中国科学技术大学
University of Science and Technology of China

GAMES102

GAMES 102在线课程

几何建模与处理基础

刘利刚

中国科学技术大学

中国科学技术大学
University of Science and Technology of China

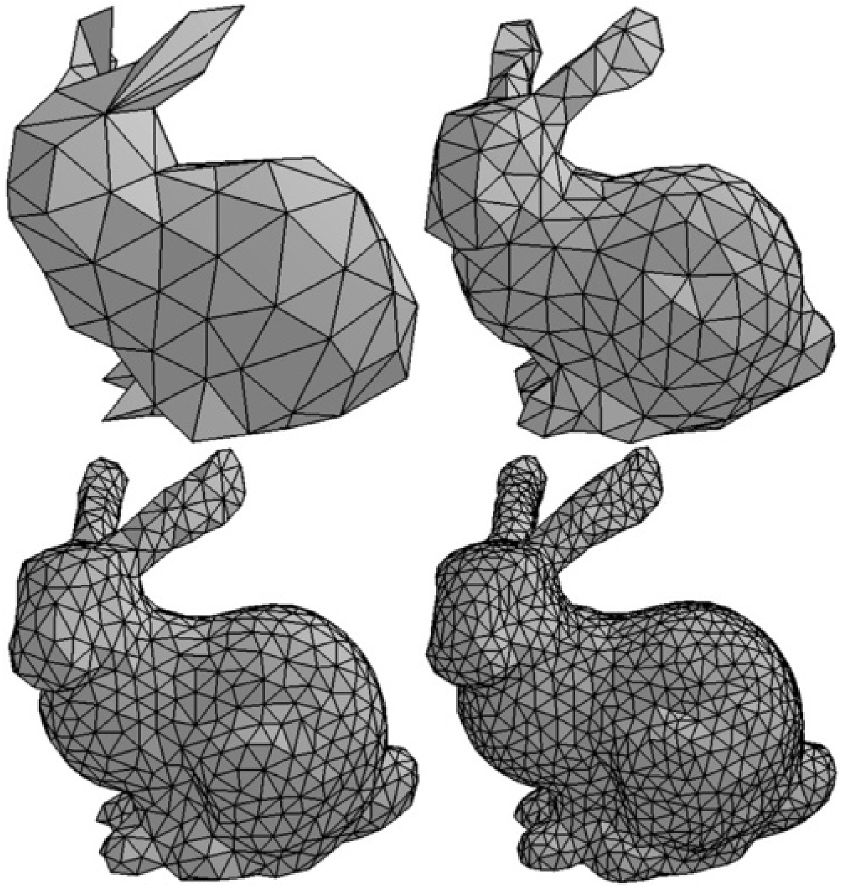
GAMES102

GAMES 102在线课程：几何建模与处理基础

离散微分几何

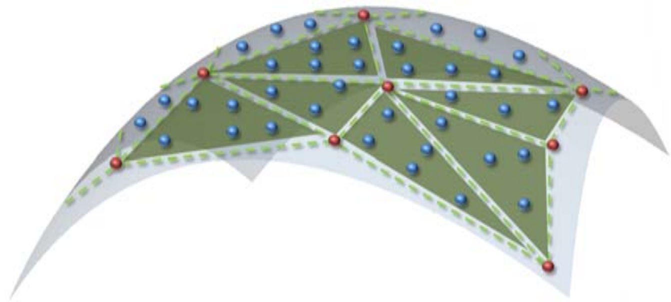
回顾：三角网格曲面

陶瓷兔子实物

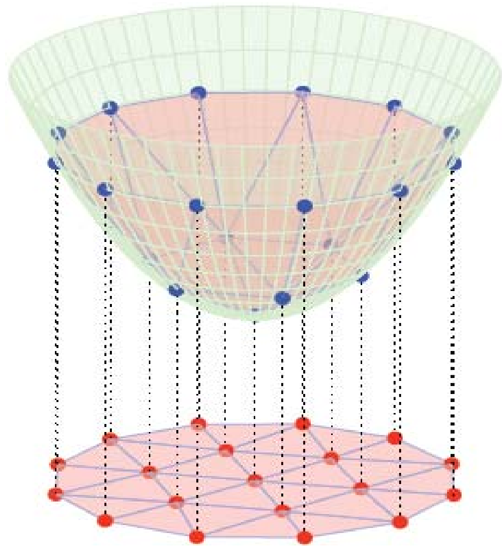


回顾：三角网格曲面

- 观点1：曲面的离散逼近
  - 采样：顶点为从曲面上的采样点
  - 构网：每个三角面为线性平面
  - 本质：分片线性逼近



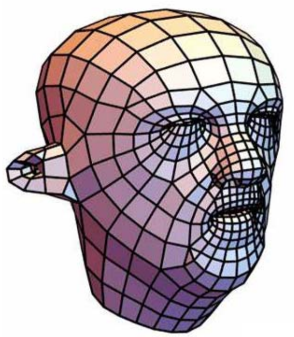
- 观点2：平面图的嵌入
  - 平面图
  - 图的顶点提升 (lifting) 至三维空间
  - 本质：二维流形

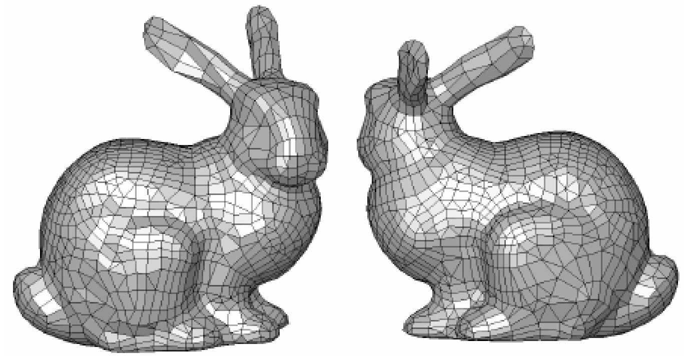


回顾：数据结构—图 (graph)

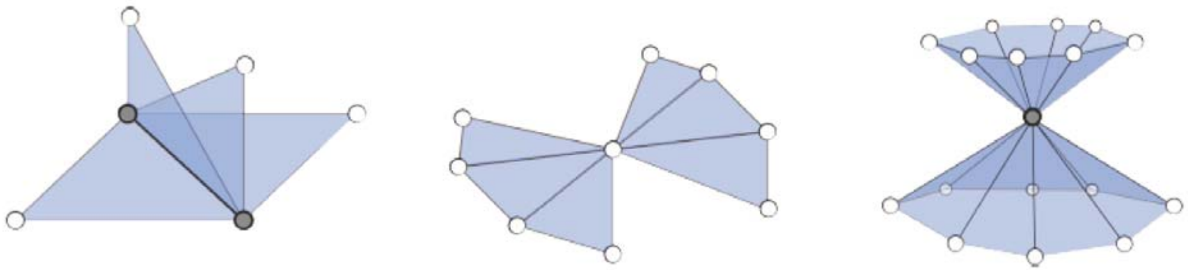
```
v 1.0 0.0 0.0
v 0.0 1.0 0.0
v 0.0 -1.0 0.0
v 0.0 0.0 1.0
f 1 2 3
f 1 4 2
f 3 2 4
f 1 3 4
```

