采样,使得该技术无须构造几何模型和进行光照运算即可基于新视点生成具有真实感的图像,而且其算法复杂性独立于场景中几何元素的数量,只与源图像的信息量有关,因此可以很容易地实现场景的实时绘制.由于目前的图形处理硬件可以很好地与传统的图形绘制流水线相结合,在当前的技术环境下,如何利用图形硬件日新月异的处理能力来提高 IBMR 的处理与成像能力,是当前对图形学研究人员提出的一项挑战,也成为 IBMR 领域中一个重要而活跃的研究方向.

为此,很多研究关注于利用图形硬件来承担IBMR中部分繁重的成像计算.其中,作为三维图像变换(3-D image warping)^[1]的一种特殊处理,浮雕纹理映射(relief texture mapping)^[2,3]能够高效地利用图形硬件的纹理映射能力进行加速,同时高质量地反映物体表面的三维凹凸细节及视差变化.该算法本身还具有较高的并行能力,使其可以方便地基于硬件加速.因此,浮雕纹理映射是一种优越而前景广阔的技术,近年来被不断地研究并加以应用.

值得注意的是,浮雕纹理映射需要通过图像的深度信息生成新视点的图像.但是,目前基于深度图像的 IBMR 算法,包括浮雕纹理映射技术,通常要将视点限制在源图像平面的外侧,即视点不能进入深度图像内部对新视点进行成像和绘制,更无法实现视点在深度图像空间中的漫游.针对当前的这些技术局限性,本文首先总结了图像深度域的变换规律,对于图像三维变换过程中,其像素深度域的转换关系进行了量化分析与归纳,完善了三维变换的理论基础.同时,本文提出了一种利用像素和极点相对位置的变化来判断像素可见性的方法,可以快速地判断源深度图像中各像素相对于新视点的可见性.在此基础上,新算法不再局限于将源图像的像素变换到新视点图像的平面上,而是可以将深度图像中的像素变换到任意空间平面上,再通过纹理映射的方式获得成像,解决了在浮雕纹理映射算法中视点移入图像时发生的成像错误,如图 1 所示.同时,新算法还能在深度图像内侧,通过虚平面拼接技术快速获得新视点的全景图,从而实现视点的实时漫游.本文最后的测试结果证明了深度域的变换规律以及虚平面映射和拼接技术的正确性和优越性,较好地弥补了前面算法中存在的不足.

本文第 1 节介绍相关工作.第 2 节介绍本文算法的理论基础.第 3 节讨论通过虚平面映射和拼接技术进行目标成像和全景图生成.第 4 节介绍实验结果.第 5 节进行总结和展望.

1 相关工作

较早的基于图像的绘制方法是全景图^[4,5].它由环绕观察者一周的连续图像构成,观察者可以通过水平旋转视点 360°以及上仰或者下俯一定角度来浏览场景.此后,Szeliski给出了全景图的拼接方法^[6].目前,全景图技术已经获得了广泛的应用.

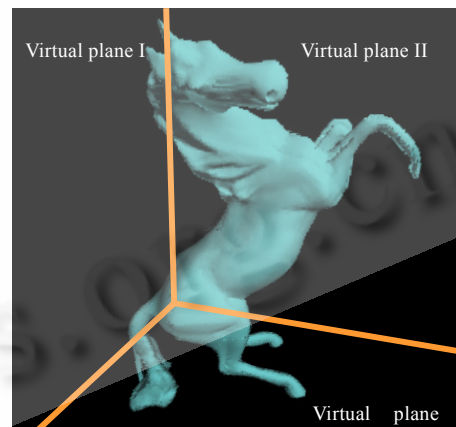


Fig.1 Rendering horse model by virtual plane mapping

图 1 用虚平面映射在深度图像空间中的绘制示例

全景图的缺陷在于观察者的位置移动时无法提供连续的图像,且视差效果不理想,而全光函数在于对场景中所有光线的量化描述.Adelson提出了7D全光函数(plenoptic function)^[7]的概念, $p=P(\theta, \psi, \lambda, V_x, V_y, V_z, t)$.其中, p 为光的亮度, (V_x, V_y, V_z) 表示视点的三维坐标, (θ, ψ) 定义了该视点的视线方向, λ 为光的波长, t 为时刻.可见,这个函数非常复杂,其采样、存储和处理都极为困难.Levoy引入了Light Field方法^[8],同时,Gortler也提出了类似的方法Lumigraph^[9],其实质在于利用光线的传播不变性,采用共用光线的方式并略去时间和波长因素,将全光函数降到4D,得到了较为可行的全光函数绘制模式.沈向洋等人提出的同心圆拼接算法(concentric mosaics)^[10]忽略了场景中漫游时的竖直误差,同时,通过将视点限定在一定范围内,将全光函数降到了3D,成为一种更加有效的绘制方法.

