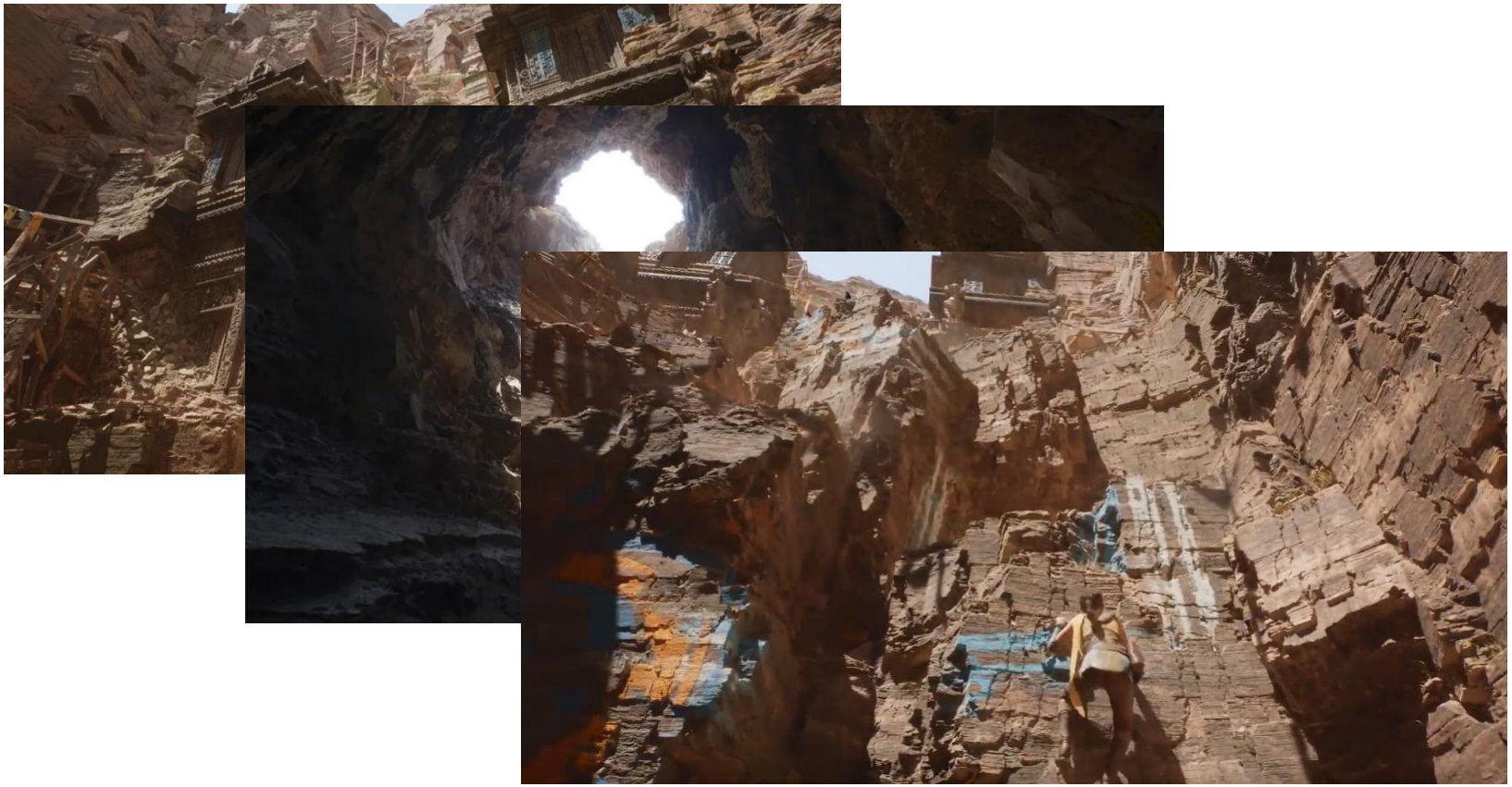
计算机图形学基础

胡事民

清华大学计算机科学与技术系

从Epic Game的虚幻引擎5说起

- 让实时渲染细节能够媲美电影CG和真实世界



网格参数化与几何图像技术

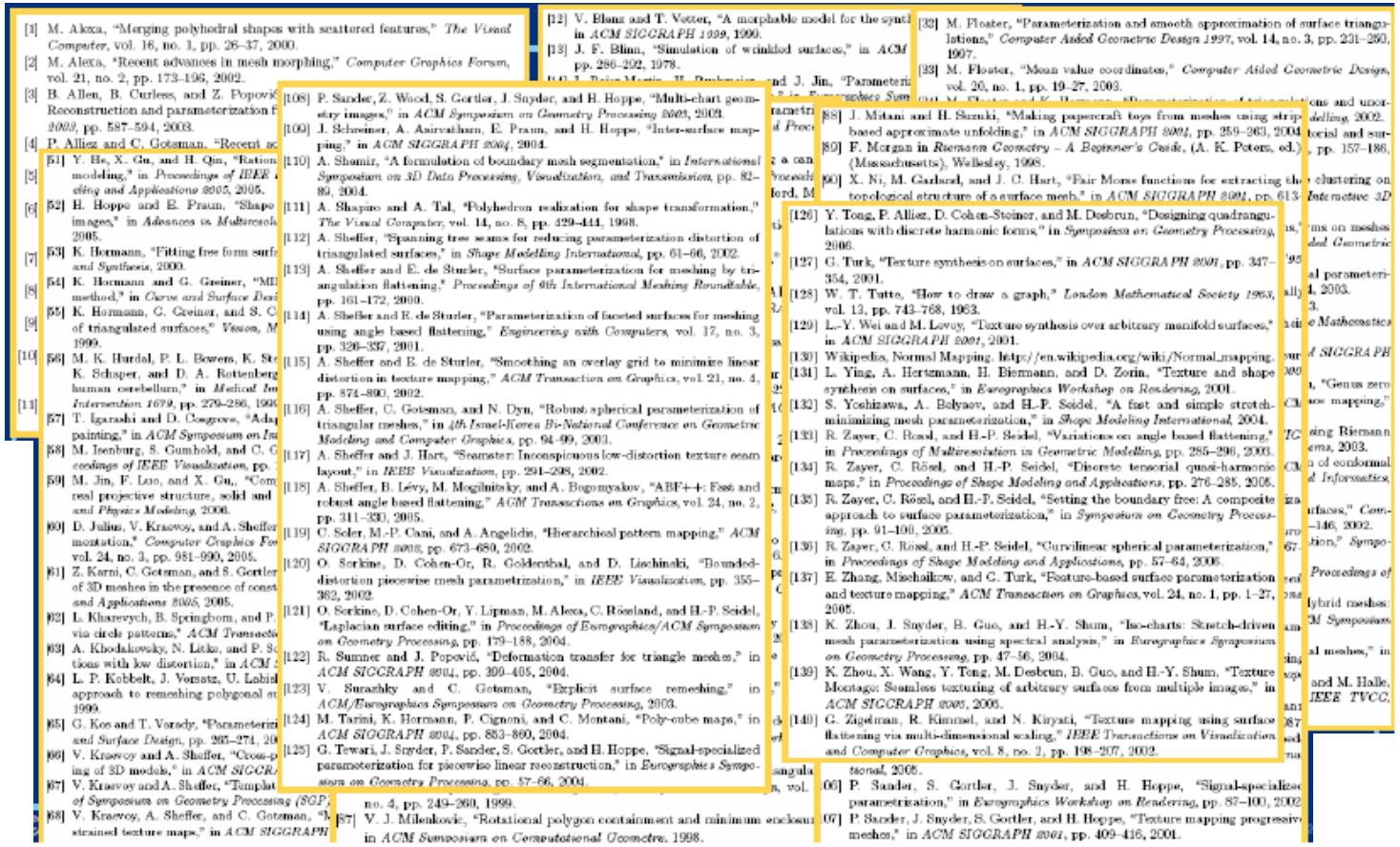
- 网格参数化 (Parametrization) 技术
  - 多面体参数化
  - 球面参数化
  - 平面参数化
- 复杂拓扑下的网格切割路径选取算法
- 几何图像 (Geometry Image) 技术

网格参数化的意义和背景

- 在图形学研究中，曲面模型的描述方法有很多，包括隐式曲面、参数曲面、细分曲面、多边形网格以及点云等等
- 网格曲面模型由于具有灵活高效的特点，在工业界中被广泛应用
- 网格参数化关注于：
  - 如何在网格曲面和给定的参数域之间建立一个映射关系
- 网格参数化是图形学中一个重要的内容

网格参数化的意义和背景

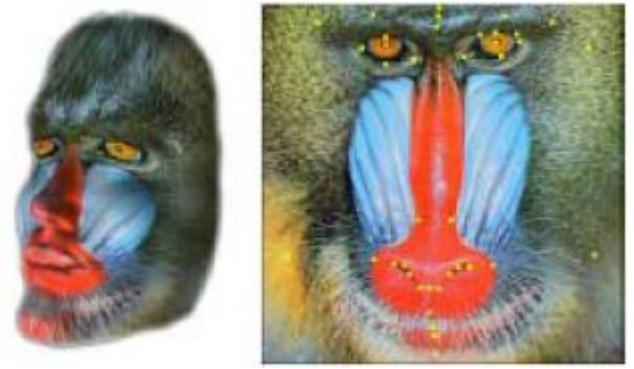
- 大量与参数化相关的前沿研究



[1] M. Alexa, "Merging polyhedral shapes with scattered features," *The Visual Computer*, vol. 16, no. 1, pp. 26–37, 2000.
[2] M. Alexa, "Recent advances in mesh morphing," *Computer Graphics Forum*, vol. 21, no. 2, pp. 173–196, 2002.
[3] B. Allen, B. Curless, and Z. Popović, Reconstruction and parameterization f... *2003*, pp. 587–594, 2003.
[4] P. Alliez and C. Gotsman, "Recent ac...
[5] Y. He, X. Gu, and H. Qin, "Ration... modeling," in *Proceedings of IEEE ... eling and Applications 2005*, 2005.
[6] H. Hoppe and E. Praun, "Shape ... images," in *Advances in Multiresol...* 2005.
[7] K. Hormann, "Fitting free form surf... *and Synthesis*, 2000.
[8] K. Hormann and G. Greiner, "MI... method," in *Curve and Surface Desi...*
[9] K. Hormann, G. Greiner, and S. C... of triangulated surfaces," *Vision, M...* 1999.
[10] M. K. Hurdal, P. L. Bowers, K. St... K. Schaper, and D. A. Rottenberg ... human cerebellum," in *Medical Im... Intervention 1679*, pp. 279–286, 199...
[11] T. Igarashi and D. Cosgrove, "Ada... painting," in *ACM Symposium on In...*
[12] M. Isenburg, S. Gumhold, and C. G... *coding of IEEE Visualization*, pp. ...
[13] M. Jin, F. Luo, and X. Gu, "Com... real projective structure, solid and ... *and Physics Modeling*, 2006.
[14] D. Julius, V. Kraevoy, and A. Sheffer ... mentation," *Computer Graphics For...* vol. 24, no. 3, pp. 981–990, 2005.
[15] Z. Karni, C. Gotsman, and S. Gortler ... of 3D meshes in the presence of const... *and Applications 2005*, 2005.
[16] L. Kharevych, B. Springborn, and P. ... via circle patterns," *ACM Transacti...*
[17] A. Khodakovsky, N. Litke, and P. Sc... tions with low distortion," in *ACM ...*
[18] L. P. Kobbelt, J. Vorsatz, U. Labik ... approach to remeshing polygonal su... 1999.
[19] G. Kos and T. Varady, "Parameteri... *and Surface Design*, pp. 265–274, 20...
[20] V. Kraevoy and A. Sheffer, "Cross-p... ing of 3D models," in *ACM SIGGRA...*
[21] V. Kraevoy and A. Sheffer, "Templat... *of Symposium on Geometry Processing (SGP)...*
[22] V. Kraevoy, A. Sheffer, and C. Gotsman, "... strained texture maps," in *ACM SIGGRAPH ...*
[23] P. Sander, Z. Wood, S. Gortler, J. Snyder, and H. Hoppe, "Multi-chart geometry images," in *ACM Symposium on Geometry Processing 2003*, 2003.
[24] J. Schreiner, A. Asirvatham, E. Praun, and H. Hoppe, "Inter-surface mapping," in *ACM SIGGRAPH 2004*, 2004.
[25] A. Shamir, "A formulation of boundary mesh segmentation," in *International Symposium on 3D Data Processing, Visualization, and Transmission*, pp. 82–89, 2004.
[26] A. Shapiro and A. Tal, "Polyhedron realization for shape transformation," *The Visual Computer*, vol. 14, no. 8, pp. 429–444, 1998.
[27] A. Sheffer, "Spanning tree seams for reducing parameterization distortion of triangulated surfaces," in *Shape Modeling International*, pp. 61–66, 2002.
[28] A. Sheffer and E. de Sturler, "Surface parameterization for meshing by triangulation flattening," *Proceedings of 9th International Meshing Roundtable*, pp. 161–172, 2000.
[29] A. Sheffer and E. de Sturler, "Parameterization of faceted surfaces for meshing using angle based flattening," *Engineering with Computers*, vol. 17, no. 3, pp. 326–337, 2001.
[30] A. Sheffer and E. de Sturler, "Smoothing an overlay grid to minimize linear distortion in texture mapping," *ACM Transaction on Graphics*, vol. 21, no. 4, pp. 874–890, 2002.
[31] A. Sheffer, C. Gotsman, and N. Dyn, "Robust spherical parameterization of triangular meshes," in *4th Israel-Korea Bi-National Conference on Geometric Modeling and Computer Graphics*, pp. 94–99, 2003.
[32] A. Sheffer and J. Hart, "Seamster: Inconspicuous low-distortion texture seam layout," in *IEEE Visualization*, pp. 291–298, 2002.
[33] A. Sheffer, B. Lévy, M. Mogilnitsky, and A. Bogomyakov, "ABF++: Fast and robust angle based flattening," *ACM Transactions on Graphics*, vol. 24, no. 2, pp. 311–330, 2005.
[34] C. Soler, M.-P. Cani, and A. Angelidis, "Hierarchical pattern mapping," *ACM SIGGRAPH 2002*, pp. 673–680, 2002.
[35] O. Sorkine, D. Cohen-Or, R. Goldenthal, and D. Lischinski, "Bounded-distortion piecewise mesh parameterization," in *IEEE Visualization*, pp. 355–362, 2002.
[36] O. Sorkine, D. Cohen-Or, Y. Lipman, M. Alexa, C. Rössland, and H.-P. Seidel, "Laplacian surface editing," in *Proceedings of Eurographics/ACM Symposium on Geometry Processing*, pp. 179–188, 2004.
[37] R. Sumner and J. Popović, "Deformation transfer for triangle meshes," in *ACM SIGGRAPH 2004*, pp. 399–405, 2004.
[38] V. Surazhsky and C. Gotsman, "Explicit surface remeshing," in *ACM/Eurographics Symposium on Geometry Processing*, 2003.
[39] M. Tarini, K. Hormann, P. Cignoni, and C. Montani, "Poly-cube maps," in *ACM SIGGRAPH 2004*, pp. 853–860, 2004.
[40] G. Tewari, J. Snyder, P. Sander, S. Gortler, and H. Hoppe, "Signal-specialized parameterization for piecewise linear reconstruction," in *Eurographics Symposium on Geometry Processing*, pp. 57–66, 2004.
[41] V. J. Milenkovic, "Rotational polygon containment and minimum enclosure," in *ACM Symposium on Computational Geometry*, 1998.
[42] V. Blanz and T. Vetter, "A morphable model for the synthesis of 3D faces," in *ACM SIGGRAPH 1999*, 1999.
[43] J. F. Blinn, "Simulation of wrinkled surfaces," in *ACM* ... pp. 286–292, 1978.
[44] J. Popović and J. Jin, "Parameteri...
[45] J. Mitani and H. Suzuki, "Making papercraft toys from meshes using strip-based approximate unfolding," in *ACM SIGGRAPH 2004*, pp. 259–263, 2004.
[46] F. Morgan in *Riemann Geometry – A Beginner's Guide*, (A. K. Peters, ed.) (Massachusetts), Wellesley, 1998.
[47] X. Ni, M. Garland, and J. C. Hart, "Fair Morse functions for extracting the topological structure of a surface mesh," in *ACM SIGGRAPH 2004*, pp. 613–...
[48] Y. Tong, P. Alliez, D. Cohen-Steiner, and M. Desbrun, "Designing quadrangulations with discrete harmonic forms," in *Symposium on Geometry Processing*, 2006.
[49] G. Turk, "Texture synthesis on surfaces," in *ACM SIGGRAPH 2001*, pp. 347–354, 2001.
[50] W. T. Tutte, "How to draw a graph," *London Mathematical Society 1963*, vol. 13, pp. 743–768, 1963.
[51] L.-Y. Wei and M. Levoy, "Texture synthesis over arbitrary manifold surfaces," in *ACM SIGGRAPH 2001*, 2001.
[52] Wikipedia, Normal Mapping. http://en.wikipedia.org/wiki/Normal_mapping.
[53] L. Ying, A. Hertzmann, H. Biermann, and D. Zorin, "Texture and shape synthesis on surfaces," in *Eurographics Workshop on Rendering*, 2001.
[54] S. Yoshizawa, A. Belyaev, and H.-P. Seidel, "A fast and simple stretch-minimizing mesh parameterization," in *Shape Modeling International*, 2004.
[55] R. Zayer, C. Rossl, and H.-P. Seidel, "Variations on angle based flattening," in *Proceedings of Multiresolution in Geometric Modelling*, pp. 295–296, 2003.
[56] R. Zayer, C. Rössl, and H.-P. Seidel, "Discrete tensorial quasi-harmonic maps," in *Proceedings of Shape Modeling and Applications*, pp. 276–285, 2005.
[57] R. Zayer, C. Rössl, and H.-P. Seidel, "Setting the boundary free: A composite approach to surface parameterization," in *Symposium on Geometry Processing*, pp. 91–100, 2005.
[58] R. Zayer, C. Rössl, and H.-P. Seidel, "Curvilinear spherical parameterization," in *Proceedings of Shape Modeling and Applications*, pp. 57–64, 2006.
[59] E. Zhang, Mischaikow, and G. Turk, "Feature-based surface parameterization and texture mapping," *ACM Transaction on Graphics*, vol. 24, no. 1, pp. 1–27, 2005.
[60] K. Zhou, J. Snyder, B. Guo, and H.-Y. Shum, "Iso-charts: Stretch-driven mesh parameterization using spectral analysis," in *Eurographics Symposium on Geometry Processing*, pp. 47–56, 2004.
[61] K. Zhou, X. Wang, Y. Tong, M. Desbrun, B. Guo, and H.-Y. Shum, "Texture Montage: Seamless texturing of arbitrary surfaces from multiple images," in *ACM SIGGRAPH 2005*, 2005.
[62] G. Zigelman, R. Kimmel, and N. Kiryati, "Texture mapping using surface flattening via multi-dimensional scaling," *IEEE Transactions on Visualization and Computer Graphics*, vol. 8, no. 2, pp. 198–207, 2002.
[63] P. Sander, S. Gortler, J. Snyder, and H. Hoppe, "Signal-specialized parameterization," in *Eurographics Workshop on Rendering*, pp. 87–100, 2002.
[64] P. Sander, J. Snyder, S. Gortler, and H. Hoppe, "Texture mapping progressive meshes," in *ACM SIGGRAPH 2001*, pp. 409–416, 2001.