- G={V, E, F}
  - V: 顶点集合；E: 边集合；F: 三角形集合
  - 有其中两个集合可推出另一个集合
- 多边形网格均可转化为三角网格





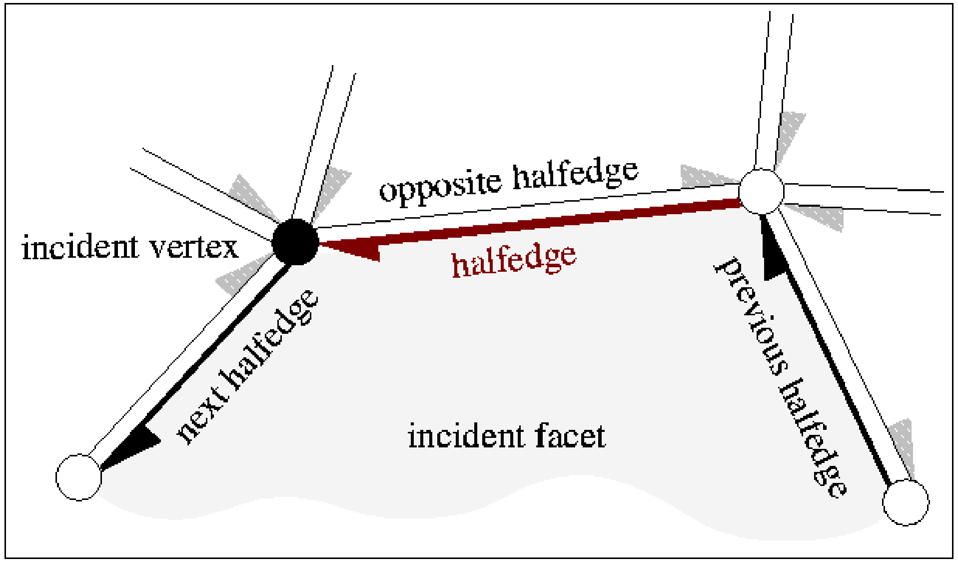
- 不考虑非流形结构



三角网格编程初步

半边 (half-edge) 数据结构

- 半边结构：以“边”为中心的数据结构
  - 网格连接关系存储在边上，每条边表达为两条“半边”
  - 目的：提高点线面的查询或增删改操作的效率



https://www.flipcode.com/archives/The_Half-Edge_Data_Structure.shtml

半边 (half-edge) 数据结构

基本边、点、面数据结构

```
struct HE_edge
{

    HE_vert* vert;
    HE_edge* pair;
    HE_face* face;
    HE_edge* next;

};
```

```
struct HE_vert
{

    float x;
    float y;
    float z;

    HE_edge* edge;

};
```

```
struct HE_face
{

    HE_edge* edge;

};
```

邻域关系查询方法

由边找两顶点及两邻面

```
HE_vert* vert1 = edge->vert;
HE_vert* vert2 = edge->pair->vert;

HE_face* face1 = edge->face;
HE_face* face2 = edge->pair->face;
```

由面找其所有半边

```
HE_edge* edge = face->edge;

do {

    // do something with edge
    edge = edge->next;

} while (edge != face->edge);
```

由顶点找其所有半边

```
HE_edge* edge = vert->edge;

do {

    edge = edge->pair->next;

} while (edge != vert->edge);
```

https://www.flipcode.com/archives/The_Half-Edge_Data_Structure.shtml

几何（网格）处理库

- CGAL: http://www.cgal.org
- Libigl: https://github.com/libigl/libigl
- MeshLab: http://www.meshlab.net
- OpenMesh: https://www.openmesh.org
- PCL (Point Cloud Library): http://www.pointclouds.org
- TriMesh: http://graphics.stanford.edu/software/trimesh
- DGtal: https://dgtal.org