1.1 基于图像的绘制技术

一般而言,当前最为常用的基于图像的绘制就是精灵(sprite)技术^[11].其中,精灵是指在屏幕上可以任意移动的小块图像,比如视窗系统中屏幕上表示鼠标的箭头图标就可看成是简单的精灵.而在场景的绘制中,由于画面具有时序相关性,在视点移动中,较远景物受视差影响小,成像变化不大.这样就可以将远距离景物在画面中的投影图像作为精灵进行保存,再通过纹理映射近似取代该景物在画面上的效果,达到减少当前场景复杂度的目的.精灵技术适合于绘制远处的复杂场景,同时对于绘制场景中整体移动的物体也十分有效.另一种十分类似的基于图像的绘制技术被称为布告板(billboard)技术^[12],布告板是指时刻面向观察者的多边形图像,当观察方位改变时,布告板的朝向也随之改变.采用布告板技术可以较好地表达一些具有几何对称和实体特征远处场景.在当前的虚拟现实系统中,布告板被广泛地用来表达树木、云、火、雾和花等景物.

然而,无论是精灵还是布告板,都难以模拟景物表面的微观细节,也无法使虚拟场景的视差效果更为真实、自然.而将纹理映射,特别是表现物体三维细节和视差变化的位移映射(displacement mapping)技术^[13],与基于图像的绘制技术相结合,成为研究者们解决这个问题的有效途径.

1.2 浮雕纹理映射

IBMR的核心在于对全光函数的采样和绘制.全光函数描述了场景中光线的所有分布,给出了场景的精确描述.但是,基于全光函数的IBMR算法通常都需要非常大的存储空间,其中一个重要原因是通过增加参考图像的数量获得场景中未知的可见性信息.而解决这个问题的一个有效办法是在源图像中提供可见点的深度信息.基于三维图像变换的IBMR方法就是从带有深度信息的源图像生成当前视点的目标图像.具体地说,就是利用在一些离散观察位置和视角下获得的图像,配合相应的深度信息和相机参数,生成针对新的观察位置和视角的图像.其基本方法是利用深度信息或其他一些几何信息对源像素进行偏移操作.McMillan在其博士论文中对该技术作了较为全面的介绍.此外,Oliveira提出的浮雕纹理映射将三维图像变换公式拆分成对源图像先后进行的两步运算:一维变换和二维纹理映射.这样,算法就可以在第2步普通纹理映射中利用硬件进行加速.为此,作为该算法的前变换(pre-warping),一维变换会将源图像变换成满足纹理映射要求的中间图像.这样,浮雕纹理映射的具体思想可以总结成:源图像上的每一像素 (u_s, v_s) 在中间图像上的对应点 (u_i, v_i) 经二维纹理映射变换到目标图像上的对应位置,与 (u_s, v_s) 直接经过三维图像变换后的目标位置一致.在此,中间图像可以通过对源图像上的每个像素在行与列上分别进行位移获得,位移大小取决于像素的深度和视点位置.该方法能够高效地反映三维凹凸细节和视差变化,并且基于纹理映射的成像速度很快.但由于浮雕纹理映射将中间图像仅能成像于源图像平面上,实际上限制了视点的活动范围.同时,浮雕纹理映射算法中还缺乏对像素可见性的判别,容易出现错误^[4].

本文就是一种基于深度图像的绘制算法.本文算法通过虚平面的映射和拼接来实现深度图像到成像面的映射,这样,物体表面间的相互遮挡、投影变换及纹理影射等都可以方便地借助图形加速硬件来实现.

2 理论基础

基于三维图像变换的IBMR方法从带有深度信息的源参考图像生成当前视点目标图像.这类方法以投影

几何为理论基础,将源深度图像中的像素变换到目标图像的对应位置,保证成像的正确性.不过,由于三维变换注重于像素在图像平面上的二维变换的讨论,且采用了非欧几何的解析方法,因此对于像素的深度域转换的情况讨论并不充分.本节中,我们对三维变换在欧式几何空间中进行分析,归纳出空间点深度域变换的规律,进而提出了快速判别深度图像空间中的像素可见性的方法,形成了本文算法的理论基础.