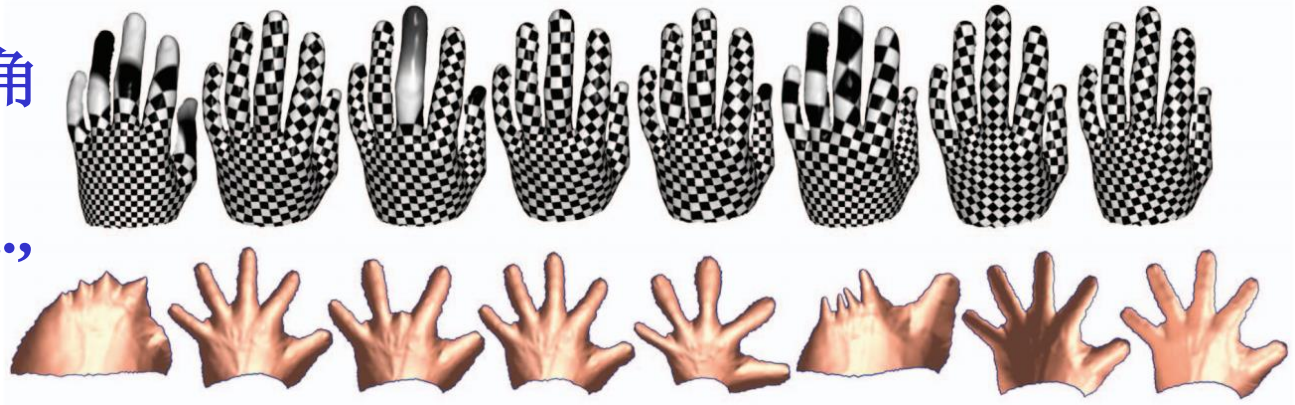
网格参数化的意义和背景

- 网格参数化在图形学中的应用
  - 纹理映射



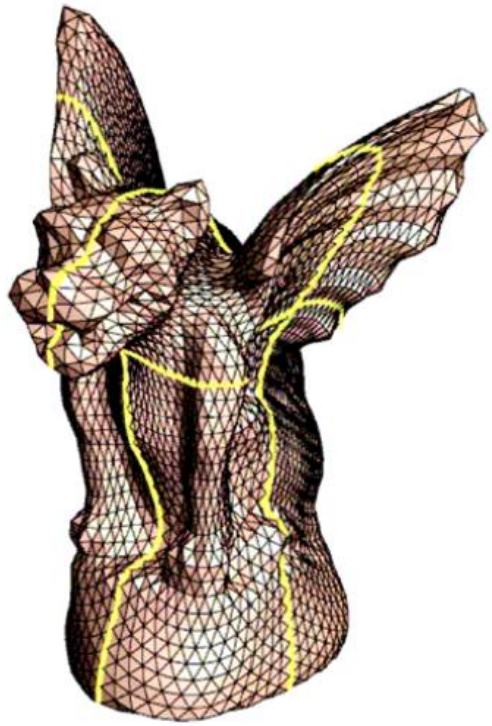
纹理映射需要尽量保角和均匀的平面参数化

Yong-Liang Yang et al., IEEE TVCG 2008



网格参数化的意义和背景

- 网格参数化在图形学中的应用
  - 纹理映射
  - 网格重剖

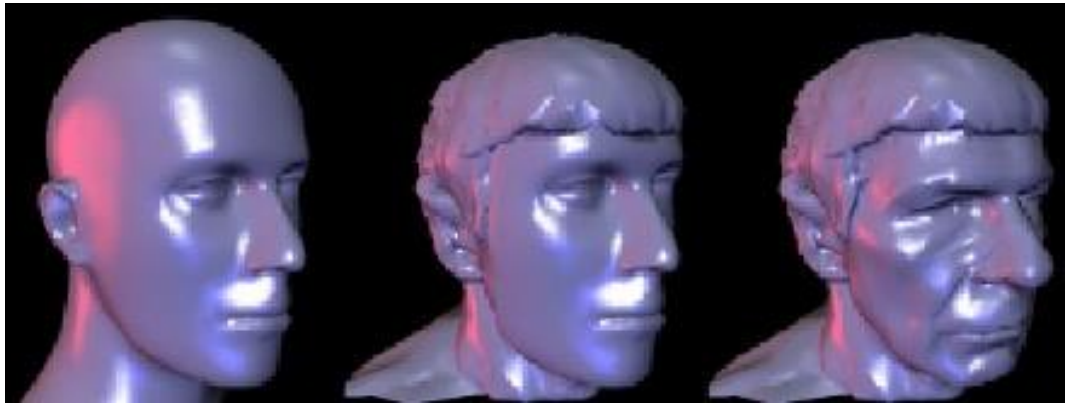


Spherical Parameterization and Remeshing

E. Praun and H. Hoppe, SIGGRAPH 2003

网格参数化的意义和背景

- 网格参数化在图形学中的应用
  - 纹理映射
  - 网格重剖
  - 几何变形



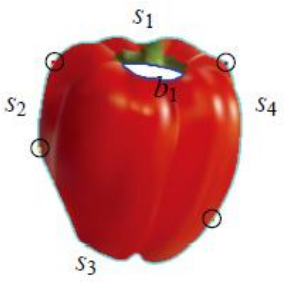
Multi-resolution Mesh Morphing

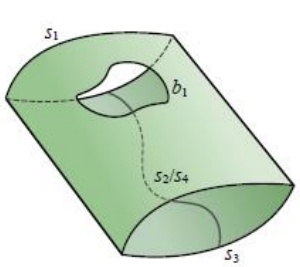
A. Lee, D. Dobkin, W. Sweldens, and P. Schroder, SIGGRAPH 1999

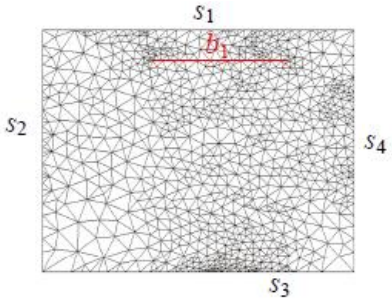
网格参数化的意义和背景

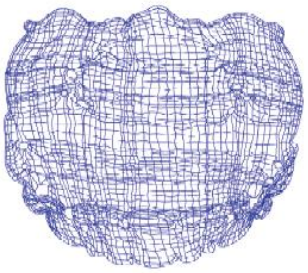
- 网格参数化在图形学中的应用
  - 纹理映射
  - 网格重剖
  - 几何变形
  - 图像矢量化

（见下页详述）







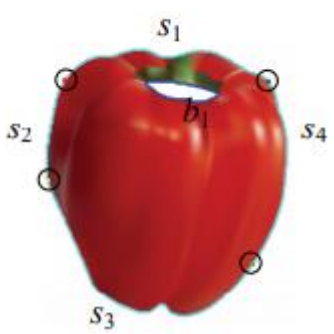


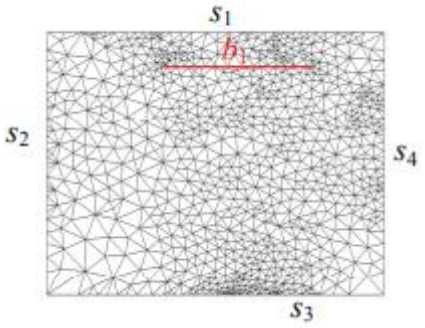
Topology-Preserving Gradient Mesh

Yu-Kun Lai et al., SIGGRAPH 2009

网格参数化的意义和背景

- 图像矢量化(SIGGRAPH 2009)
  - 将一张图像转化为保持拓扑结构的梯度网格
  - 什么是梯度网格？
    - 每个网格面片由四个顶点构成
    - 在每个顶点存储位置、颜色、网格参数uv的导数
  - 梯度网格的性质：
    - 几何保持G1连续性
    - 颜色保持C1连续性



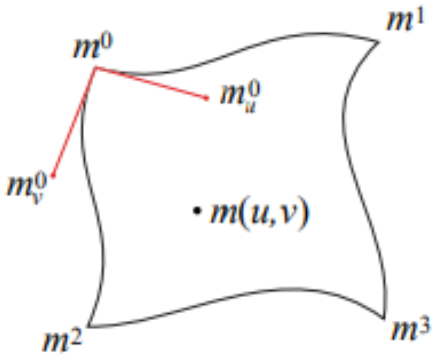


网格参数化的意义和背景

- 图像矢量化(SIGGRAPH 2009)
  - 参数方程：

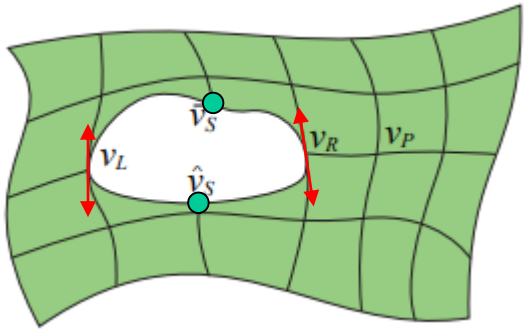
$$F(u,v) = UCQ^f C^T V^T,$$

$$Q^f = \begin{bmatrix} m^0 & m^2 & m_v^0 & m_v^2 \\ m^1 & m^3 & m_v^1 & m_v^3 \\ m_u^0 & m_u^2 & m_{uv}^0 & m_{uv}^2 \\ m_u^1 & m_u^3 & m_{uv}^1 & m_{uv}^3 \end{bmatrix} \quad C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -3 & 3 & -2 & -1 \\ 2 & -2 & 1 & 1 \end{bmatrix}$$



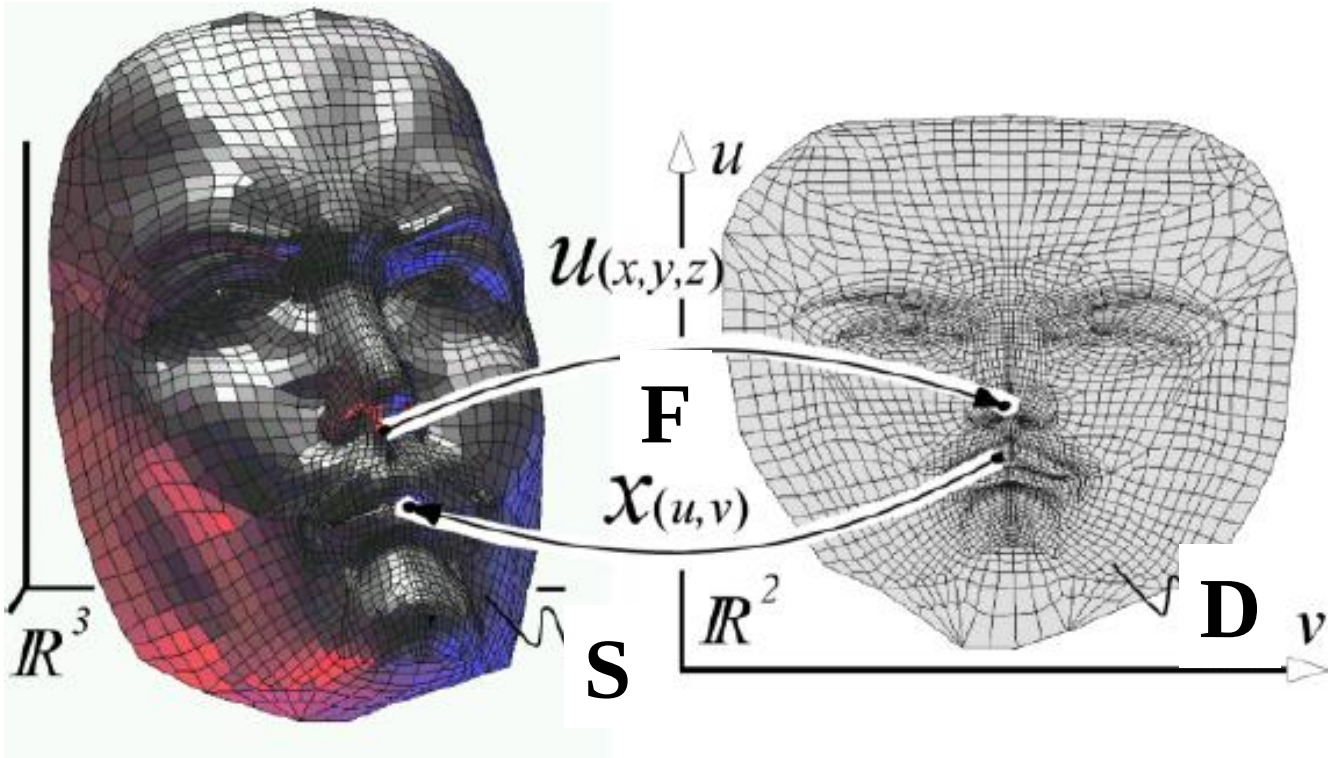
$$U = \begin{pmatrix} 1 & u & u^2 & u^3 \end{pmatrix}$$
$$V = \begin{pmatrix} 1 & v & v^2 & v^3 \end{pmatrix}$$

  - 如何表示有洞的网格 (slit map)：
    - 把一个点拆成两个(v_s)
    - 让关联点v_l, v_r的偏导互为相反数



网格参数化的数学描述

- 给定R^3中的网格曲面S以及一个参数域D，求解一个映射关系F（通常要求是一一映射）：



网格参数化的分类

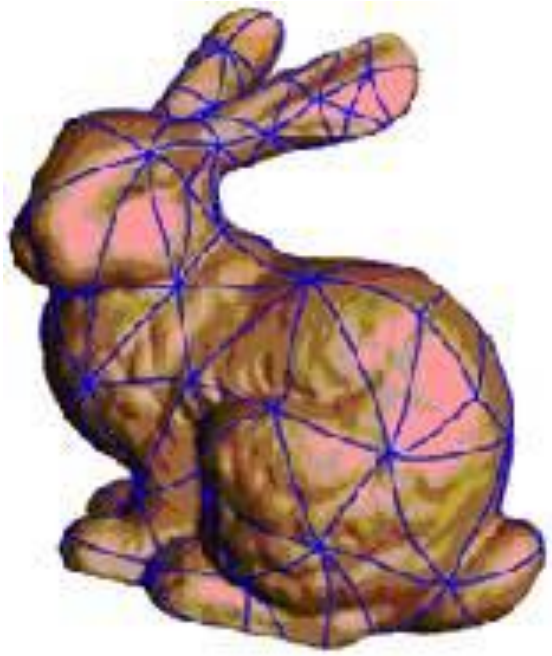
- 根据参数域的不同，网格参数化方法可以分为三类：
  - 网格的多面体参数化
  - 网格的球面参数化
  - 网格的平面参数化