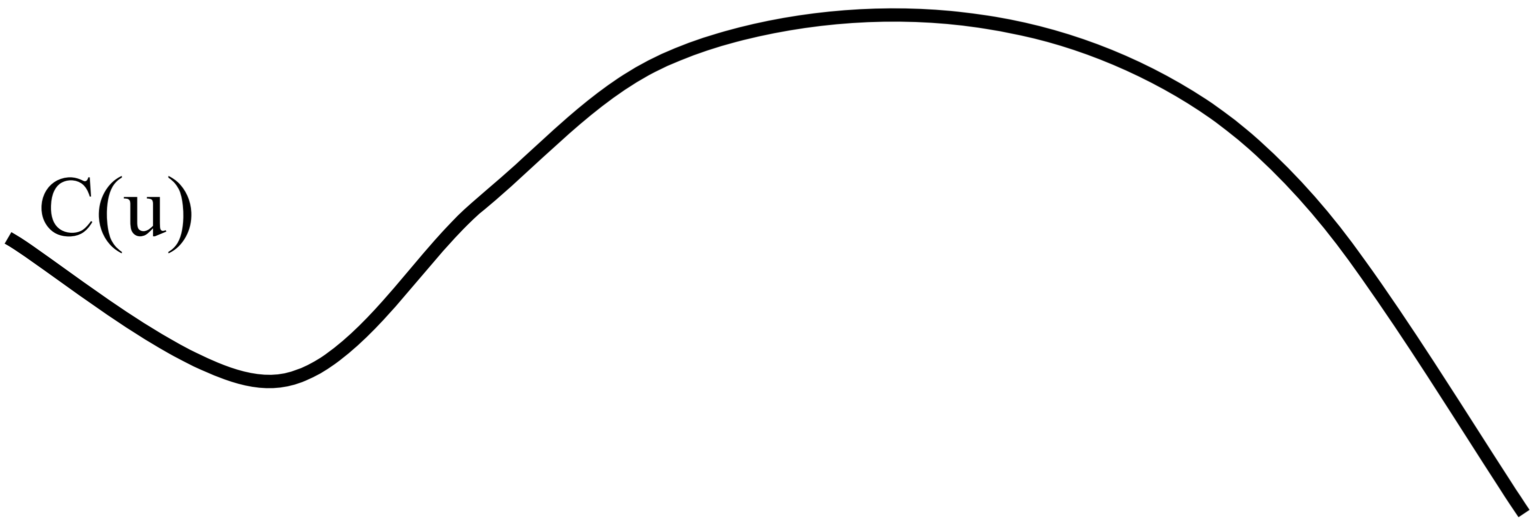
- 本课程作业框架： Utopia (USTC自研）

http://staff.ustc.edu.cn/~lgliu/Resources/CG/3D_modeling.htm

曲线曲面的微分几何

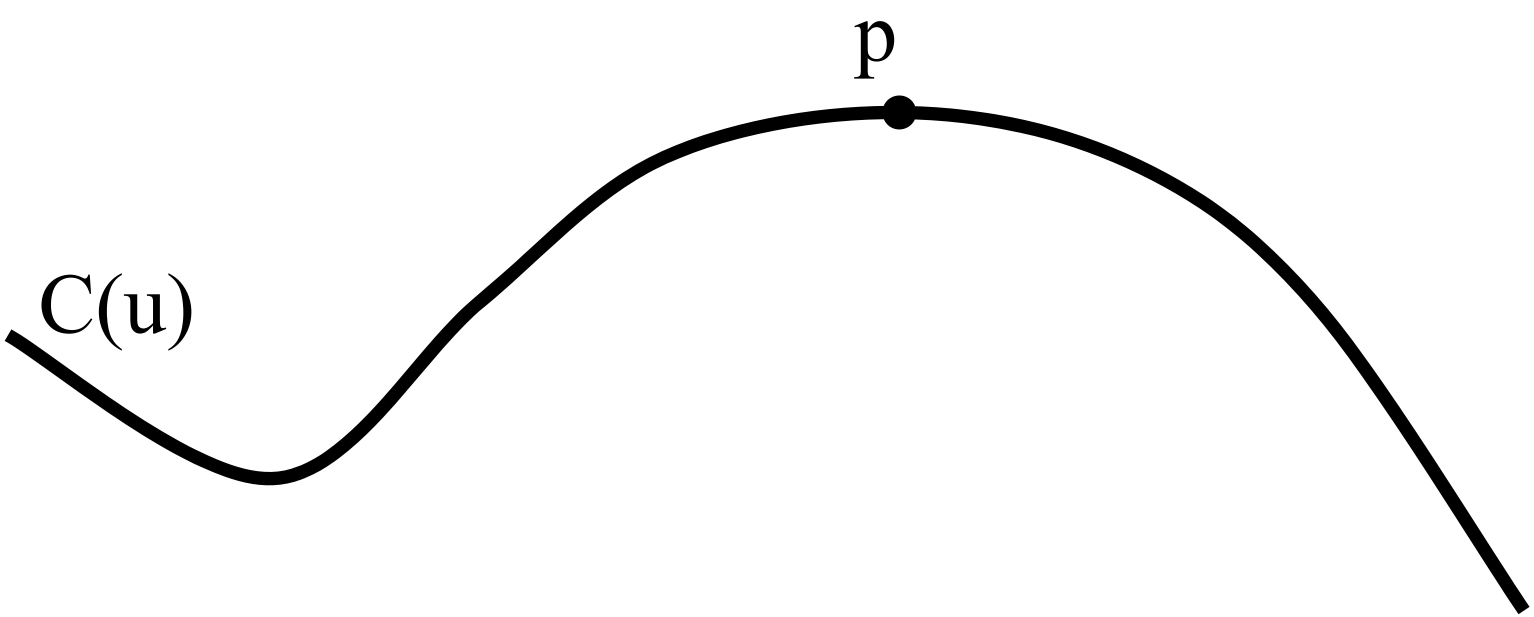
Curves

Differential Geometry of a Curve



Differential Geometry of a Curve

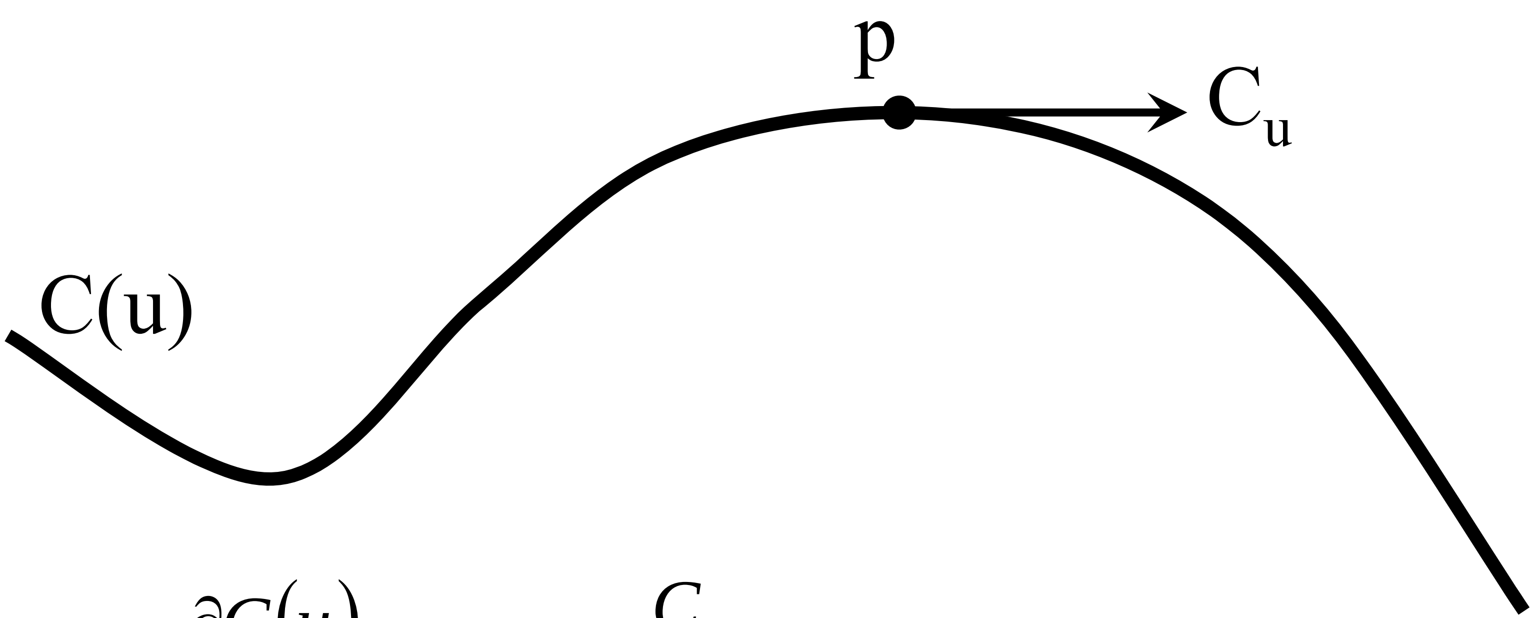
Point p on the curve at u_0



$p=C(u_0)$

Differential Geometry of a Curve

Tangent T to the curve at u_0

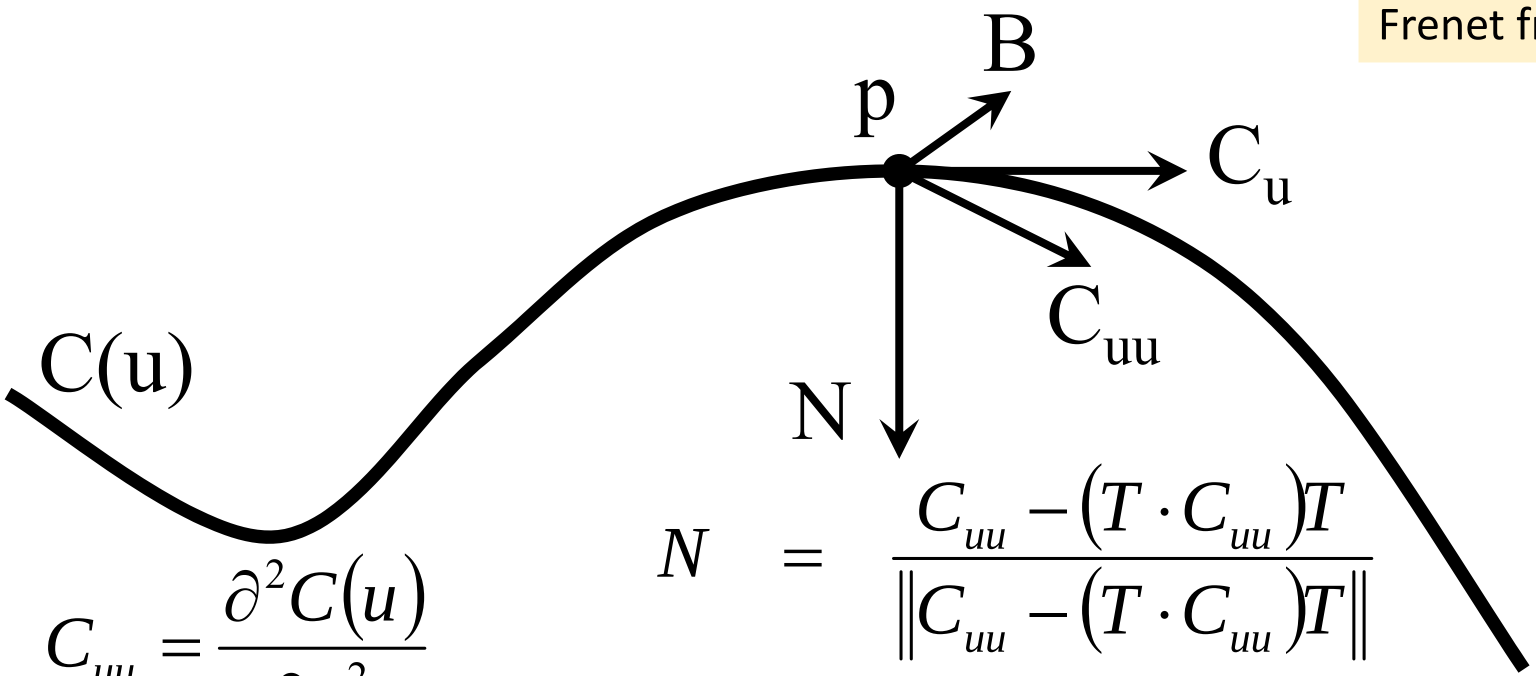