2.1 像素深度场转换

三维图像变换^[7]描述了空间点在不同图像平面上的对应像素间的变换关系.它根据带有深度信息的源图像 i_s 生成关于新视点的目标图像 i_t ,源图像所表达场景的三维几何信息是通过每个可见点的深度和此图像的针孔相机模型来表示的.假设空间中一点 $X(x,y,z)$ 在源图像平面 i_s 上的投影点 x_s 的坐标是 (u_s, v_s) ,则这一点在目标图像 i_t 上的对应点可以表示为

$$C_s + P_s \bar{x}_s = X = C_t + P_t \bar{x}_t t_t(u_t, v_t) \tag{1}$$

这里, P_s 和 P_t 分别是源图像 i_s 和目标图像 i_t 的相机矩阵, $x_s=[u_s, v_s, 1]^T$, C_s 和 C_t 分别是图像 i_s 和 i_t 的针孔相机模型的投影中心.值得注意的是,式(1)中 $t_t(u_t, v_t)$ 是一个重要的参数值,它实际上是比例标量.为方便理解,其具体形式可以改写为

$$t_t(u_t, v_t) = \frac{z_k(u_t, v_t)}{D_{It}}$$

这里, $z_k(u_t, v_t)|_{k=s,t}$ 表示像素所对应的空间位置 X 相对于视点的深度,而 $D_{Ik}|_{k=s,t}$ 则表示了图像平面 $I_k|_{k=s,t}$ 相对于视点的深度.由此可知, $t_t(u_t, v_t)$ 的几何意义是成像体相对于成像面的相对深度值,而其倒数形式通常在投影几何中被称为普遍视差(generalized disparity) δ .

在浮雕纹理映射中,由于浮雕纹理是通过正投影获得的深度图像,其像素的深度为相对于成像平面的垂直距离,可以用 d_s 表示.对于式(1),与传统的三维图像变换基于投影几何求解不同,本文在欧式几何空间进行了严格推导,由此得到以下结果:

$$\begin{cases} u_t = \frac{m_{11}u_s + m_{12}v_s + m_{13}d_s + s_1}{m_{31}u_s + m_{32}v_s + m_{33}d_s + s_3} \\ v_t = \frac{m_{21}u_s + m_{22}v_s + m_{23}d_s + s_2}{m_{31}u_s + m_{32}v_s + m_{33}d_s + s_3} \\ z_t = D_{It}[(m_{31}u_s + m_{32}v_s + m_{33}d_s) + s_3] \end{cases} \tag{2}$$

其中, $(m_{ij})_{3 \times 3} = P_t^{-1}P_s$, $(s_i |_{i=1,2,3})_{3 \times 1} = P_t^{-1}(C_s - C_t)$.

在传统的浮雕纹理映射算法中,在其前变换过程中引入以上推导结果,就可以求得中间图像中关于像素偏移量的简洁表达:

$$\begin{cases} u_i = u_s + (e_u - u_s)r_s \\ v_i = v_s + (e_v - v_s)r_s \end{cases} \tag{3}$$

其中, $r_s=(1-\delta(u_t, v_t)/\gamma)$.

这里需要注意的是,上式中的 γ 是相机(视点)的成像参数,当 $\gamma < 0$ 时,称为“后”成像模式,光线到达并穿越视点后,在成像平面形成倒立的图像;反之,当 $\gamma > 0$ 时,称为“前”成像模式,光线在到达视点前穿过成像平面形成正立的图像.而我们可以通过简单的反转图像操作将这两种成像模式形成的图像相互转化.本文算法为讨论方便,不失一般性,采用“前”成像模式来设置成像面,即 $\gamma > 0$.而根据本文推导得出的式(3)可以发现,在进行生成中间图像的前变换中,所得中间图像上每个像素沿着极线(epipolar line)^[2]方向进行位移,位移值与该像素的普遍视差 δ 相关.由此可以得到关于新视点的中间图像,再经过纹理映射就可以最终成像.

2.2 过极点可见性判别

IBMR技术中需要解决的一个重要问题就是处理视点变化时的像素可见性问题.对此,McMillan和Bishop提

出极线排序法^[2]来解决该问题.作为一个类画家算法,极线排序法可以保证在图像映射中,相对于当前视点处于相同成像位置的若干像素,能够由远到近地进行处理,从而正确地反映场景中的遮挡关系.

极点排序算法可以简要描述为:首先,计算新视点(相机位置)在源参考图像平面上的投影为极点(epipole).它实际上是源视点与新视点的连线,与源参考图像平面 I_s 的交点.源参考图像平面在极点处沿水平和垂直方向分割成 4 个矩形子区域.如果新视点与源视点位于源图像平面的同一侧,那么当进行三维图像变换时,对于每个子区域中的像素按朝向极点的顺序进行处理;反之,若新视点与源视点分处于源图像平面的两侧,则每个区域中的像素按照背向极点的顺序处理.