多面体参数化

- 多面体 (又称基网格) 参数化最早是由 Eck 提出的，方法是将一张复杂的网格映射到一个与它同拓扑的相对简单的网格上，这个简单的网格就是所谓的基网格
- 用基网格做为参数域实际上就是将原网格曲面分割成很多面片 (Patch)，每一个面片对应着基网格上的一个面 (Face)

多面体参数化

- 网格面片 (Patch) 与基网格上的多边形面 (Face) 的对应关系如下图：



多面体参数化 – 步骤

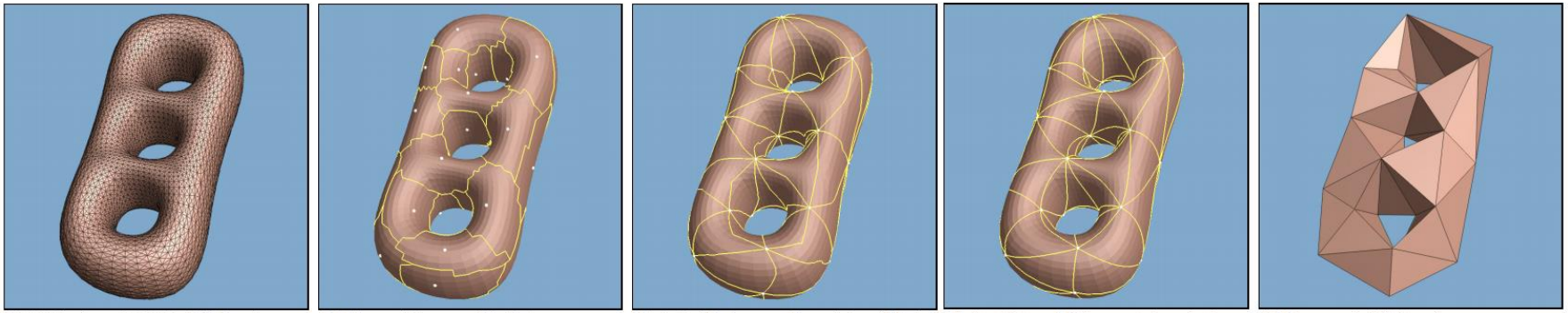
- 多面体参数化方法主要包含两个步骤：
  - 建立一个与原曲面同拓扑的基网格，可以通过将原网格进行分割或者对原网格进行化简来实现；
  - 在原网格和基网格之间建立映射关系，具体方法，可以参考后面的平面参数化方法

多面体参数化 – 基网格生成

- 在建立基网格这一步，**Eck** 等人使用对原网格进行分割的方法；而 **Lee** 等人使用对原网格进行简化的方法：
  - **Multi-resolution Analysis of Arbitrary Meshes, M. Eck, Tony Derose et al., SIGGRAPH 1995 (2028 citations)**
  - **MAPS: multi-resolution adaptive parameterization of surfaces, Aaron W. F. Lee et al., SIGGRAPH 1998 (976 citations)**

多面体参数化 – 基网格生成

- **Eck等人的基网格生成步骤：**
  - **首先，在网格上均匀地选取一组初始顶点；**
  - **然后，借助Voronoi图进行Delaunay三角化。**
  - **最后，将初始Delaunay结果中的边拉直。**



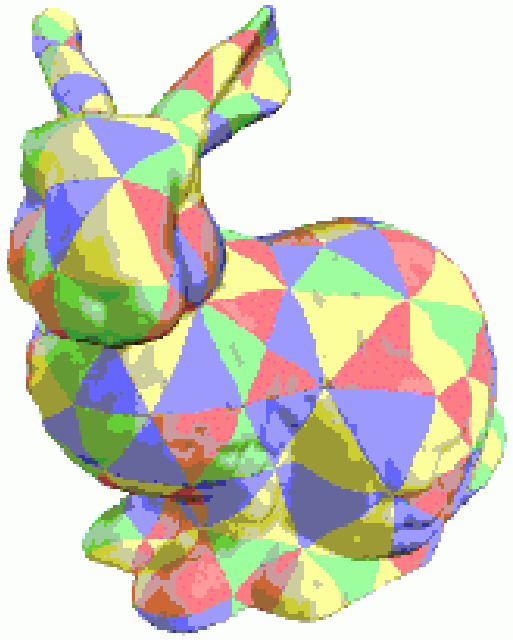
(a) Original mesh *M* (11,776 faces) (b) Voronoi diagram (31 tiles) (c) Initial Delaunay triangulation (70 tri.) (d) Straightened Delaunay triangulation (e) Base mesh (70 faces)

多面体参数化

- 由于基网格与原网格的拓扑一致而且基本保持了原网格的形状，因此该参数化方法最大的优点就是变形小，因此它的应用范围也最广，如多分辨率分析、网格重剖、变形等等
- 但是，由于该参数化方法是一种局部参数化方法，也就是说基网格上不同的面对应着不同的参数域，这样在参数域的交界容易产生问题

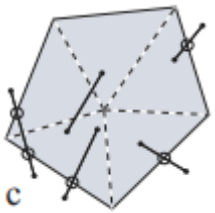
多面体参数化

- **2003年，Khodakovsky 给出了一个新的方法，在不固定参数域边界的情况下，对整个曲面的参数化进行光滑处理，进一步增强了这种参数化模型的效果。**

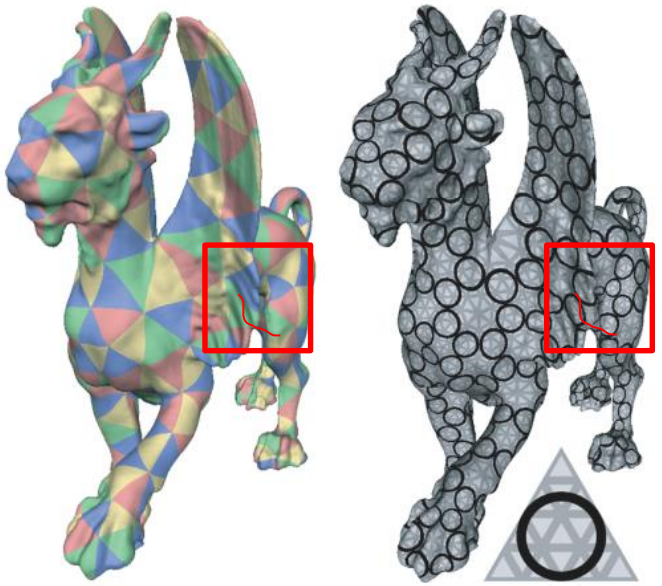


多面体参数化

- **Khodakovsky等人认为，保证全局光滑性的关键是：等值线在经过不同Patches时，它在交界处的切向量应该保持连续。**



Patches



纹理贴图示例

Globally Smooth Parameterizations With Low Distortion
A. Khodakovsky, N. Litke, P. Schroder @ CalTech
SIGGRAPH 2003 （256 citations）

球面参数化

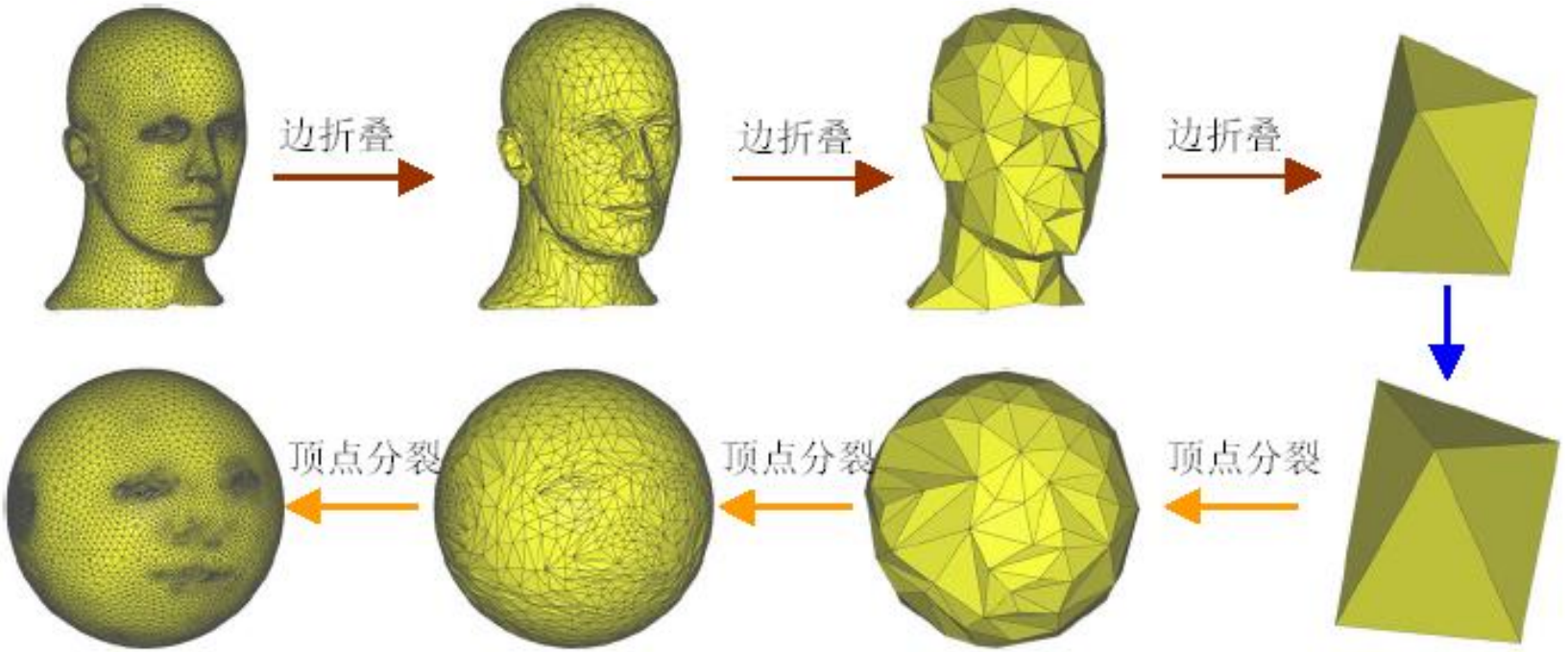
- 球面参数化比较容易理解，尤其是对于亏格为零（没有洞）的封闭的网格，我们可以将网格的每条边想象成一根皮筋，那么如果把它套在一个塑料球外面，由于弹力作用它必然会将球包住，这样也就自然地建立了一个原网格到球面的映射关系
- 但是由于球面参数化缺少一个明确的应用背景，研究工作相对较少

球面参数化

- 周昆和鲍虎军等人给出一个球面参数化方法，思路是:
  - 先通过边折叠操作将网格化简成一个凸多面体，然后用中心投影对这个凸多面体建立一个球面映射，从这个初始映射出发，通过点分裂操作将凸多面体还原成原始网格，每还原一个点都计算出该点的球面参数，最终生成原网格的球面映射
  - *Kun Zhou et al., 3D Surface Filtering Using Spherical Harmonics, CAD, 2004. (93 citations)*

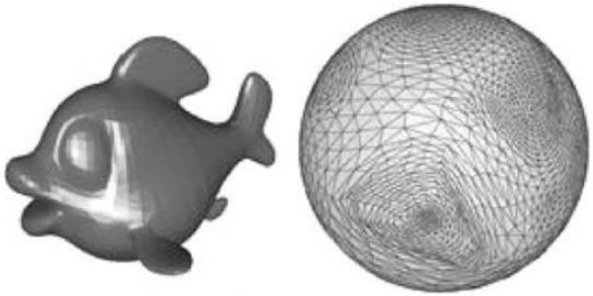
球面参数化

- 球面参数化算法流程：

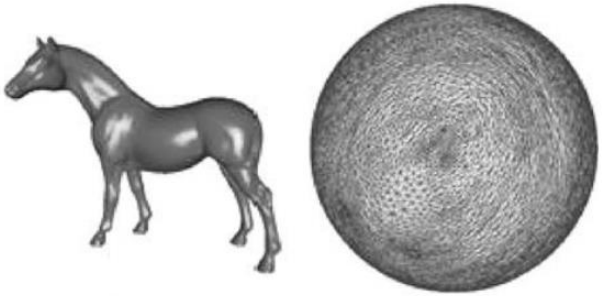


球面参数化

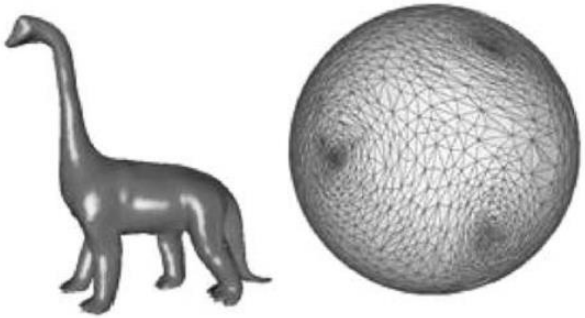
- 一些例子：



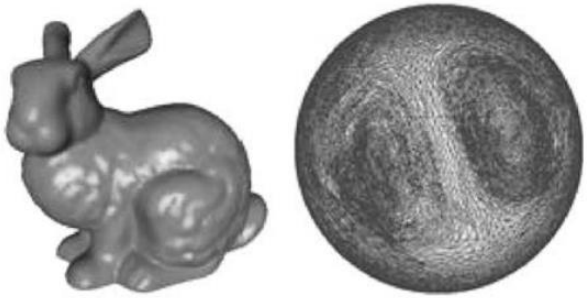
(a) Fish and its parameterization.



(b) Horse and its parameterization.



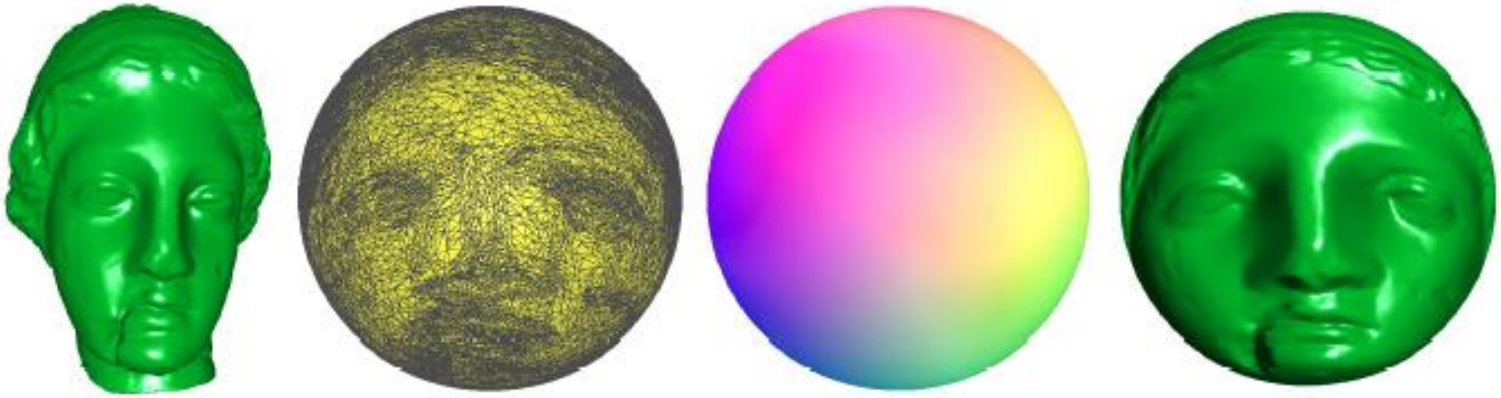
(c) Dino and its parameterization.



(d) Bunny and its parameterization.