$$C_u = \frac{\partial C(u)}{\partial u} \qquad T = \frac{C_u}{\|C_u\|}$$

Differential Geometry of a Curve

Normal N and Binormal B to the curve at u_0

Frenet frame: {T,N,B}



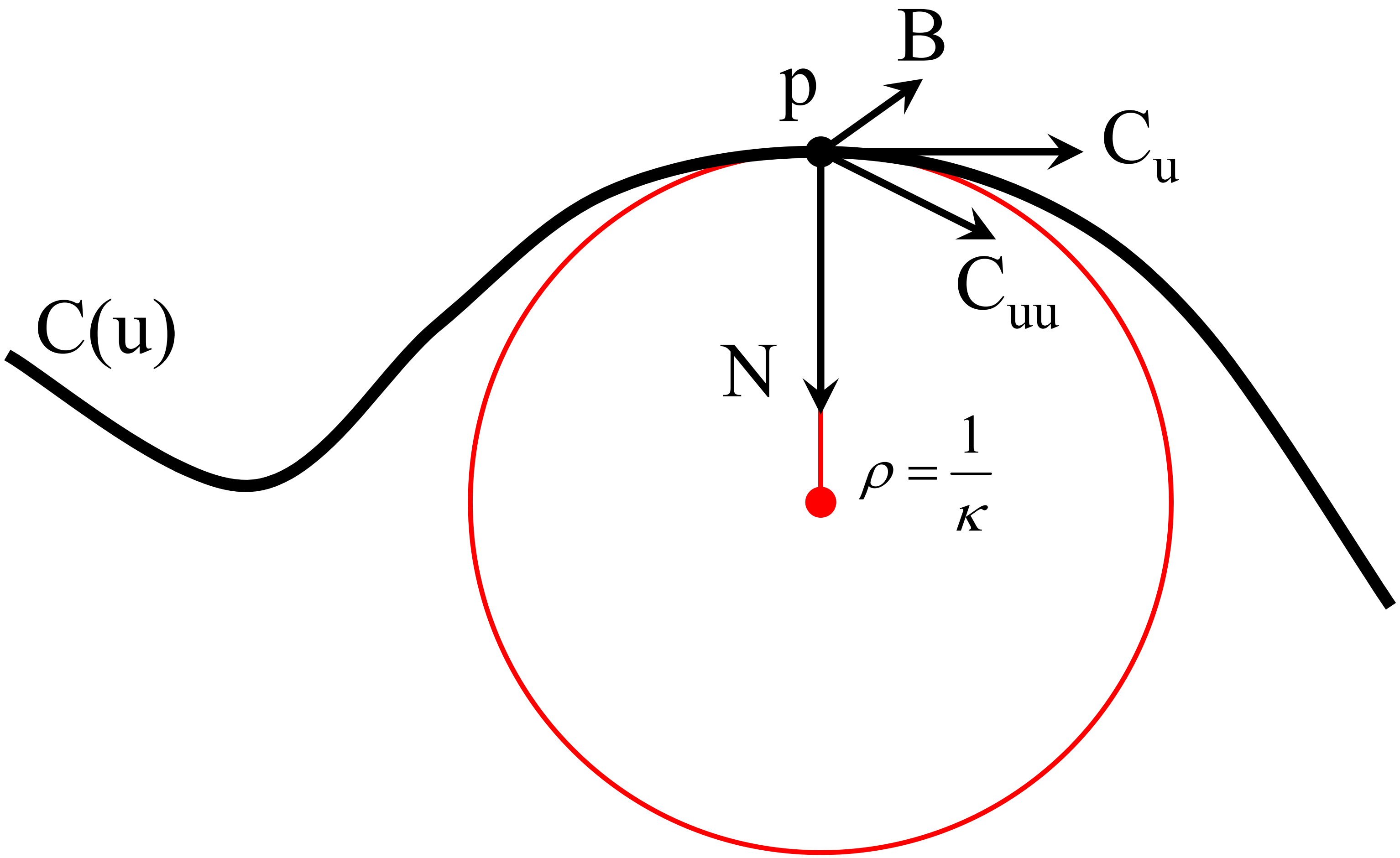
$$C_{uu} = \frac{\partial^2 C(u)}{\partial u^2}$$

$$B = \frac{C_u \times C_{uu}}{\|C_u \times C_{uu}\|}$$

$$\begin{aligned} N &= \frac{C_{uu} - (T \cdot C_{uu})T}{\|C_{uu} - (T \cdot C_{uu})T\|} \\ &= B \times T \\ &= \frac{(C_u \times C_{uu}) \times C_u}{\|C_u \times C_{uu}\|\|C_u\|} \end{aligned}$$