在浮雕纹理映射中,由于源深度图像由正投影获得,为了确定在源图像上的极点位置,不失一般性,可以将当前视点在源图像平面上直接作正投影,即为所求极点.

极线排序算法可在生成的图像中保证像素间正确的遮挡关系.由此,我们还能得到一个重要结论:在图像的三维变换中,图像中某像素 P 与其他像素间的相互遮挡现象只发生在一条射线区域内,这条射线从极点出发,且指向该像素 P .特别地,当图像平面被极点分割成 4 个矩形区域时,不同区域间的像素是不会发生相互遮挡的.

作为三维图像变换的特例,浮雕纹理映射在应用极线排序的过程中同样可以发现,像素在前变换过程中的偏移是从该像素点出发,相对于极点位置进行了发散或聚拢.这里有一个值得关注的问题:虽然当视点移入深度图像内部时极线原则在理论上仍然适用,但是在某些情况下会出现像素点在图像平面上成“虚像”的情况,即某些像素由于深度关系,对于视点和视窗而言是不可见的,但根据传统的浮雕纹理前变换公式,这些像素仍然会进入中间图像,并出现在最终成像结果上.这个问题最早曾被Oliveira在关于浮雕纹理的SIGGRAPH论文^[3]中有所提及,但据我们所知,至今在公开发表的文献中还没有找到快速、有效的解决办法.为了解决这个错误现象,必须在浮雕纹理的前变换过程中剔除这些不可见的像素.为叙述方便,本节着重分析源图像像素通过三维变换映射到虚平面上后,如何剔除上述不可见的像素.至于将源图像像素变换到虚平面上进行成像的处理过程会在下一节具体讨论.

如图 2 所示,点 A 和点 B 是源深度图像中某两个像素相对应的几何位置,而 A_0 和 B_0 分别是 A 点和 B 点各自在虚平面上的正投影位置.这样,如果基于当前的视点在虚平面上进行的前变换,像素 A_0 就会偏移到 A_1 处,而像素 B_0 就会变换到 B_1 的位置;另一方面,如果基于当前视点 COP 来观察虚平面,那么点 A 是可见的,而点 B 是不可见的.也就是说,点 A_0 到 A_1 的偏移会形成正确的成像,而由 B_0 到 B_1 的变换则会带来错误的成像效果.

如果利用上节中所推导的结果,可知,当 $\delta_i(u_i, v_i) > 0$ 且 $\gamma > 0$ 时,源图像的像素会在目标成像面上正确成像;反之,像素的变换会导致“虚像”出现.由此,结合公式(3)就可以得到前变换过程中的可见性判别关系式:

$$\frac{e_u - u_i}{e_u - u_s} = \frac{e_v - v_i}{e_v - v_s} = \frac{\delta_i(u_i, v_i)}{\gamma} > 0 \quad (4)$$

由以上关系式可知,在前变换中,正确的偏移结果不会跨越极点,错误的偏移结果则必然要跨越极点.这一发现与极点原则本身的内在意义是一致的.平面上的任何像素在三维坐标变换中都只在极点的一侧进行偏移和相互遮挡,不存在跨越极点进行偏移和相互遮挡的情况.如果发生过极点偏移的情况,则意味着像素本身已不可见,像素的成像并不来自于真实光线,而是由光线的反向延长线形成的“虚像”,需要进行剔除.

本文工作对浮雕纹理映射进行了推广,即不再将源深度图像中的像素直接在源图像平面偏移而形成中间图像,而是先将像素映射到空间中的虚平面上再进行前变换.由图 3 可知,当像素点的深度位于视点前方(即该像素可见)时,该像素的正投影位置与其进行前变换后的位置同处于极点的一侧;反之,当像素的深度位于视点后方(即该像素不可见)时,该像素的正投影位置与其进行前变换后的位置则分处于极点的两侧.由此可以得到像素在虚平面上进行浮雕纹理前变换的可见性判断方法——过极点可见性判别.

为了快速、有效地利用像素的过极点偏移现象来判断像素在虚平面上的可见性,受McMillan极线排序算法思想的启发,实际中我们也将平面依照极点的位置划分成 4 片矩形区域(由于受极点的实际位置和虚平面实际大小的影响,实际上矩形区域的数目可能只有 1 个或者两个).如图 3 所示,在前变换过程中,图像平面分成了 Sheet I, Sheet II, Sheet III 和 Sheet IV 这 4 个矩形区域.将虚平面上的像素按照极点原则依次进行偏移,再根据该像

素在前变换过程中的偏移情况进行判断:如果像素在偏移过程中没有跨越自己原先所在的矩形区域(如图 3 中 A 偏移到 A_1 或者 A 偏移到 A_2 的情况),则说明该像素经过前变换后仍能在虚平面上正确成像,应予以保留;反之,如果像素在偏移过程中跨越了自己原先所在的矩形区域(如图 3 中 A 偏移到 A' 中的情况),则说明该像素经过前变换后在虚平面上成“虚像”,需要被剔除.