球面参数化

- 使用球面参数化，可以将零亏格封闭网格模型的几何坐标信息以及法向量信息分别使用球面信号函数来进行表达：



(a)原始模型　(b)球面映射　(c)位置坐标信号　(d)法向量信号

图 2.3 VenusHead 模型球面参数化及球面信号

球面参数化

- 在参数化的理论方面，一个核心问题是：如何保证参数化的网格不产生翻转（**overlap**），或者说，保证是一个一一映射
- **C. Gotsman, X. Gu,** 和 **A. Sheffer** 提出了一个球面参数化的理论框架，同时给出了一个保证参数化不产生翻转(non-overlapping)的解空间

Fundamentals of Spherical Parameterization for 3D Meshes

C. Gotsman, X. Gu, and A. Sheffer, SIGGRAPH 2003 (390 cites)

球面参数化

- 球面参数化的应用：网格重剖

Spherical Parameterization and Remeshing

E. Praun and H. Hoppe, SIGGRAPH 2003 (597 cites)

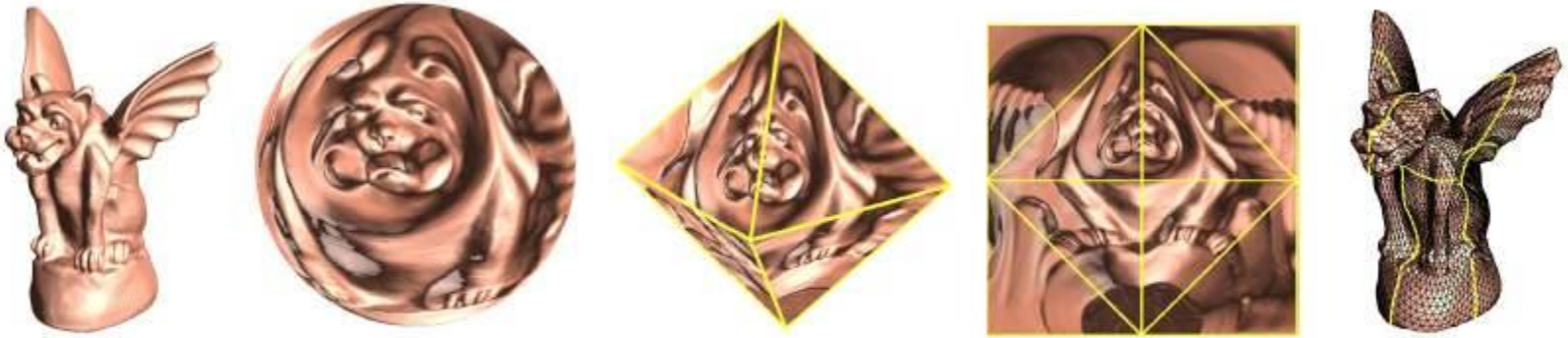


Figure 1: Demonstration of spherical parametrization and subsequent resampling into a geometry image.

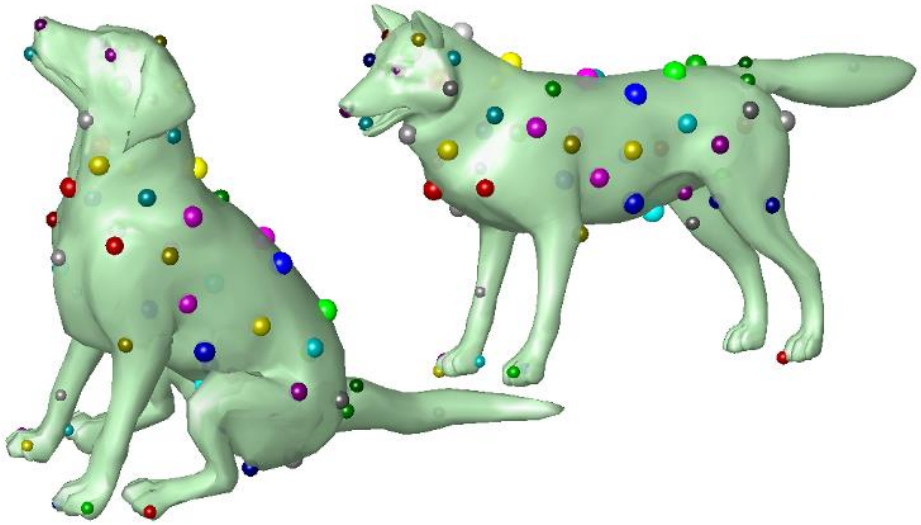
通过球面参数化获得规整的网格剖分

球面参数化

- 球面参数化的应用：形状分析

  Mobius Voting for surface correspondence

  Yaron Lipman et al., SIGGRAPH 2009 (404 citations)



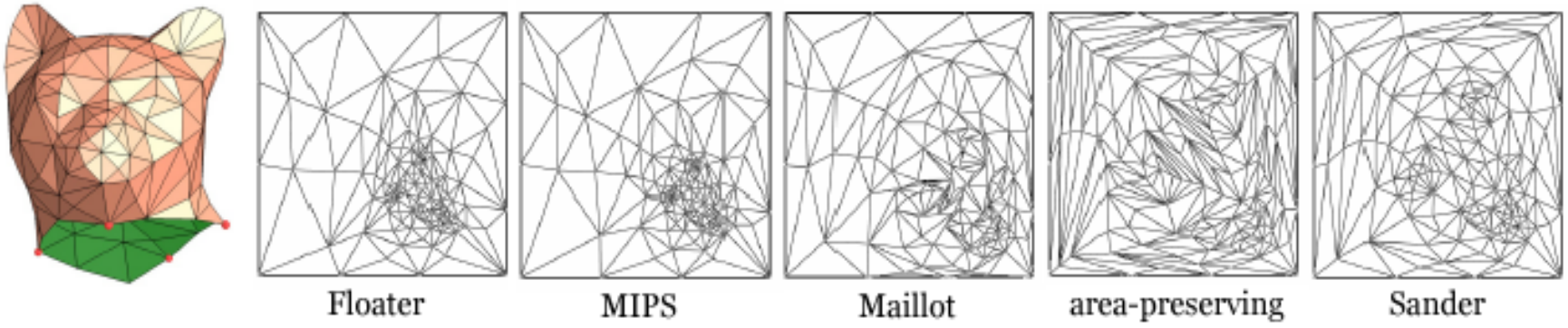
通过球面参数化，并投票搜索两个球面之间最佳的Mobius变换，获得两个原先网格之间的对应关系

球面参数化

- 球面参数化的缺点：对网格拓扑有限制
- 一般只适用于亏格为零的网格，如果亏格大于零，就必须将网格切开，而切割路径的选取也是一个很难的问题，无形中增加了球面参数化的复杂度，而且在后面我们将会看到，与其将切割后的曲面映射到球面上，还不如直接映射到平面

平面参数化

- 与参数曲面一样，我们同样希望将网格曲面映射到平面上一个有限的区域内，如果曲面与圆盘同拓扑且有界，我们自然可以很容易的将它“拍”到平面上，同时利用各种不同的方法减小参数化所产生的变形，如下图



平面参数化

- 对于与圆盘同拓扑的网格曲面，进行平面参数化的方法有许多，包括：
  - 基于顶点坐标表示的方法
  - 基于角度表示的方法 (ABF, Angle-Based Flattening)
  - 基于Circle Packing 的方法
- 所有这些方法都关注于如何让参数化前后的网格扭曲 (distortion) 尽量最小

平面参数化

- 基于顶点坐标表示的方法
  - 将每个顶点参数化后的平面 **UV** 坐标作为未知变量，通过求解一个能量优化问题，获得一个扭曲较小的参数化结果
  - 进行优化的能量往往代表了希望最小化的扭曲，包括角度上的扭曲 (代表了参数化在局部上是否保角) 和面积上扭曲 (代表了参数化在全局上是否均匀)

平面参数化

- 基于顶点坐标表示的方法包括：
  - **Coordinates-based 方法**

    Parameterization and smooth approximation of surface triangulations

    M. Floater, CAGD 1997
  - **LSCM 方法**

    Least Squares Conformal Maps for automatic texture atlas generation

    B. Levy, et al., SIGGRAPH 2002

（待续）

平面参数化

- 基于顶点坐标表示的方法包括：
  - **Discrete Conformal Parameterization（DCP）方法**

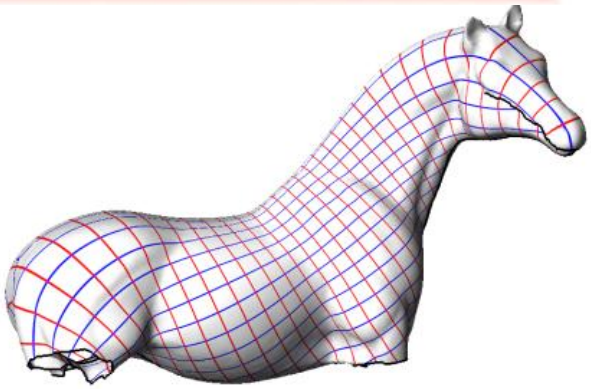
    Intrinsic parameterizations of surface meshes

    M. Desbrun, M. Meyer, and P. Alliez, CGF 2002
  - **MIPS 方法**

    MIPS: An efficient global parametrization method

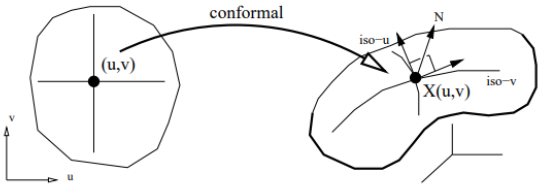
    K. Hormann and G. Greiner, Curve and Surface Design 2000

平面参数化



- 保形变换：Conformal Maps
  - 在保形变换中，等参线在交点处的切向量应该正交且等长

$$N(u,v)\times\frac{\partial X}{\partial u}(u,v)=\frac{\partial X}{\partial v}(u,v)$$



- **LSCM算法提出用复数来完成保形约束。对于每一个三角面片：**

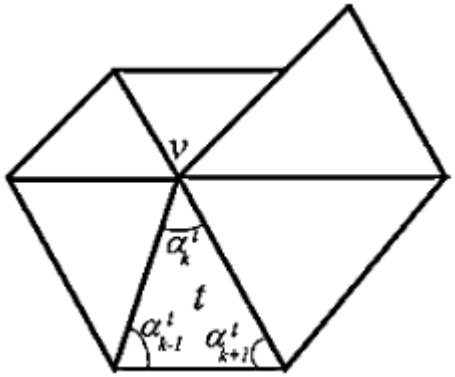
$\frac{\partial X}{\partial u}-\mathrm{i}\frac{\partial X}{\partial v}=0$ **，因此，X的逆映射U应满足** $\frac{\partial U}{\partial x}+\mathrm{i}\frac{\partial U}{\partial y}=0$

所以，最终的变换为了让所有三角面片都尽可能保形，应优化最小二乘函数：$C(T)=\int_T\left|\frac{\partial U}{\partial x}+i\frac{\partial U}{\partial y}\right|^2 dA$

平面参数化

- 基于角度表示的方法 (ABF)
  - 将三角形的内角作为未知变量，要参数化到平面上，这些变量需要满足限制条件：
  - 每个三角形内角和为 π
  - 每个顶点的一圈角度和为2π
  - $$\prod_{(t,k)\in v^*} \sin\alpha_{k\oplus 1}^t - \prod_{(t,k)\in v^*} \sin\alpha_{k\ominus 1}^t = 0$$

    (这条性质来自于正弦定理)



平面参数化

- 基于角度表示的方法 (ABF)
  - 基于角度表示的方法使用角度作为变量，与基于顶点坐标的方法类似，也需要优化扭曲能量
  - 不同的是，使用角度作为变量，描述角度扭曲的能量 (或者说，共形能量) 的形式将变得更简单

平面参数化

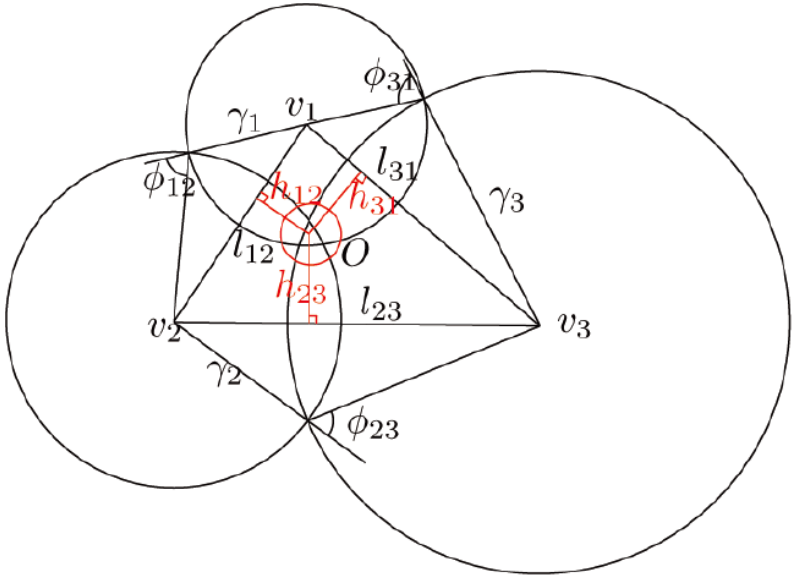
- 基于角度表示的方法 (ABF)
  - **Parameterization of faceted surfaces for meshing using angle based flattening**

    A. Sheffer and E. de Sturler, Engineering with Computers 2001
  - **ABF++: Fast and robust angle based flattening**

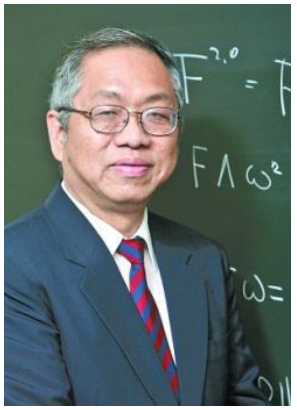
    A. Sheffer, B. L´evy, M. Mogilnitsky, and A. Bogomyakov, SIGGRAPH 2005

平面参数化

- 基于 Circle Packing 的方法
  - 在每个顶点做一个圆，将圆的半径作为变量，而保持相邻圆的交角不变
  - 因为能够完美保证共形，所以通常在 **Circle Packing** 框架下优化面积扭曲



平面参数化



- 基于 Circle Packing 的方法
  - Circle Packing (丘成桐、顾险峰)

    Global Conformal Surface Parameterization

    X . Gu and S.-T. Yau, SGP 2003, 127-137

    Optimal Surface Parameterization Using Inverse Curvature Map

    Yong-Liang Yang et al., TVCG 2008 (清华)
  - Circle Pattern

    Discrete conformal mappings via circle patterns

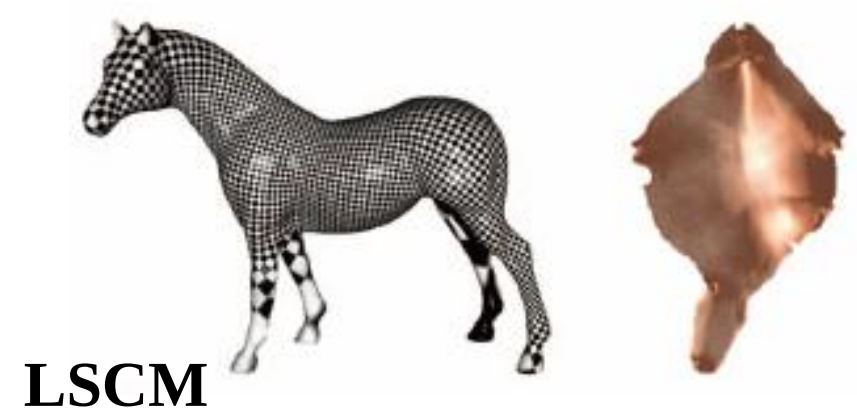
    L. Kharevych, B. Springborn, and P. Schroder,