Differential Geometry of a Curve

Curvature κ at u_0 and the radius ρ osculating circle



Curves

- Tangent vector to curve $C(t)=(x(t),y(t))$ is

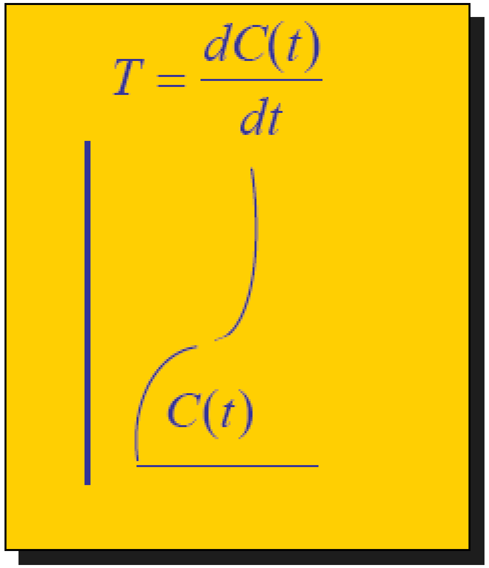
$$T = C'(t) = \frac{dC(t)}{dt} = [x'(t), y'(t)]$$

- Unit length tangent vector

$$\vec{T} = \vec{C}(t) = \frac{[x'(t), y'(t)]}{\sqrt{x'(t)^2 + y'(t)^2}}$$

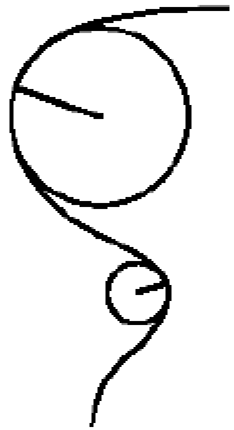
- Curvature

$$k(t) = \frac{x'(t)y''(t) - y'(t)x''(t)}{\left(x'(t)^2 + y'(t)^2\right)^{3/2}}$$

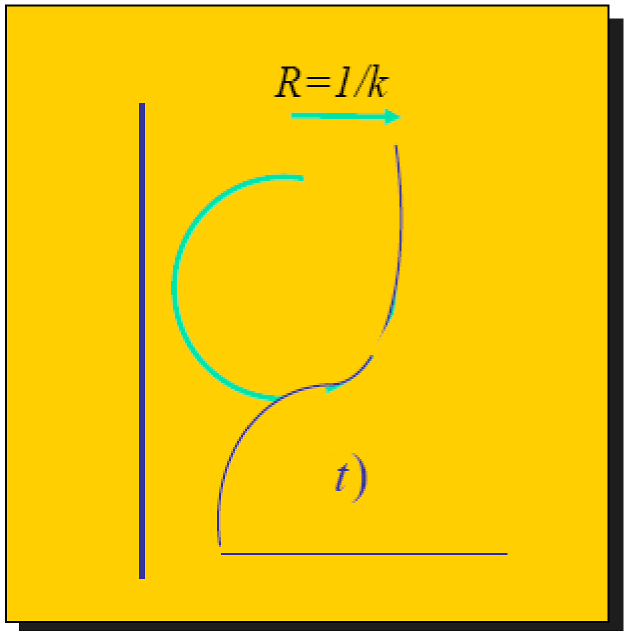


Curve Curvature

- Curvature is **independent** of parameterization
  - $C(t)$, $C(t+5)$, $C(2t)$ have same curvature (at corresponding locations)



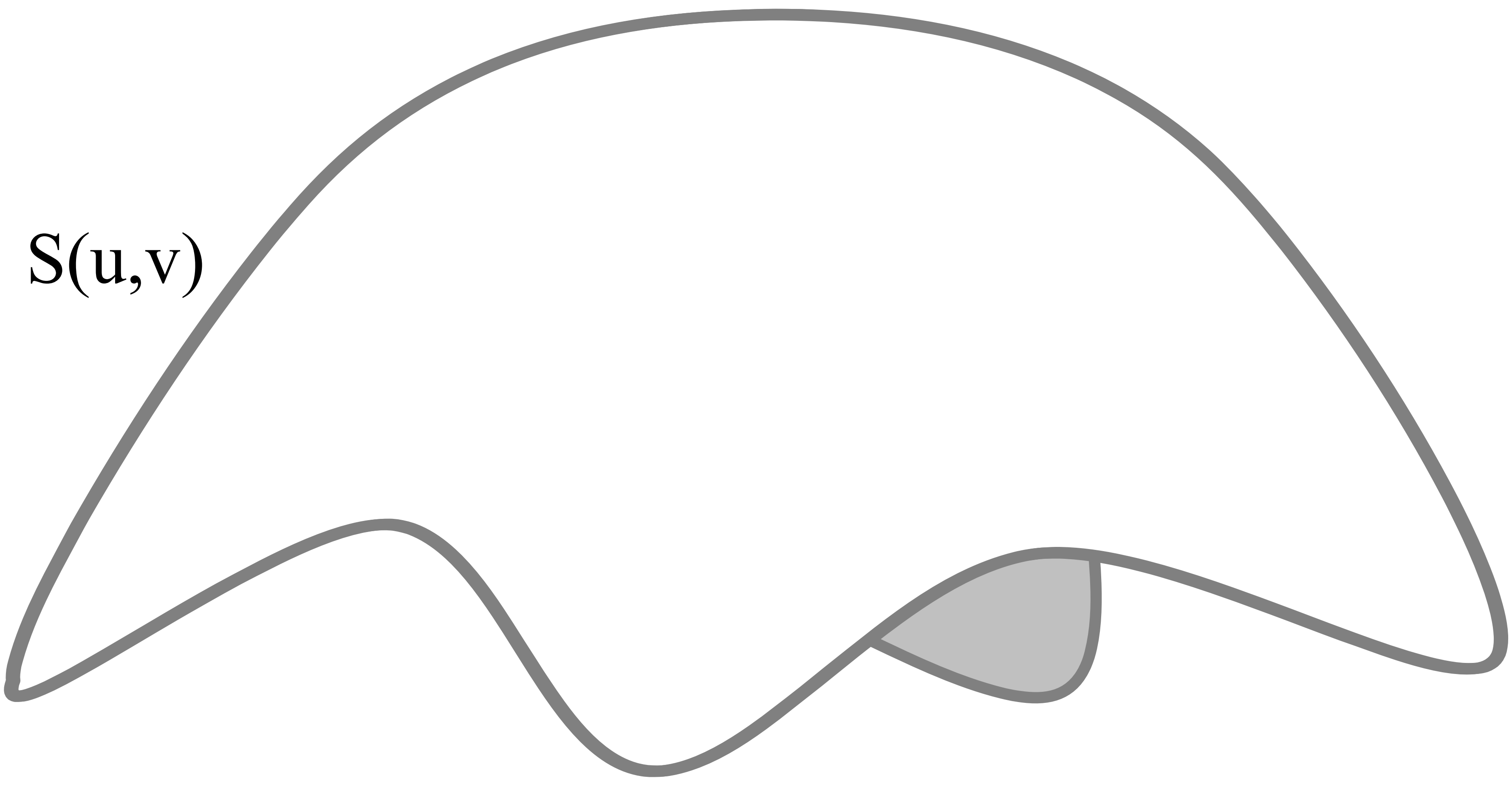
- Corresponds to radius of osculating circle $R=1/k$