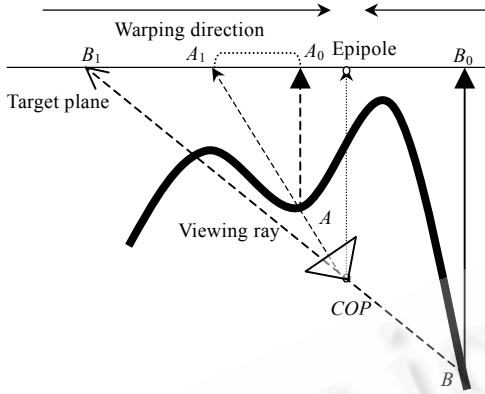


Fig.2 Over-Epipole warping the invisible texel
图 2 过极点偏移与像素可见性示意图

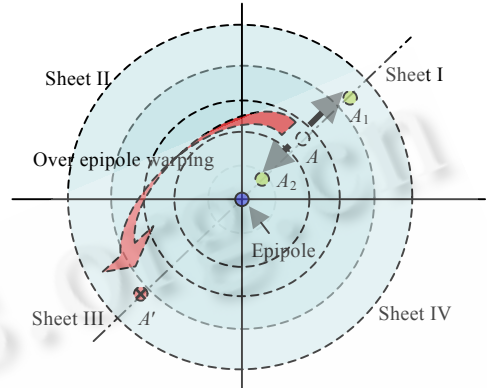


Fig.3 Over-Epipole warping check
图 3 过极点可见性判别

3 基于虚平面的成像绘制

在浮雕纹理映射算法中,图像三维变换是在浮雕纹理图像平面上进行的,当视点进入深度图像空间中时,往往会使一些原先不可见的物体变得可见,出现错误的成像效果.根据图像的深度信息,本节中所提出的新算法不再局限于将像素在图像平面上进行变换,而是首先将深度图像中的像素映射到空间中某一相对于视点可见的虚拟平面上,然后再在该虚拟平面上相对于视点进行源像素的前变换,最后将该虚拟平面以纹理映射的方式显示出来,从而得到最终成像,如图 4 所示.

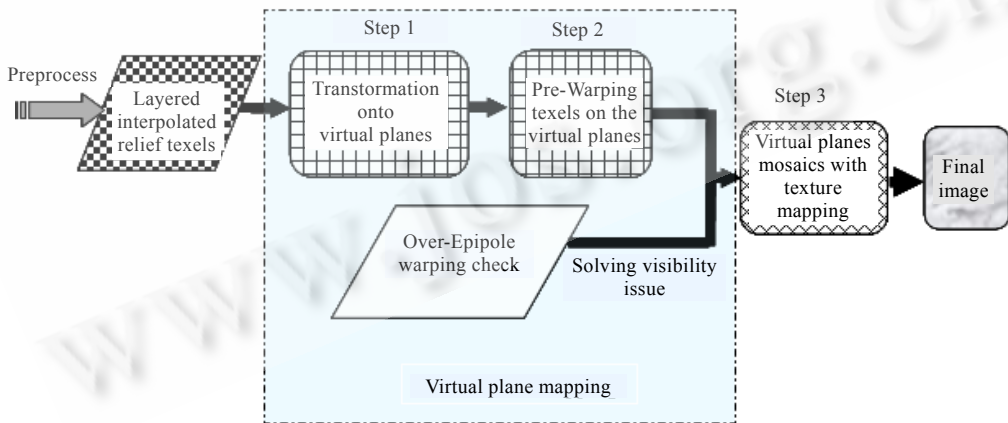


Fig.4 The pipeline for virtual plane mapping and mosaics
图 4 虚平面映射与拼接流程示意图

3.1 虚平面映射技术

从图像的三维变换的公式中可以发现,深度图像的三维变换过程在一定条件下可以转化为关于像素坐标的线性变换过程,即可以利用纹理映射的方式来完成.这样做可以更加充分地利用图形硬件的能力,获得较好的图像显示效果.浮雕纹理映射正是基于三维变换的这一特性而产生的:该算法通过对源图像进行前变换,在源图