    SIGGRAPH 2006

平面参数化

- 不同平面参数化方法的效果比较：

ABF++



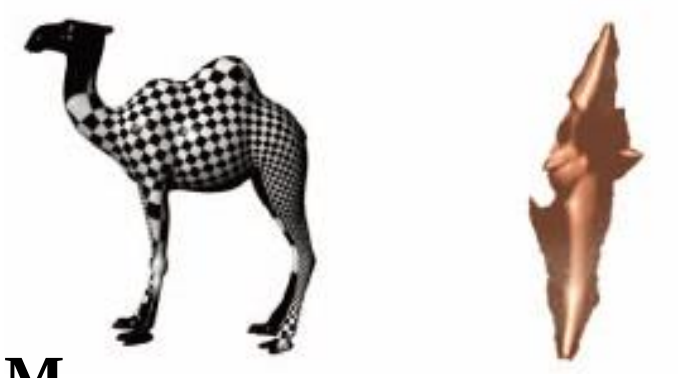
LSCM

Circle Pattern

平面参数化

- 不同平面参数化方法的效果比较：

ABF++



LSCM

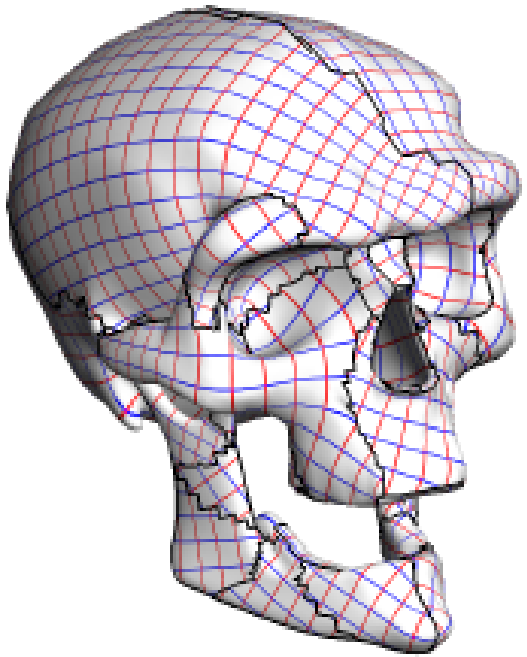
Circle Pattern

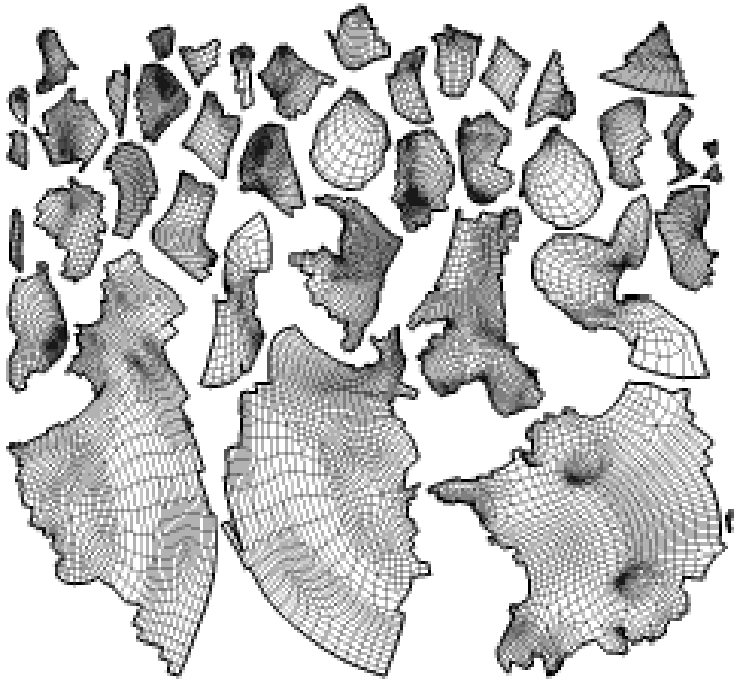
平面参数化

- 对于拓扑结构复杂的曲面，我们就必须将曲面切开，传统的方法是将曲面分割成若干与圆盘同拓扑的面片，再将每个面片映射成平面上的一个片 **(chart)**，最后利用图册 **(atlas)** 方法把这些片 合理的分布到一个指定的区域内，一般是一个正方形，这种方法主要应用在纹理映射中，而这个正方形区域就是纹理图 **(Texture Map)**

平面参数化

- 复杂拓扑网格曲面的平面参数化示例：





平面参数化

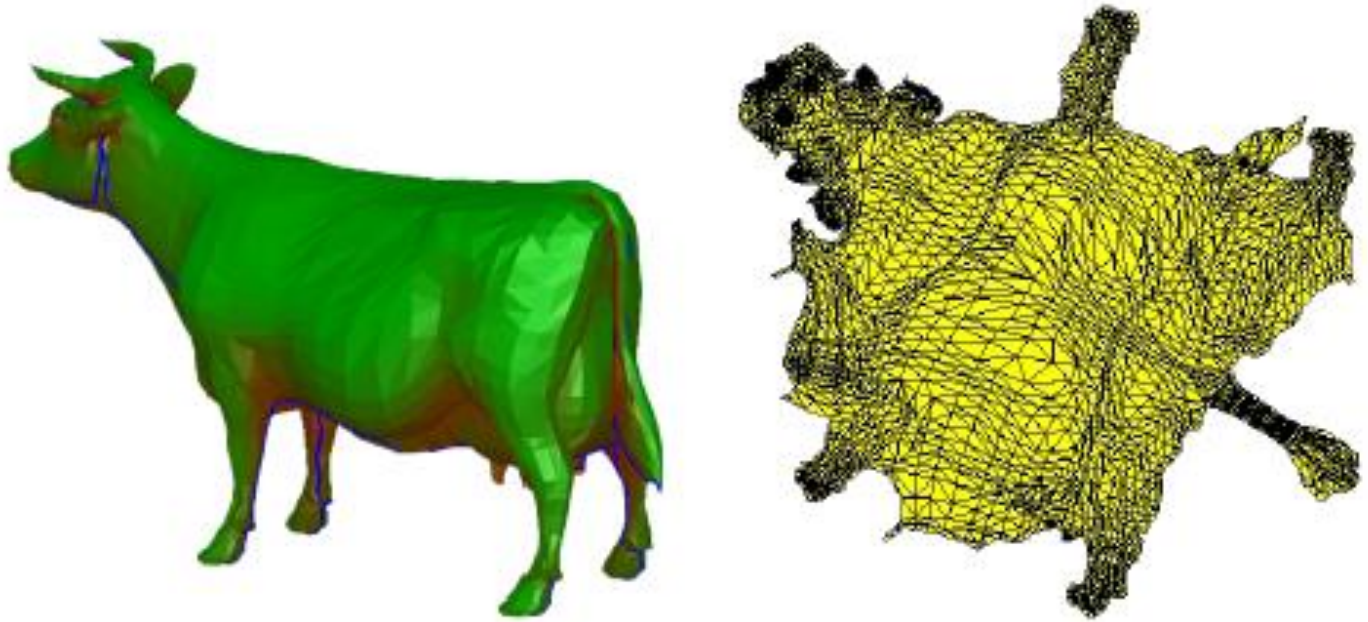
- 很明显，分片越多参数化产生的变形就越小，极端情况下，如果将网格上每个面都映射成一个片，变形就为零；相反分片越少，变形就越大，而我们的目的就是将整个曲面映射成一片，同时尽量减小变形
- 理论上，任何一个连通的二维流形曲面都可以找到一条切割路径将曲面展开，使之与圆盘同拓扑，继而将它映射到平面上

平面参数化

- 不同的切割路径可能导致展开后的曲面具有不同的最小参数化扭曲，于是我们的主要问题变成如何选取一条合适的切割路径，使得展开后的曲面能够较小变形地参数化到平面上
- **A. Sheffer** 和 **X. Gu** 针对这个问题进行了研究，他们发现只要增加切割路径将曲面上的突起点剪开，就可以达到减小参数化变形的目的
- 我们下面介绍切割路径的选取算法

切割路径的选取算法

- 一个例子：将同胚于封闭球面的模型切开并参数化到平面上



切割路径的选取算法

- 对于一般拓扑的网格模型，切割路径的选取可以分为两步：
  - 切割点的计算：找出网格上所有突起位置作为切割点；
  - 切割路径优化：在网格上找出一条连接所有切割点的尽量短的路径 作为切割路径，如果网格有边界，切割路径还必须与边界相连

切割路径的选取算法

- 切割点的计算
  - **A. Sheffer** 通过计算网格上所有点的局部曲率来寻找切割点 **(**曲率大处作为切割点**)**
  - **X. Gu** 则利用 **Floater** 参数化保形的特点，首先将曲面参数化到一个单位圆内，再在参数域上找出变形最大最明显的一些地方作为切割点

切割路径的选取算法

- 切割路径优化
  - 目的是在网格上找出连接所有切割点以及边界的最短路径，作为切割路径
  - 如果我们把边界也视作一个点的话，这个最短路径问题实际上就是一个带权图 **(weighted graph)** 上的 **Steiner** 树

切割路径的选取算法

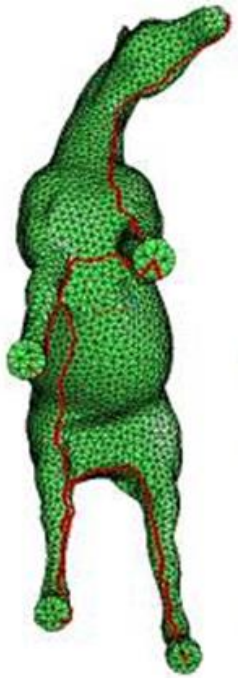
- 切割路径优化
  - **Steiner 树是指在一个带权图上连接若干给定点的最短树，我们将这些给定点称作端点(terminals)，包括所有切割点和边界**
  - **求解带权图的 Steiner 树是一个 NP 完全问题，只能近似求解**

切割路径的选取算法

- **A. Sheffer 用最短生成树来近似 Steiner 树，但是这样做不能获得足够好的切割路径，其结果常常会在一个切割点处引出两条割线**
- **例如右图中的例子，切割路径通过马腿的次数太多**

Spanning tree seams for reducing parameterization distortion of triangulated surfaces

A. Sheffer, SMI 2002 （63 cites）



切割路径的选取算法

- **X. Gu 采用增量法进行近似, 也就是每增加一个切割点，就把该点到当前切割路径的最短路径加入到当前切割路径中**
- **增量法结果的好坏主要依赖于切割点增加的顺序**

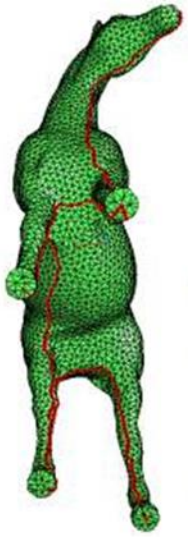
Geometry Images

X. Gu, S. Gortler, and H. Hoppe,

SIGGRAPH 2002 （1125 cites）

切割路径的选取算法

- 我们下面介绍 朱旭平 的方法，可以获得比前两种方法更好的结果：



A. Sheffer

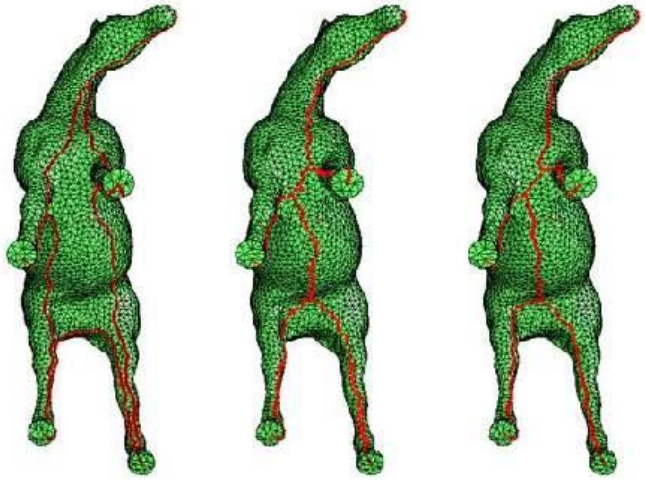


Fig. 4. Illlustration of each steps of seam computing

X. Zhu

Zhu X P, Hu S M, Martin R. Skeleton-based seam computation for triangulated surface parameterization[M]//Mathematics of Surfaces. Springer, Berlin, Heidelberg, 2003: 1-13.

切割路径的选取算法

- 算法思路：由于切割点恰恰都是在突起上，因此，我们应该让切割路径尽可能少地通过这些切割点，为了满足这个需要，我们给 **Steiner**树加一个约束条件，要求所有的端点在 **Steiner** 树上都是叶节点
- 在这个条件下，我们结合骨架，可以设计一个算法以获得理想的切割路径，该算法在实际应用中效果非常好

切割路径的选取算法

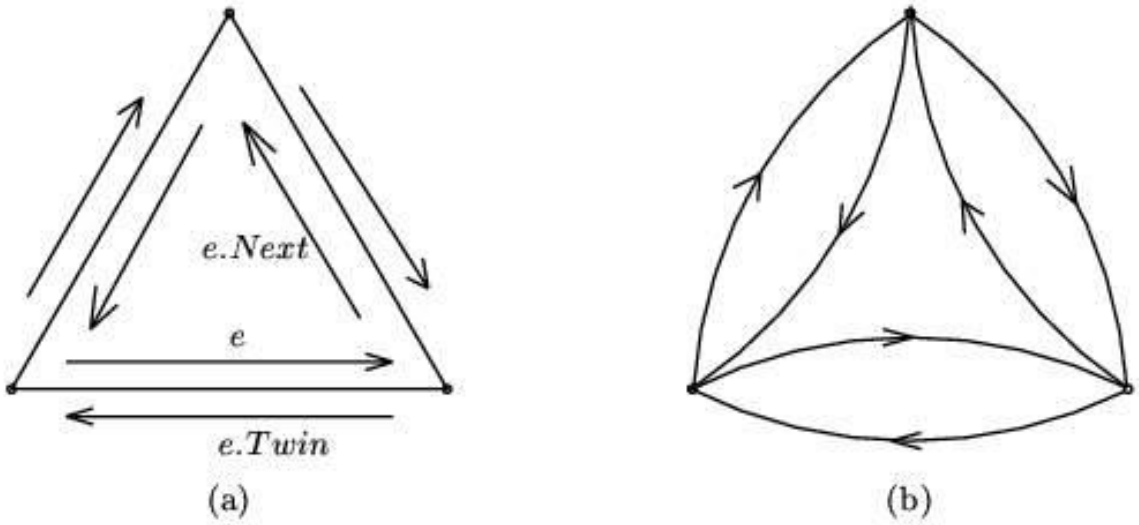
- 算法预处理：
  - 如果输入的网格曲面亏格大于零，那么先利用 **X. Gu** 的方法找到一条初始切割路径将曲面展开成同培于圆盘的片
  - 求任意两个 **terminal (**包括边界和所有切割点**)** 间的最短路径

切割路径的选取算法

- 算法概述
  - 在网格上找一个通过所有 **terminal** 的回路 **(tour)**，启发策略是回路长度最短
  - 对回路进行收缩，得到一个通过所有 **terminal** 的骨架 **(skeleton)**
  - 将骨架拉直作为最终的切割路径输出

切割路径的选取算法

- 回路 (Tour) 的计算
  - 网格的半边结构 (half-edge) 表示



Half-edge data structure and its corresponding directed weighted graph

切割路径的选取算法

- 回路 (Tour) 的计算
  - 为了能进一步收缩并得到切割路径，该回路必须满足两个要求：
    - 所有 terminal 在回路上出现且仅出现一次；
    - 回路不自交，或者说，回路将网格划分为两个区域，由于回路是有向的，我们将回路右侧的区域定义为该回路的 patch

切割路径的选取算法

- 回路的自交条件
  - 如下图， o 是 **Terminal**，● 是 非**Terminal**
  - 下图中左边回路存在自交，右边不存在

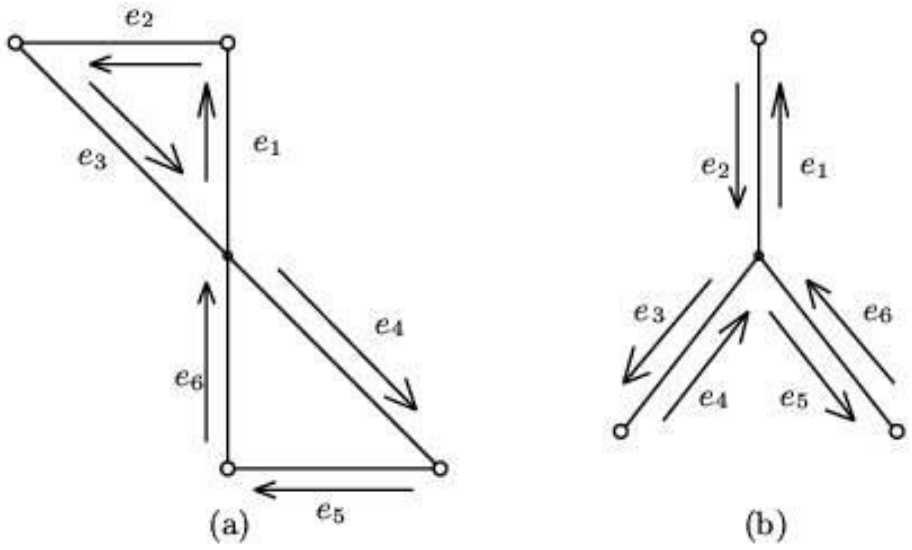


Illustration of self-intersecting and not self-intersecting tour

切割路径的选取算法

- 使用半边结构可以快速检测回路自交：

Algorithm 1 (Tour self-intersection testing)

For each half-edge e_i on a directed tour T,
For which e_j is its next adjacent edge on T,
{
  $e_k = e_i.Next$; // Next half-edge in the data structure (see Fig.5)
  While ($e_k \neq e_j$) // If is not the next edge of the tour
  {
    If e_k is a half-edge in T,
      T is self-intersecting: return;
    Else
    {
      $e_k = e_k.Twin$;
      $e_k = e_k.Next$;
      // Move to the next half-edge around the vertex
      // at the end of e_i (see Fig. 7)
    }
  }
}