- Measure curve bending

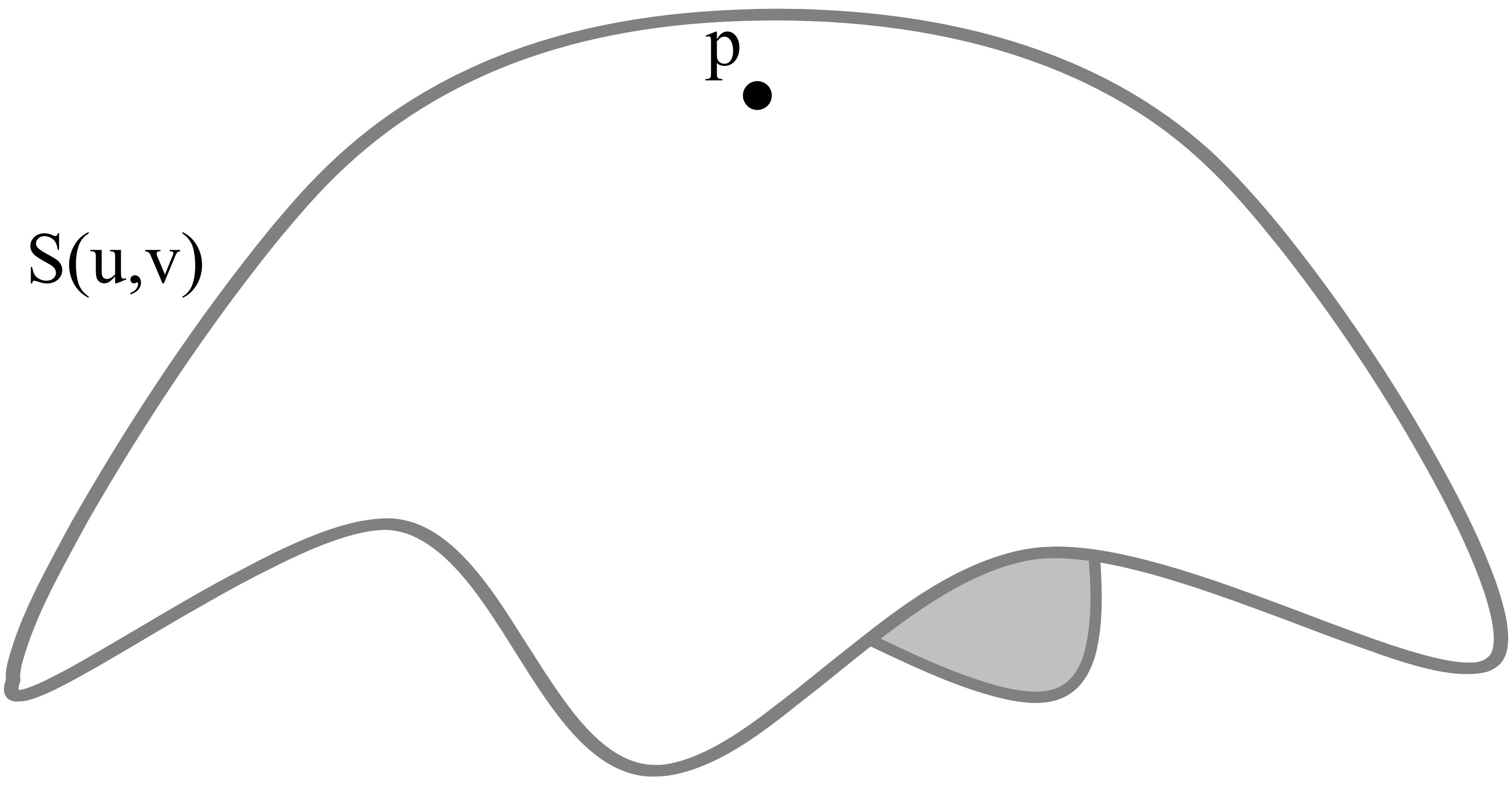
Surfaces

Differential Geometry of a Surface



Differential Geometry of a Surface

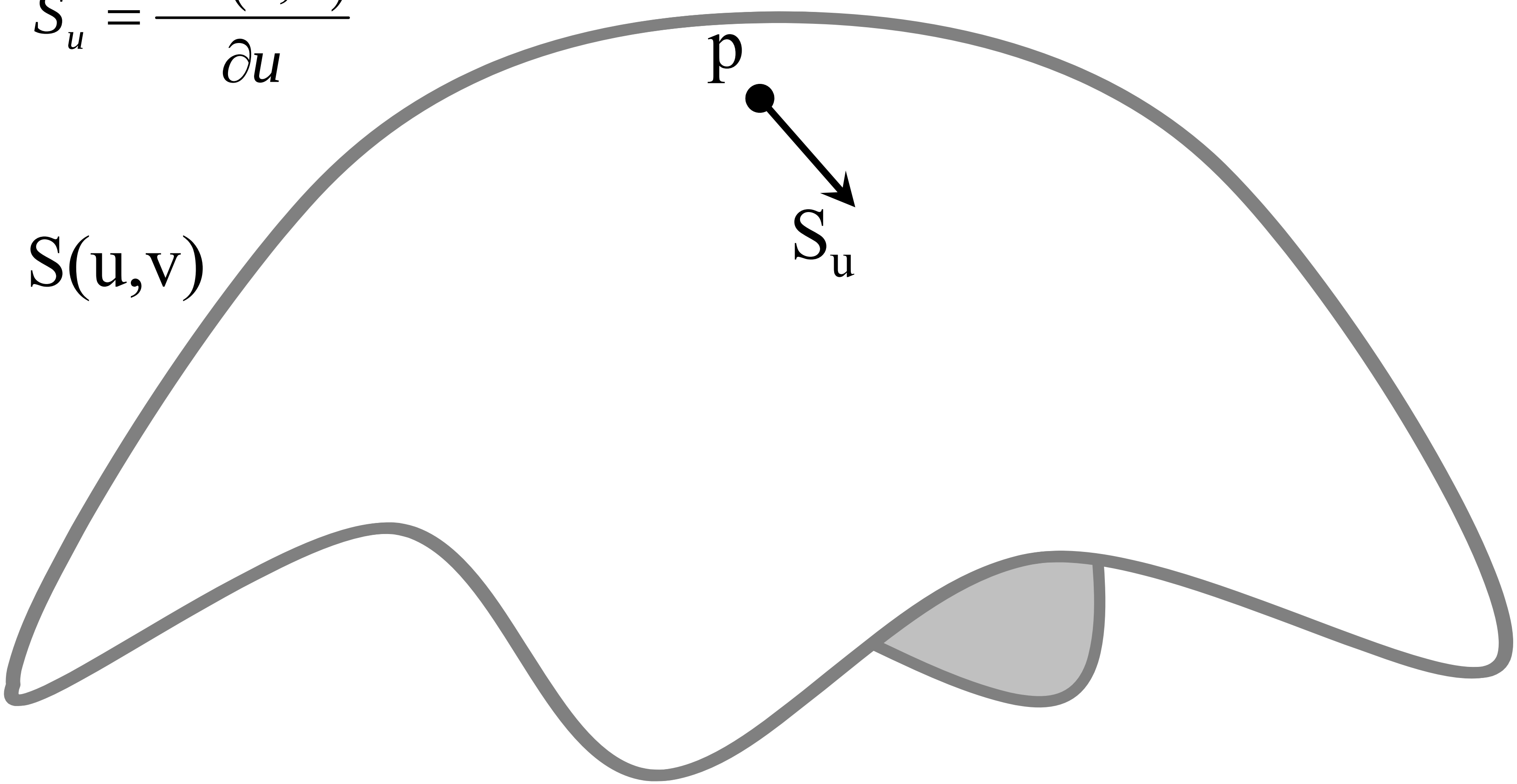
Point p on the surface at (u_0,v_0)