像平面上生成了中间图像,然后对该中间图像基于视点进行平面纹理映射而最终成像。

尽管浮雕纹理映射利用了纹理映射,使得成像速度大为提高,但其算法中对成像平面的选取有严格的限制:作为前变换生成的中间图像必须与源图像平面重合.这样就限制了视点活动范围,并且仅当源图像平面处于视点的可见范围内时,针对源图像上的像素进行浮雕纹理映射才是有效的,否则,当源图像平面相对于视点不可见时,源图像上的像素就无法进行正确前变换、纹理映射以及最终成像。

在虚平面映射技术中,有关前变换和纹理映射的技术可以被扩展到空间中任意设置的某个平面中,而不仅局限于源图像所在的平面.这样做不仅将分解图像三维变换过程并利用纹理映射进行成像的技术得到极大的推广,而且可以根据当前视点的位置和视线方向,由用户选择合适的平面进行成像,并由此使视点进入到深度图像内部进行漫游。

在浮雕纹理映射中还存在一个问题:该方法虽然提高了速度,但仍然不可避免地要进行图像空洞的填补操作,这样就降低了算法的运行效率.对此,我们在虚平面映射过程中结合了有关层次纹理技术^[13,14]的思路,将深度图像的各个像素进行分层处理,即深度图像转化为一系列插值纹理(interpolated texels,简称ITs),亦即根据一个像素及其邻近像素的深度值,沿着深度方向为该像素插值生成多个具有不同深度值的插值像素,它们一起构成该像素的插值纹理,这样就生成了场景的一种近似连续的三维表达.在进行虚平面映射时,只需利用图形硬件

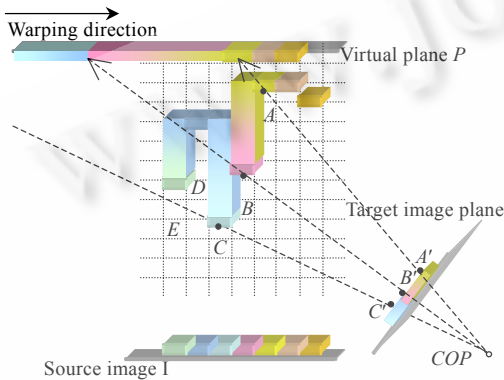


Fig.5 ITs pre-warped onto a virtual plane

图5 将插值纹理前变换到虚平面的示意图

来将这些插值纹理对应于虚拟平面处理,就可以快速生成目标图像,而不必进行空洞填补^[15],如图5所示.通过插值纹理生成的图像能够高质量地反映场景的三维细节及视差变化。

由于源图像插值纹理的生成可以在预处理过程中完成,虚平面映射的算法过程可具体分成以下3步:

(1) 通过图像的三维变换,将源深度图像中的插值纹理变换到某一给定的虚拟平面上。

由于源图像中的各个像素带有深度,根据本文前面第2节对于含深度像素的三维变换规律的总结,可以将源图像I上的像素映射到虚拟平面P上。

这里需要注意两个问题:首先,理论上虚拟平面可以任意给定,但是只有当虚拟平面相对于视点可见时

才有意义,此时还须基于虚拟平面建立局部坐标系.因为只有建立了该局部坐标系,才能基于虚拟平面对像素和视点分别进行定位,也才能使像素在图像与虚平面间的三维转换以及随后的纹理映射有意义。

其次,要按照极线原则的顺序依次处理源图像所有的像素,即按照相对于极点由远及近的顺序依次完成这些像素到目标平面的映射.在本文的实验中,由于源深度图像是采用了正投影的浮雕纹理,其源视点可被视为在无穷远处,所以将当前视点对源图像进行正投影,得到的投影点坐标即为源图像的极点坐标。

(2) 基于当前视点对虚平面上的插值纹理进行前变换,在虚平面上生成适应于二维纹理的中间图像。

经过设置虚平面以及插值纹理的转换后,我们就可以按照虚平面上极点处对虚平面进行水平和垂直的分割,形成4个矩形子区域.若极点落在该虚平面范围之外,则该虚平面被分割成两个或1个矩形区域.如果该虚平面相对于目标视点位置是可见的,则按极点排序算法的顺序依次对每个矩形区域内的插值纹理进行前变换,该变换使用了简化的一维前变换公式(1-D pre-warping equation),即本文中的式(3),来计算源浮雕纹理图像的像素在中间图像上的目标位置.而在前变换阶段,由于原相邻像素深度差过大所引起的中间图像上的空洞,则可以通过前面插值纹理的直接映射而填补^[15]。

在基于虚平面的前变换过程中,由于虚平面上像素的深度值是由三维变换得到的,所以其具体数值存在正负之分,因此不同于传统的浮雕纹理映射中深度值只有正值的情况.可以发现,无论像素深度值是正值还是负值,本文所采用的一维前变换公式都能有效适用.由此进一步分析可得,当像素深度为正值时,像素的偏移远离

虚拟平面上的极点位置;当像素深度为负值时,像素的偏移向虚拟平面上的极点位置靠近.