切割路径的选取算法

- 回路自交检测算法示意图：

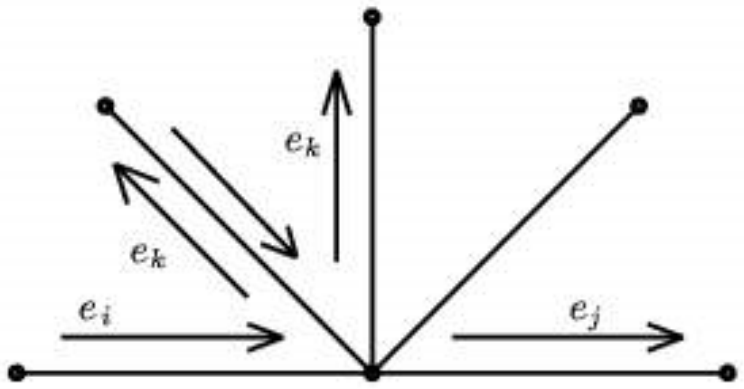


Fig. 7. Illustration of tour self-intersection testing

切割路径的选取算法

- 回路 (Tour) 的计算
  - 为了获得一条满足以上两个条件的回路，我们构造一个完全图 G_T：G_T 由所有 terminal 和它们之间的最短路径作为边构成
  - 求出上述的最短生成树 MST(G_T)，沿最短生成树遍历一圈就可以得到一个有向回路
  - 这样得到的回路包含重复的 terminal，我们需要将重复的 terminal 移除

切割路径的选取算法

- 重复 **terminal** 的移除算法：

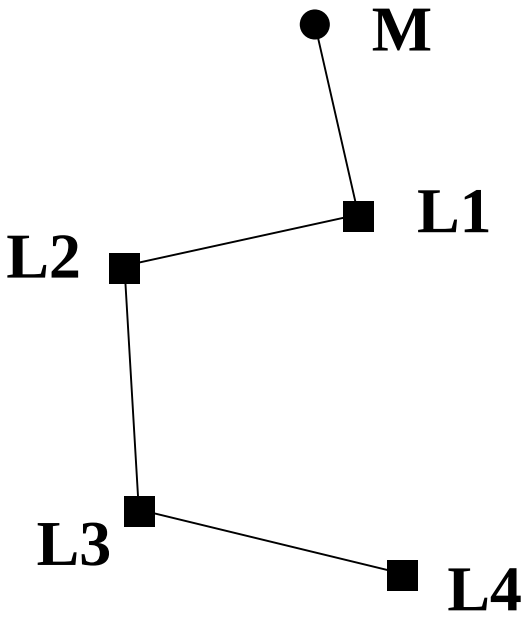
Algorithm 2 (Computing a Tour using MST)

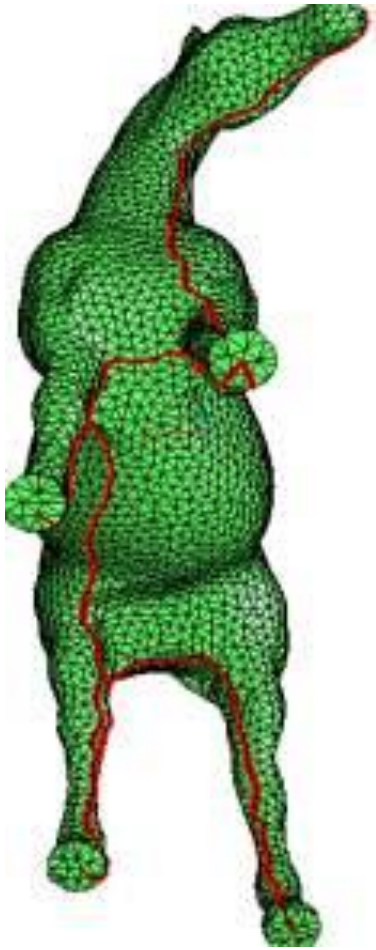
Compute the MST of G_T and walk around the tree starting at any terminal. Record the sequence of terminals in a circular list $TourList$.

```
While (size of TourList > number of terminals)
//There are repeated terminals in the tour
{
    For each repeated terminal t in TourList
    {
        If the tour is not self-intersecting when the appropriate visit
        to t is removed
            t.priority = The length of shortest path
            between t → Pre and t → Next
        Else
            t.priority = ∞
    }
    Remove the visit with minimum priority from TourList
}
```

切割路径的选取算法

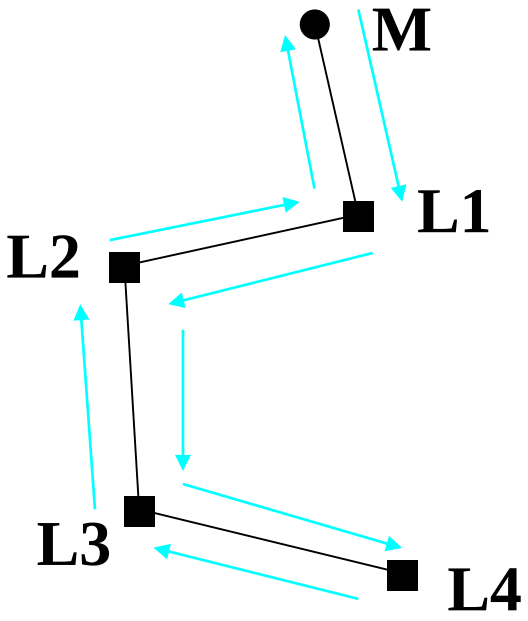
- 回路 (Tour) 的计算
  - 如右图马的例子，我们得到 G_T 的 MST，如下左图：





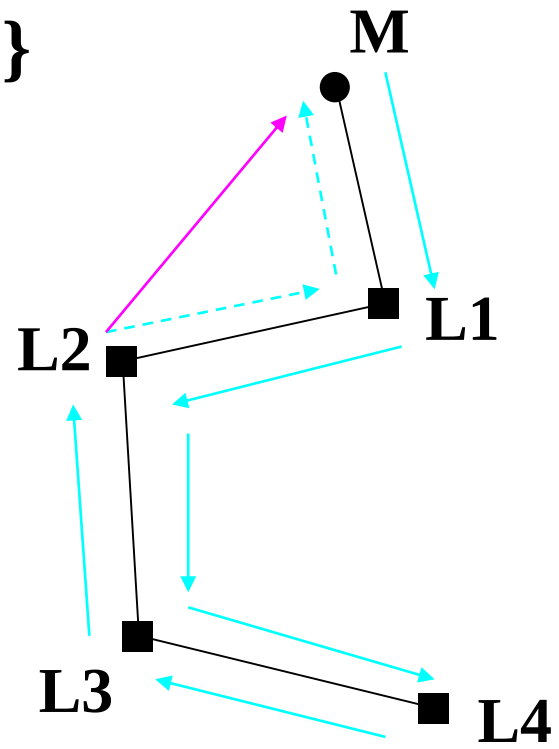
切割路径的选取算法

- 回路 (Tour) 的计算
  - 于是得到初始的回路 {M, L1, L2, L3, L4, L3, L2, L1}；显然，L1, L2, L3 重复访问



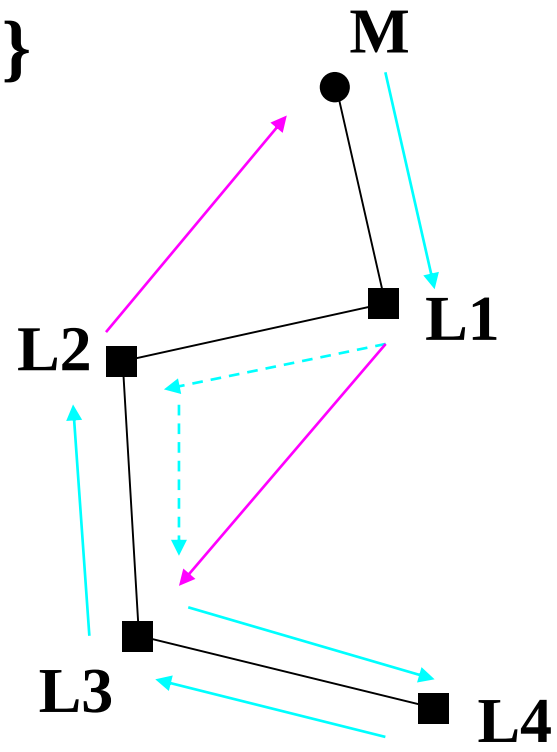
切割路径的选取算法

- 回路 (Tour) 的计算
  - 用 (L2, M) 取代 (L2, L1, M)，得到：{M, L1, L2, L3, L4, L3, L2}



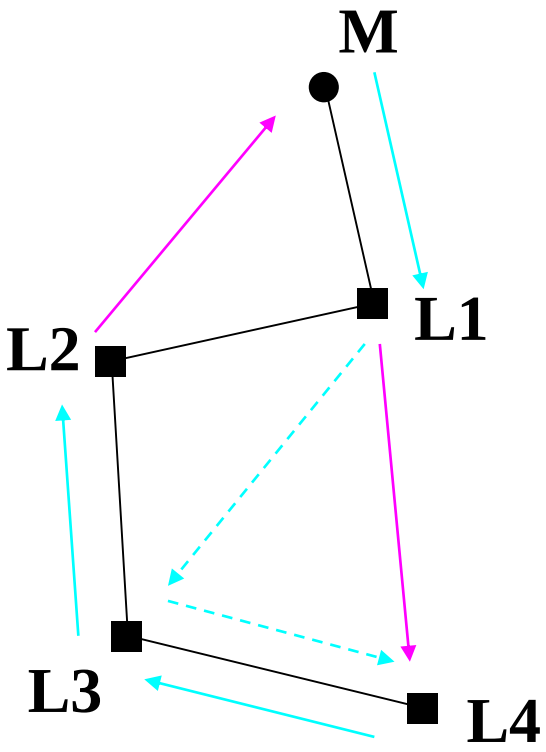
切割路径的选取算法

- 回路 (Tour) 的计算
  - 用 (L1, L3) 取代 (L1, L2, L3)，得到：{M, L1, L3, L4, L3, L2}



切割路径的选取算法

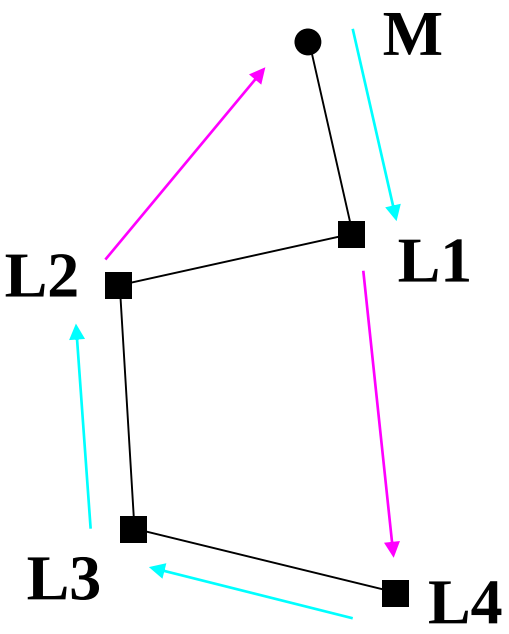
- 回路 (Tour) 的计算
  - 用 (L1, L4) 取代 (L1, L3, L4)，得到：{M, L1, L4, L3, L2}

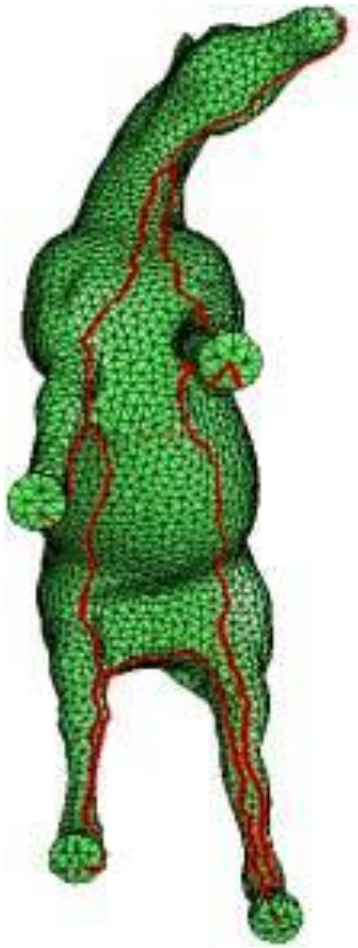


切割路径的选取算法

- 回路 (Tour) 的计算
  - 最后我们得到一个回路：

    {M, L1, L4, L3, L2}





切割路径的选取算法

- 回路 (Tour) 到骨架 (skeleton) 的收缩
  - 边标志：Tour 上的每条边都被指定一个唯一的标志
  - 标志通过BFS传播

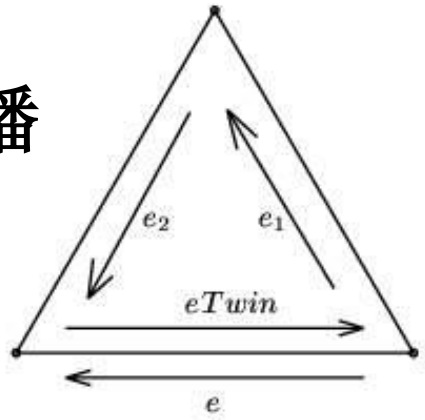


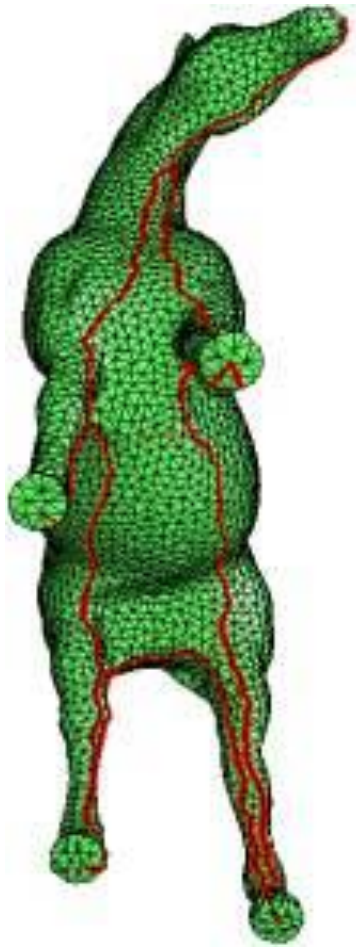
Fig. 8. Illustration of mark propagation in skeleton computing

  - 正向反向标记不同的edge即可加入骨架

切割路径的选取算法

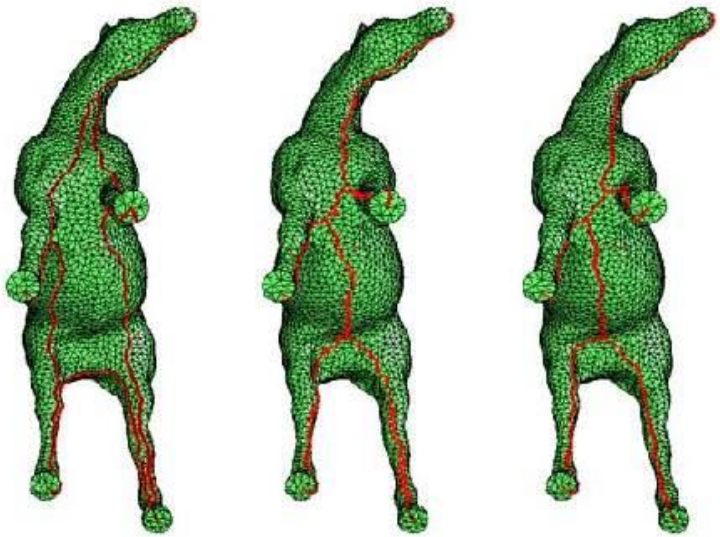
- 骨架 (skeleton) 的生成算法

$eTwin = e.Twin$;
If($eTwin$ is not marked)
// Then propagate the mark to the twin half-edge
{
    $e_1 = eTwin.Next$;
    $e_2 = e_1.Next$;
    $eTwin.mark = e_1.mark = e_2.mark = e.mark$;
    $q.add(e_1)$;
    $q.add(e_2)$;
}
Else if ($eTwin.mark \neq e.mark$)
Export e; // This edge is part of the skeleton