Differential Geometry of a Surface

Tangent S_u in the u direction

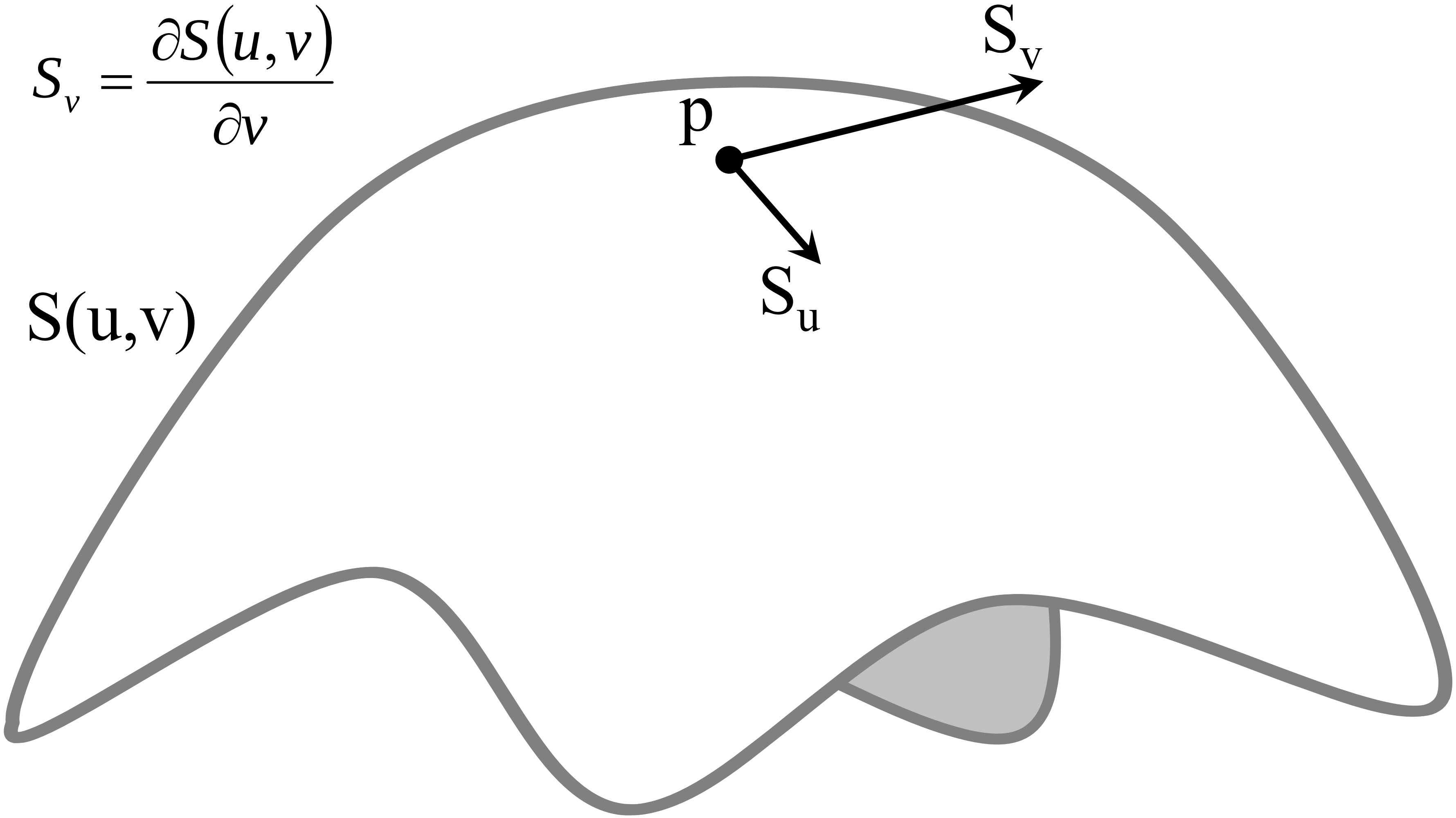
$$S_u = \frac{\partial S(u,v)}{\partial u}$$