此外,在前变换过程中,可能出现某像素的深度小于当前视点的深度情况,这使得该像素在虚平面上会呈现“虚像”.为了避免这种错误现象的发生,我们采用前面提出的过极点可见性判别,就可以将这类像素迅速、有效地进行剔除.

(3) 根据当前视点和投影矩阵对虚拟平面上生成的中间图像进行纹理映射,进行成像.

与浮雕纹理映射类似,本文同样利用平面纹理映射技术来加速成像,即将每一个设置的虚平面都当成一个二维平面纹理,然后基于当前视点进行纹理映射,这样就可以利用当前图形硬件的纹理映射能力加速.

在实际中,对于虚平面的确定需要考虑各种情况:一方面,像素从源图像到虚拟平面的变换过程可以看成是对于源图像数据的一次重采样过程,所以,当虚平面与源图像平面夹角越小、距离越近时,采样密度越大,在虚拟平面上获得中间图像的质量也就越高;另一方面,像素在虚平面上经过前变换和纹理映射最终成像的过程也可以看成是基于视点对虚平面上像素的重采样过程,当虚平面与视平面夹角越小、距离越近时,采样密度越大,视平面上获得的最终图像效果越好.因此,虚平面的确定需要对这两方面因素加以折衷,通过对场景和视点的相对位置进行分析得到.

3.2 虚平面拼接与全景图合成技术

为了取得更好的成像效果,通常需要对相对于视点建立多个虚拟平面,成像时对每个虚拟平面以平面纹理的方式进行拼接,这就需要使用虚平面的拼接技术.

采用虚平面拼接的主要原因有两个:首先,实际中虚平面的大小是有限的,而视点位置的改变或者视线方向的变动会使虚平面在经过透视投影后不能完全覆盖视平面窗口,直接导致成像的丢失;其次,即使虚平面在映射后可以覆盖视窗,单一的虚平面对源图像仍很难得到较好的采样效果,最后,在成像时会使得纹理的走样现象变得明显,不利于成像质量的提高.因此,实际中常常在场景中根据成像的需要设置多个虚拟平面,再将这些虚平面作为平面纹理进行相互的拼接,从而获得较好的成像效果.

理论上,关于虚平面的设置与选择还是相当灵活的.我们可以采用一种直观的设置策略:将场景中的虚平面直接设置为场景包围盒的各个内表面上.这样做的好处在于:(1) 通过在包围盒的内表面上成像,可以使视点在场景内部进行任意移动,从而使得虚平面的设置具有与视点位置无关的特点,简单、易行;(2) 在基于图像的绘制中能够方便地基于新视点生成全景图,就可以实现对于场景实时漫游.而利用在包围盒内内表面的映射和拼接,可以通过纹理映射来直接获得当前视点的全景图;(3) 由于包围盒的内表面在几何上呈现两两垂直、对面平行的特点,从而使源图像像素在各个虚平面上的相互转换变得快速、简便.

为了使视点在场景中任意位置都能生成全景图,本文借鉴 cube map 环境映射的基本思想:首先针对整个场景和视点建立虚拟包围盒,再把包围盒的每个表面各附着一个虚拟平面,这样就可以按照虚平面映射和拼接的办法将源图像像素变换到虚平面上,之后对包围盒表面上的虚平面相互拼接,就可以获得基于新视点全景图.

图 6 给出了多个虚平面进行相互拼接后,对维纳斯模型进行成像的实例.

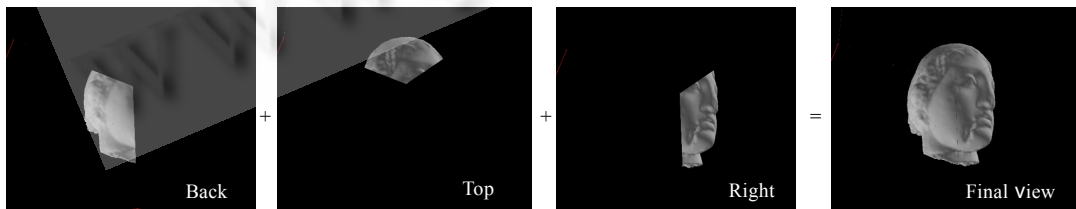


Fig.6 Virtual plane mosaics for the Venus model by texture-mapping the 3 planes of its virtual bounding box