切割路径的选取算法

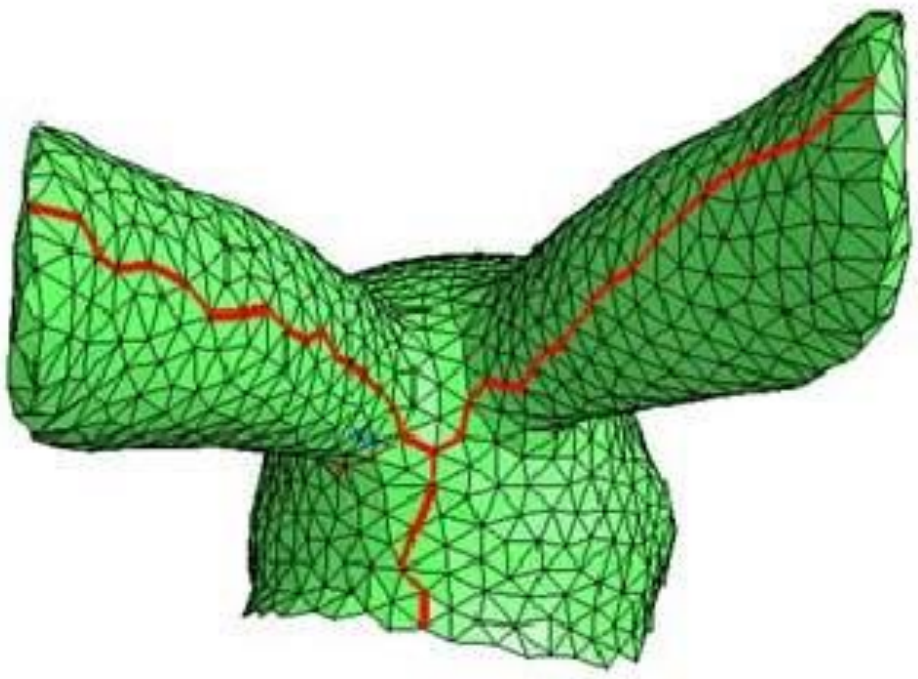
- 骨架的拉直
  - 由于骨架引进了Steiner点，会有锯齿状
  - 对 Steiner 点和 terminal 之间的路径用最短路径代替



切割路径的选取算法

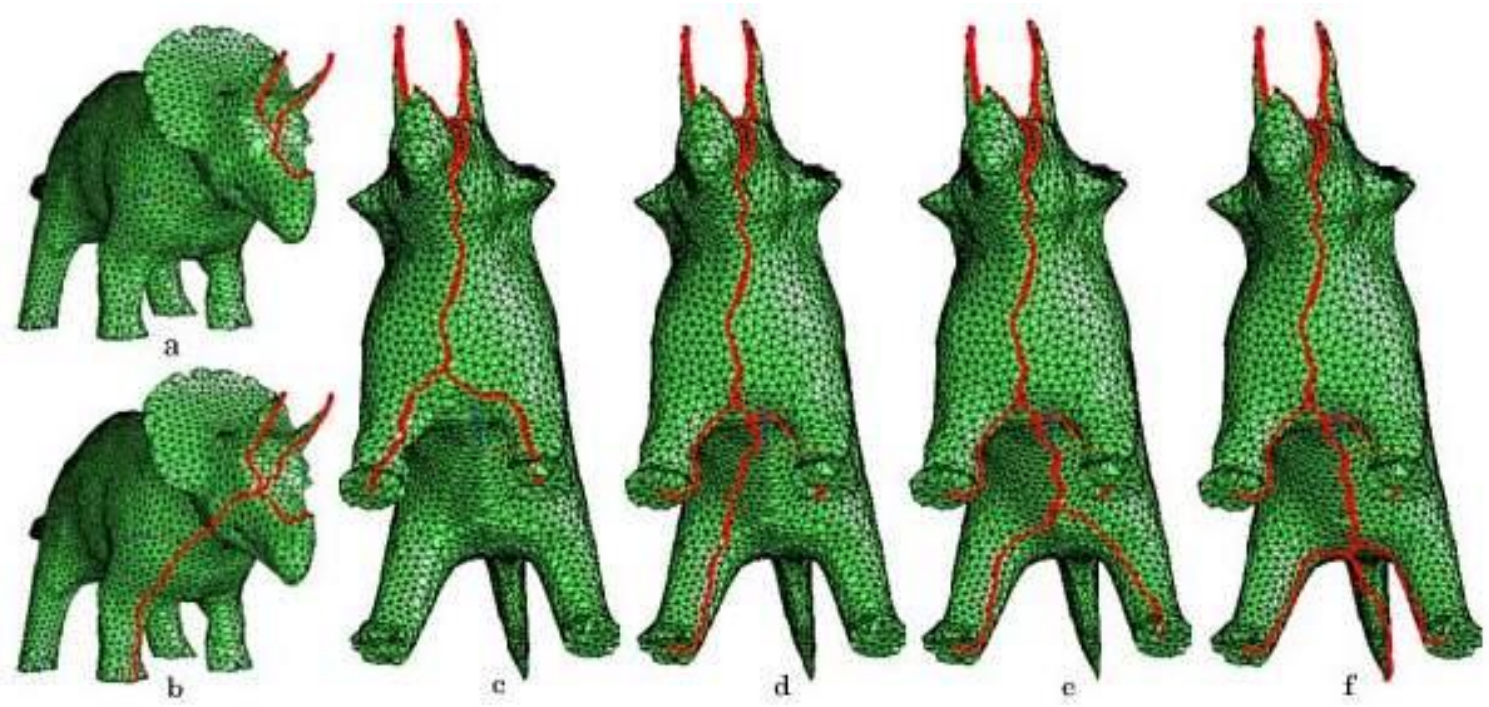
- 实例
  - 开边界模型：边界作为 **terminal**
  - 手工确定 **terminal**
  - 视觉效应：路径从何处走？

切割路径的选取算法



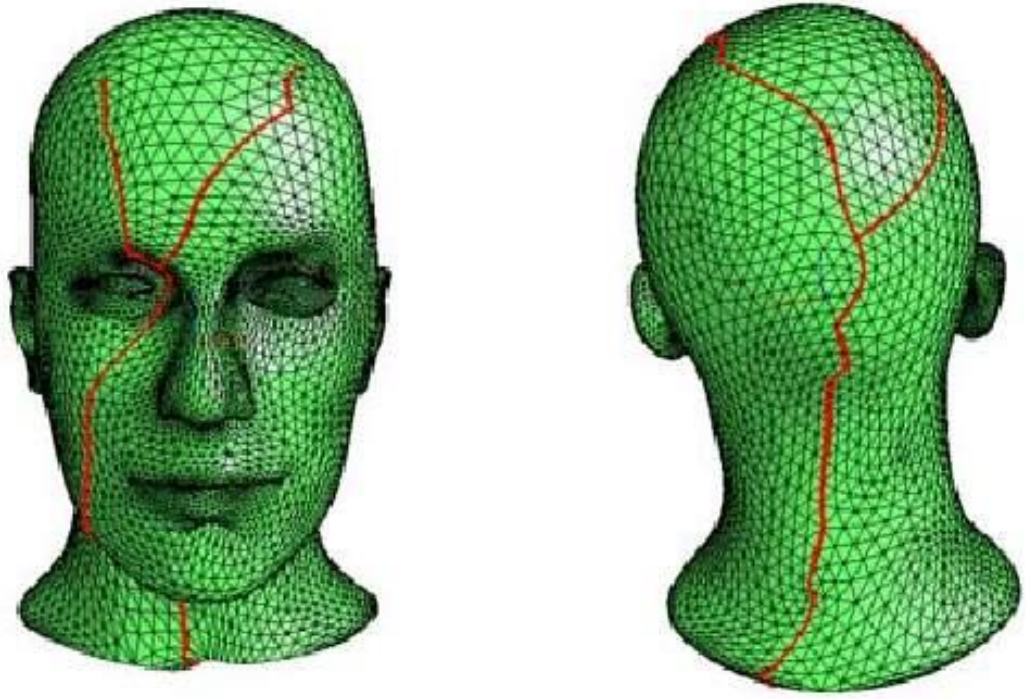
Computing seam on an open model (a bunny head)

切割路径的选取算法



Seams by interactively adding new extremal vertices

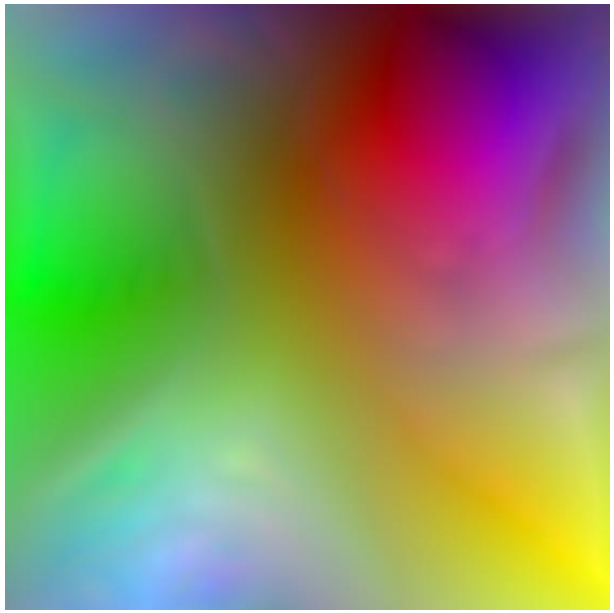
切割路径的选取算法

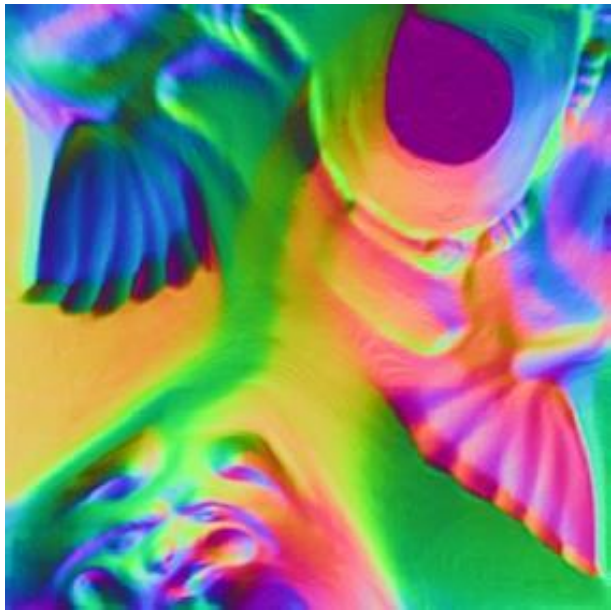


Seam computed according to different view points

几何图像技术

- 几何图像技术关注于如何将二维网格流形使用均匀参数域的图像来进行表示



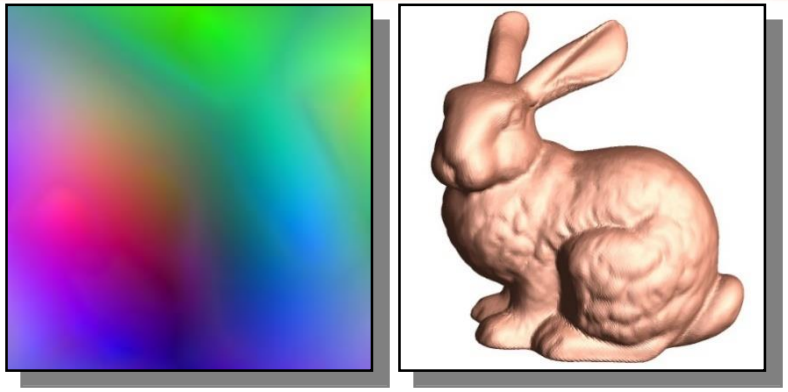


几何图像技术

- 几何图像 (Geometry Image)
  - **Xianfeng Gu, Steven J. Gortler, and Hugues Hoppe**
  - **SIGGRAPH 2002**
  - **1125** 次引用

- 意义：通过几何图像技术，许多图像处理技术将可以直接应用到网格模型上
- 几何图像技术的核心是平面参数化

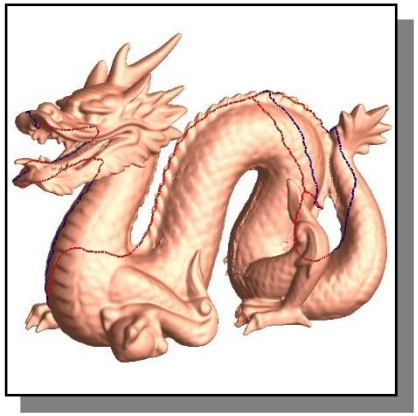
几何图像技术

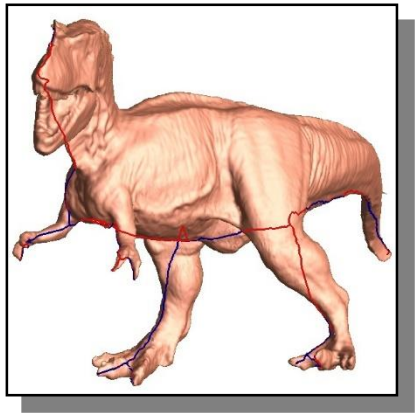


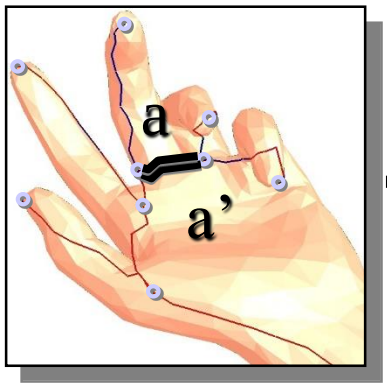
- 什么是几何图像？
  - 用图像的方式表示任意三维几何模型
  - 例如：右图中的**Bunny**可以使用左边这种的图像进行表示。
  - 为了保证模型质量，几何图像采用了三通道、**12**位来表示三维几何位置**X/Y/Z**；采用三通道、**8**位来表示三维表面法向**Nx/Ny/Nz**。

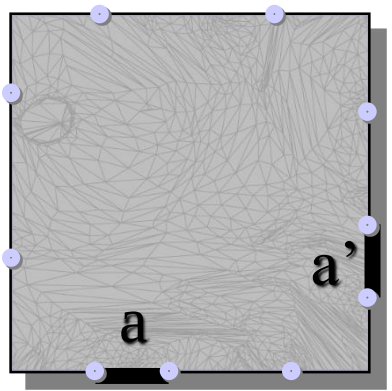
几何图像技术

- 如何构造几何图像？
  - 将连通的二维网格流形切割并展开成为一个与圆盘同胚的网格
  - 切割后，首先对边界进行参数化，需满足两个条件：
    - 参数化后，两侧等长
    - 端点位于图像格点
  - 否则将会产生裂痕



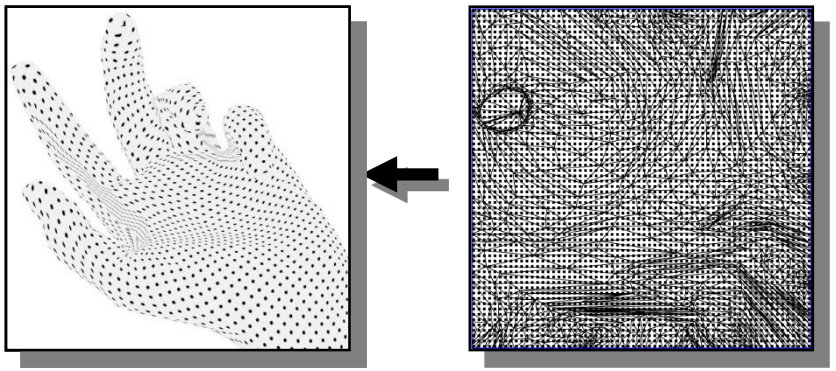






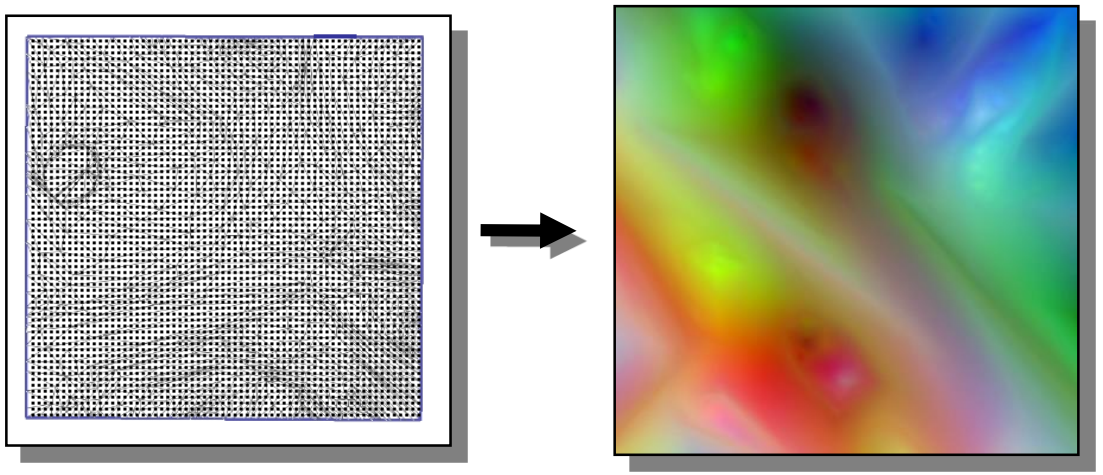
几何图像技术

- 如何构造几何图像？
  - 完成边界参数化后，需进一步对内部进行参数化。
  - 这里，需要对均匀采样点在模型表面的间距进行度量并最小化，使得采样点得以均匀分布在模型表面上。