图 6 利用虚平面拼接技术在维纳斯头像的包围盒后、顶、右 3 个内表面进行纹理映射后的成像示意图

为了使视点能够在深度图像内部进行漫游,并且解决由于信息缺失而产生的空洞,本文算法的处理是描述场景的源深度图像中的像素,按照三维变换的思想首先映射到包围盒的各个表面.例如,当视点位于图 7 中所示

位置时,可以将源深度图像中的像素依次变换到 front,back,top,right,bottom,left 这 6 个表面,然后按照前变换公式,在包围盒的各个表面上分别进行一次前变换,这样就生成了基于当前视点的中间图像,由此需要做 6 次“前变换”。显然,这些变换增加了成像时间.但是可以发现,根据图像前变换的特点,在包围盒表面生成的中间图像独立于视线方向和视角大小,因此,如果完成了对于包围盒所有内表面的前变换,我们就可以获得当前视点在任意方向和视角下的成像效果,即得到该视点的全景图成像,如图 7 所示.

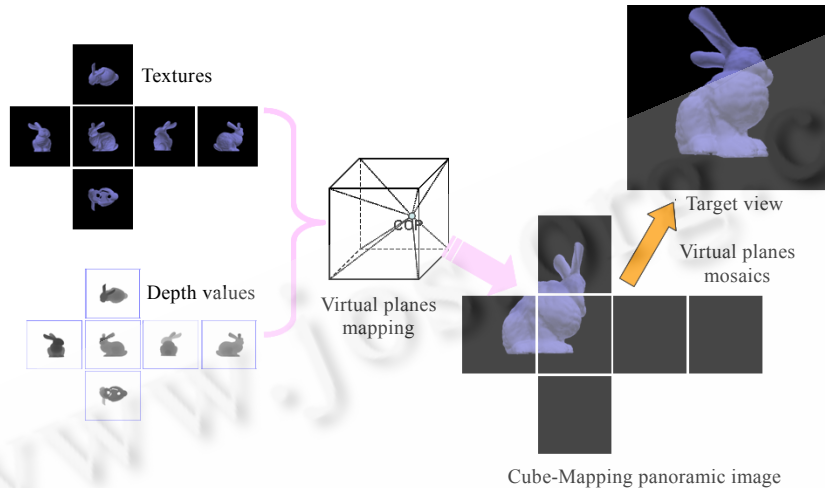


Fig.7 Construction of the panoramic image in depth image space by using virtual plane mapping

图 7 利用虚平面映射与拼接技术实现深度图像空间中全景图的生成

为了提高对深度图像处理的效率,这里,虚平面映射的优化策略是,根据当前视点的位置和方向,首先判断场景包围盒的各个内表面的可见性.这样,绘制时只需对这些可见内表面对应的浮雕纹理进行前变换,完成它们从内表面到成像面的纹理映射,就可以获得完整的目标图像.

由本文中的全景图生成算法可以进一步获得场景中的任意视点的全景图,从而实现视点在深度图像空间内部的任意漫游.而这种方式获得的成像具有较为合理的水平视差和竖直视差,其效果明显好于此前的其他基于图像的全景图生成方法.

4 实验结果与讨论

本文算法的效率与图形硬件的纹理加速功能、最终目标图像的分辨率以及源深度图像的分辨率和深度层次相关.本文方法的实验平台是运行 Windows XP 操作系统的微机,配有 Pentium IV 2.5G CPU,1G DDR 内存以及 nVidia GeForce 6800 图形卡.我们在将源深度图像与目标图像的分辨率设置为 256×256 的情况下,采用多种模型及场景进行了测试,其成像结果如图 8~图 10 所示.在表 1 里,我们给出了虚平面映射算法在预处理时间和外存储空间代价上的结果.

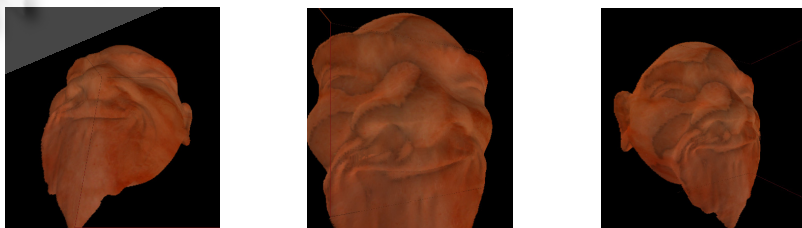


Fig.8 Rendering results of head model by virtual plane mapping and mosaics from different views

图 8 利用虚平面映射技术从不同视角对场景进行绘制(实验模型:老人头像)