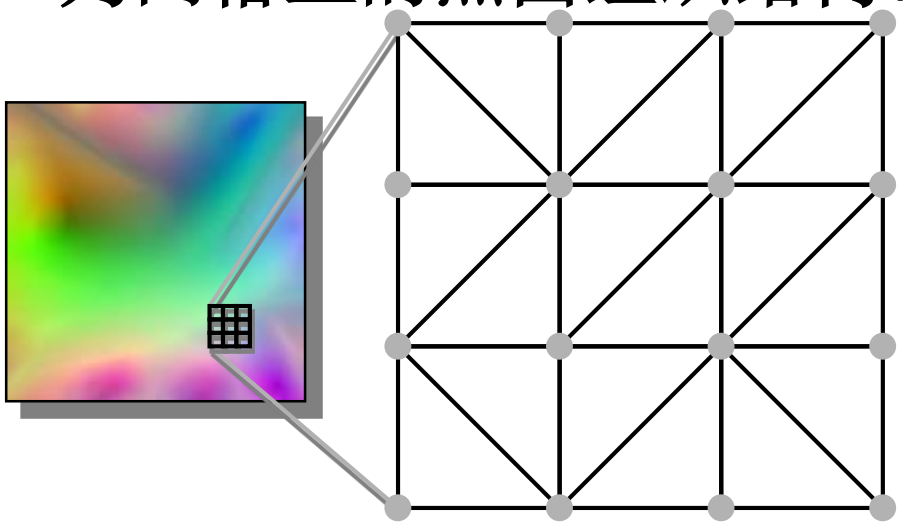
几何图像技术

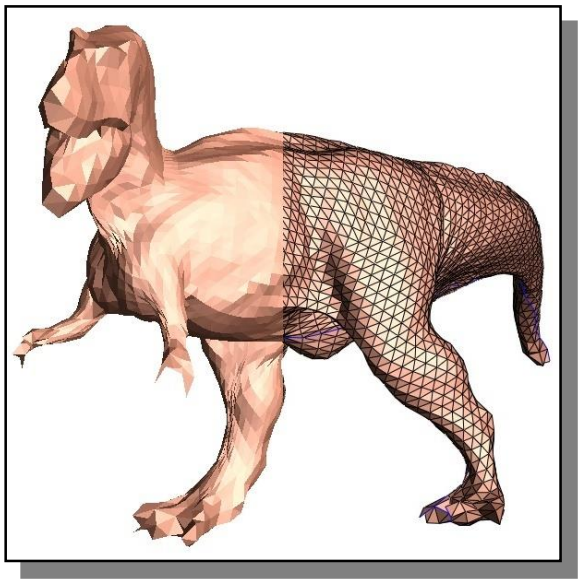
- 如何构造几何图像？
  - 最后，根据采样点所对应的模型表面的属性，将采样点的模型特征(位置和表面法向)写在几何图像上。



几何图像技术

- 如何根据几何图像重新得出三维模型？
  - 对于每一个四像素组成的块，选取空间距离较小的对角线，将其分为两个小三角形，作为网格上的点面组织结构。

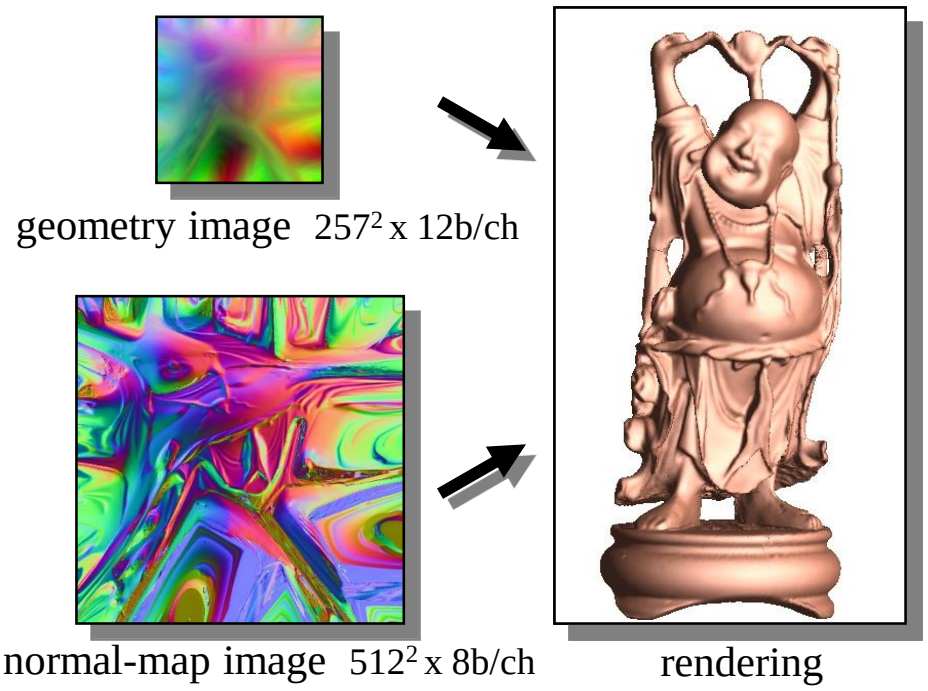




(65x65 geometry image)

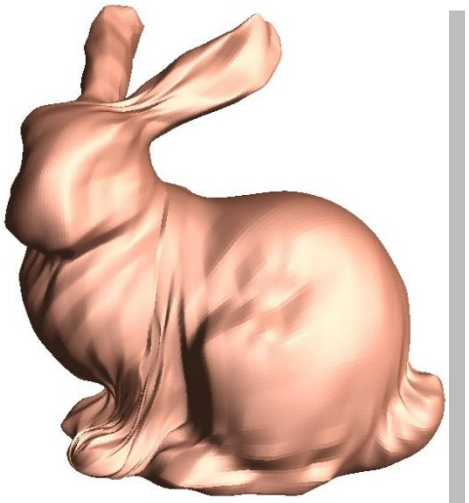
几何图像技术

- 如何根据几何图像重新得出三维模型？
  - 法向量纹理可以用相同的参数化方式，和几何图像一并得出，并用于渲染过程。

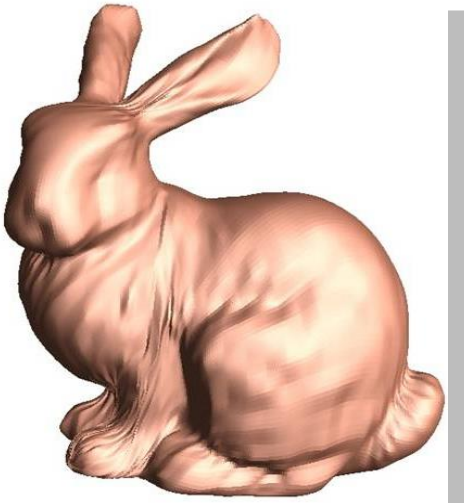


几何图像技术

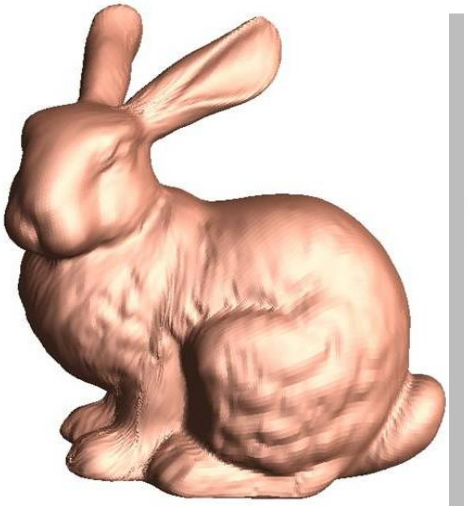
- 几何图像的应用1：
  - 使用图像压缩的技术压缩网格。
    - 原网格大小：**295KB**，采用小波技术进行图像压缩：



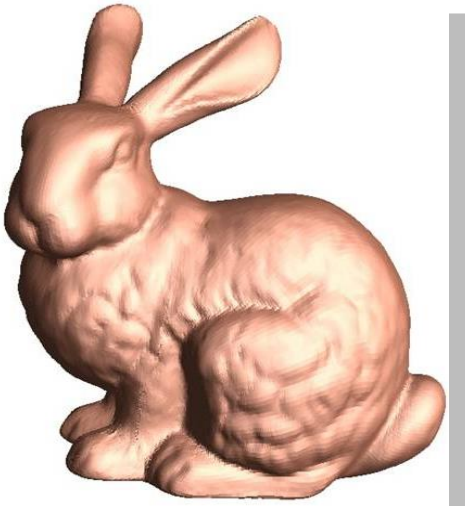
1.5 KB



3 KB



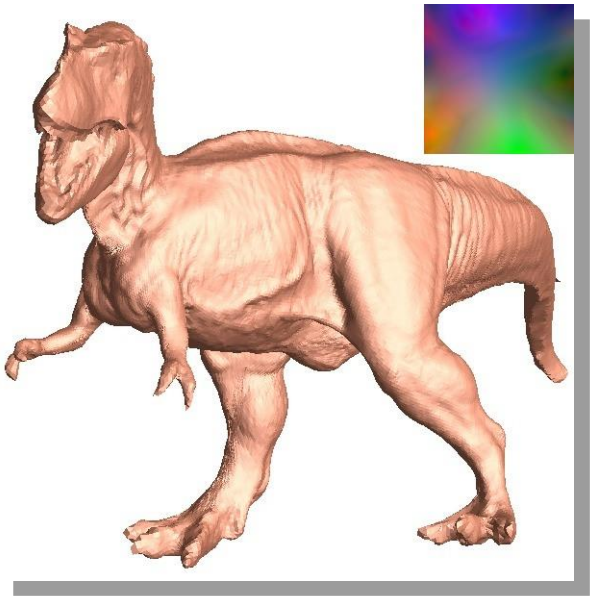
12 KB



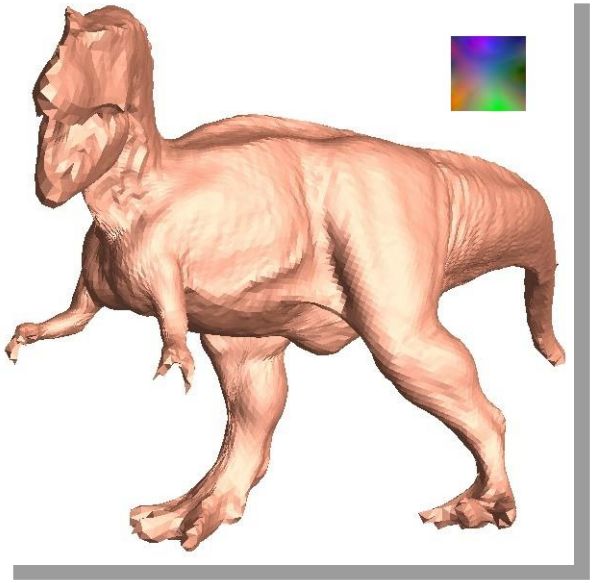
49 KB

几何图像技术

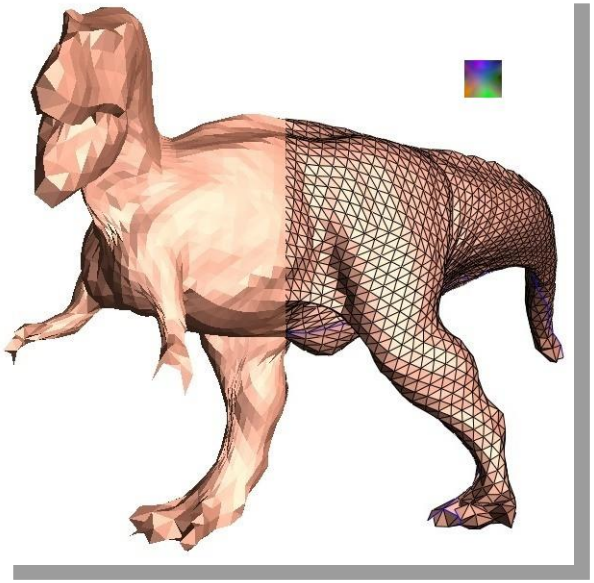
- 几何图像的应用2：
  - 使用图像LoD(level-of-detail)完成网格简化。



257x257



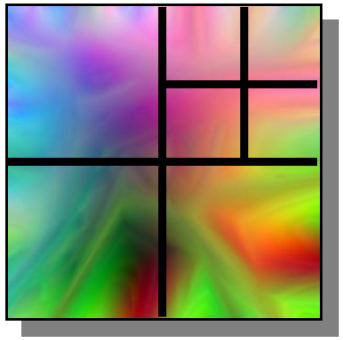
129x129



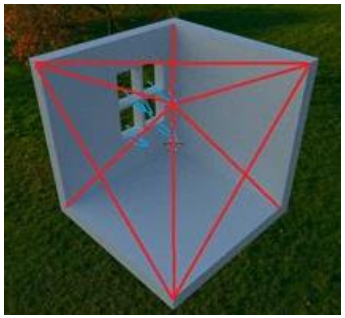
65x65

几何图像技术

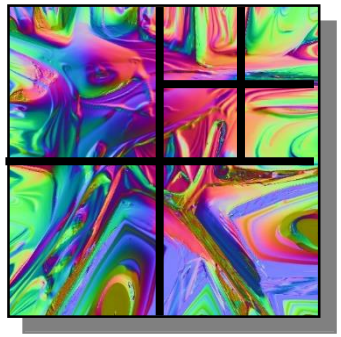
- 几何图像的应用3：
  - 使用四叉树加速求交判断。
    - 几何图像上的四叉树：判断模型区域是否在视锥内。
    - 法向图像上的四叉树：判断背朝视点的面片(backface)。



几何图像



Backface Culling



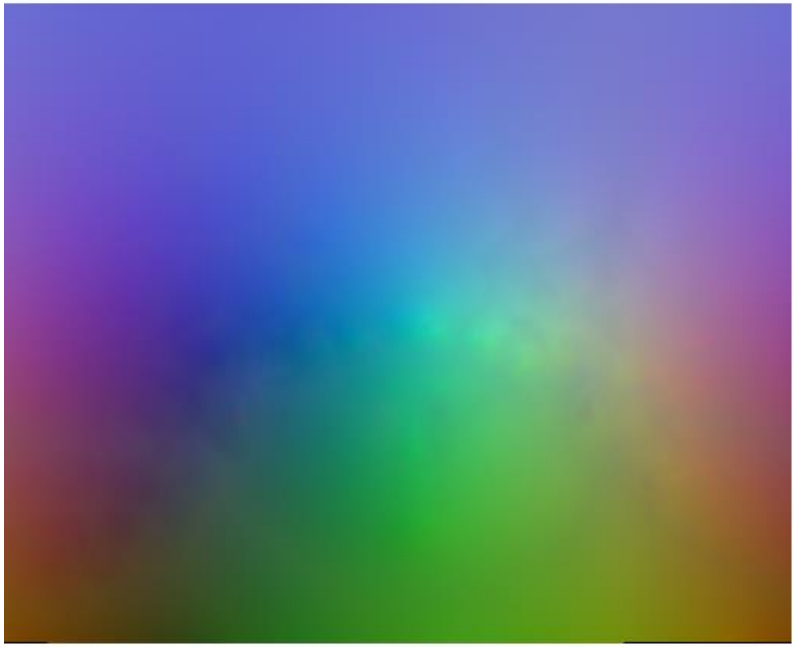
法向图像

Epic Games最新公布的虚幻引擎5

虚幻引擎5

- 核心技术：
  - **Nanite**虚拟微多边形几何技术
  - **Lumen**动态全局光照技术
- “Nanite几何体可以被实时流送和缩放，因此无需再考虑多边形数量预算、多边形内存预算或绘制次数预算了；也不用再将细节烘焙到法线贴图或手动编辑细节层次（LOD）”

虚幻引擎5采用的几何图像技术



虚幻引擎5

- Nanite技术中的曲面包含几何图像、颜色纹理、金属材质纹理、光学特性等几张纹理。
- 优点：渲染过程中可以动态决定解析率(mipmap pyramid)，可以动态细分加密(subdivision)，动态生成纹理，存储全局光照的中间计算结果等等。
- 几何图像的格式，使得内存访问模式更加有规律，更加容易预测，从而提高内存访问效率，极大地提高了数据调度的速度。

虚幻引擎5

- https://www.bilibili.com/video/BV1BK411W75W

今日人物： Daniel Cohen-OR

- Ben-Gurion大学获得数学和计算机双学士学位（1985）、计算机硕士学位（1986），纽约州立大学石溪分校博士学位（1991年）现为特拉维夫大学教授
- Cohen-OR教授发表128篇ACM SIGGRAPH /TOG论文，排名第一，论文引用42464次
- 培养的学生和博士后多人获SIGGRAPH杰出新人奖，本人获2012年度中国友谊奖，2018年ACM SIGGRAPH成就奖。

谢谢！