Differential Geometry of a Surface

Tangent S_v in the v direction

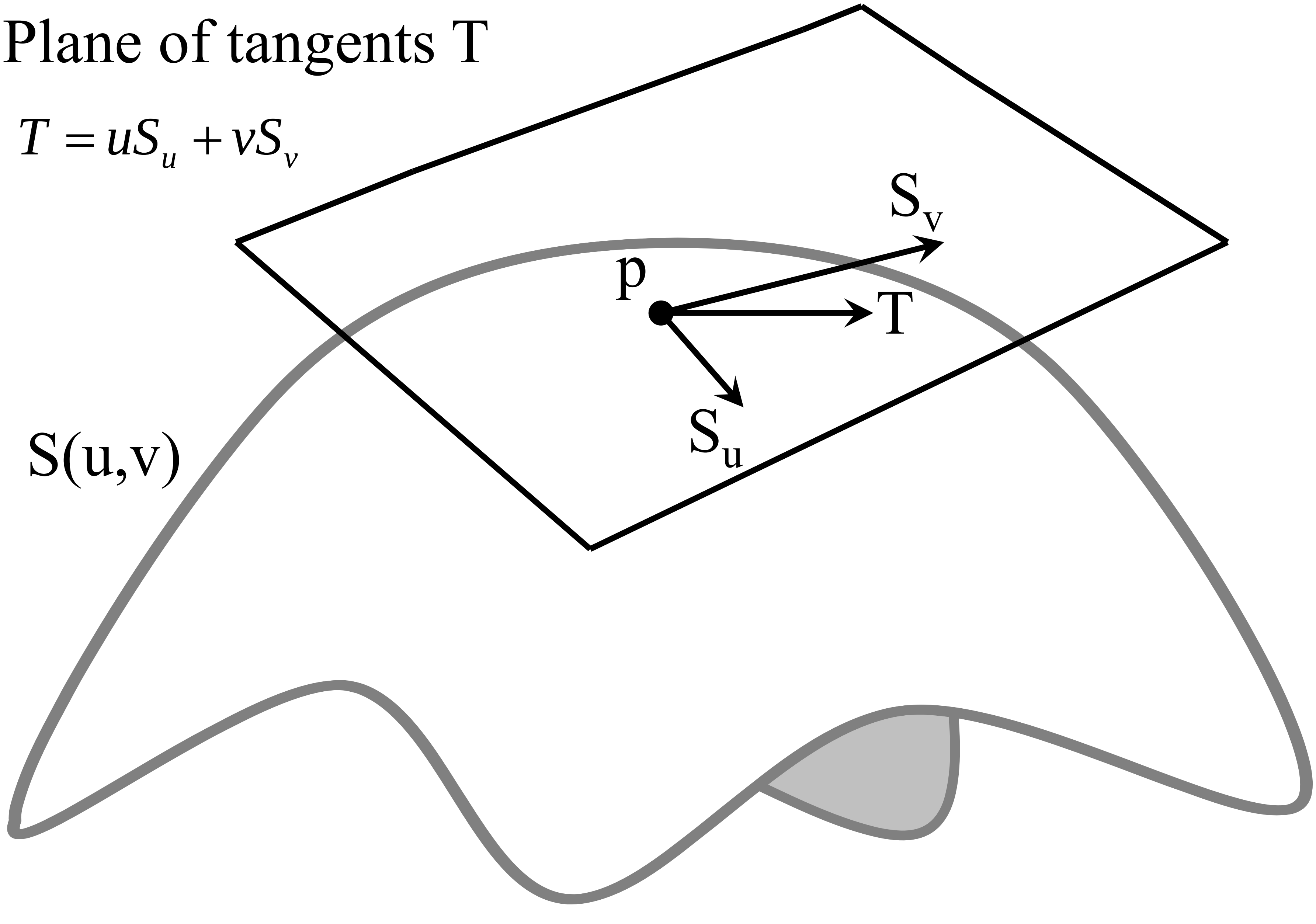
$$S_v = \frac{\partial S(u,v)}{\partial v}$$



Differential Geometry of a Surface

Plane of tangents T

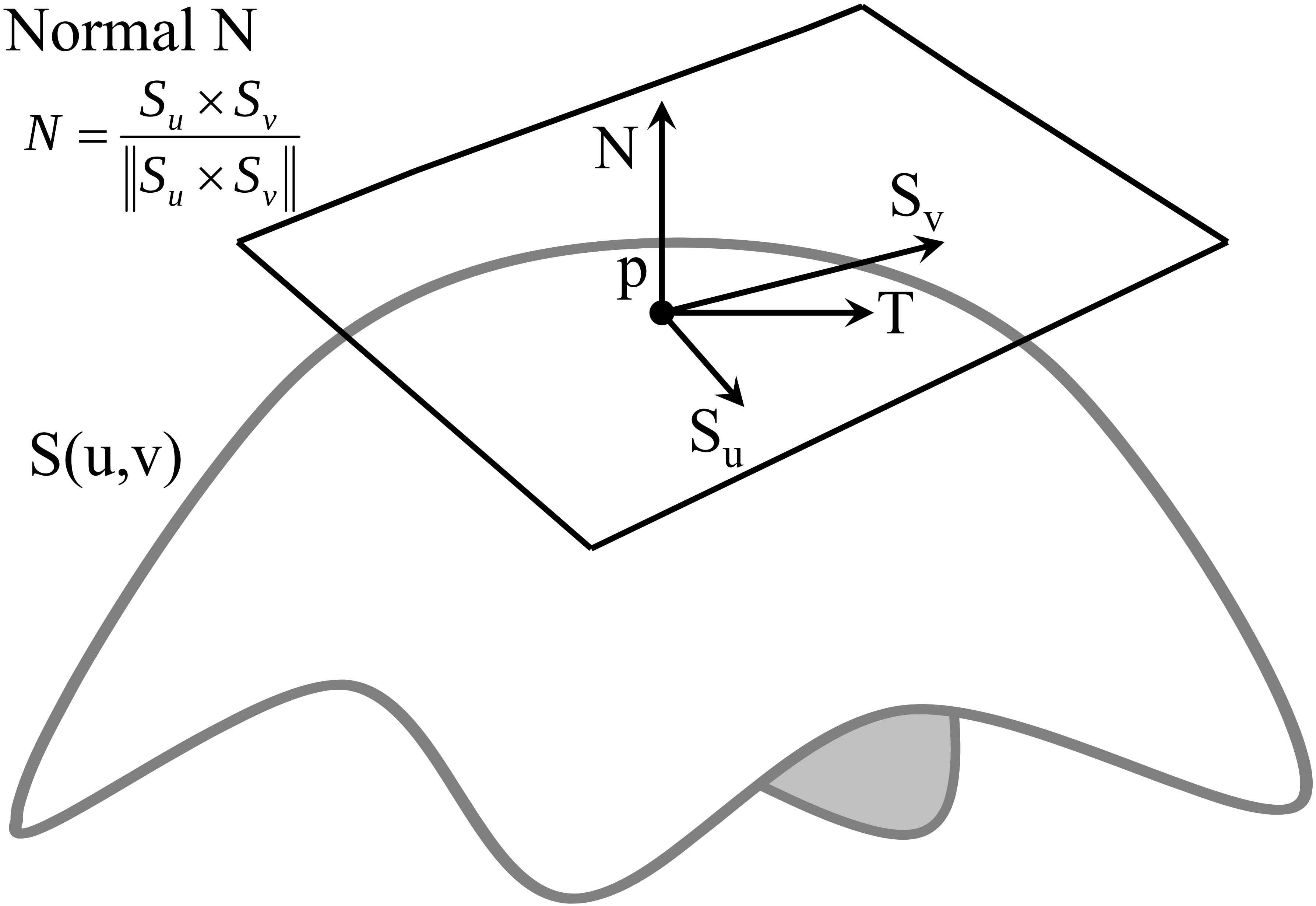
$$T = uS_u + vS_v$$



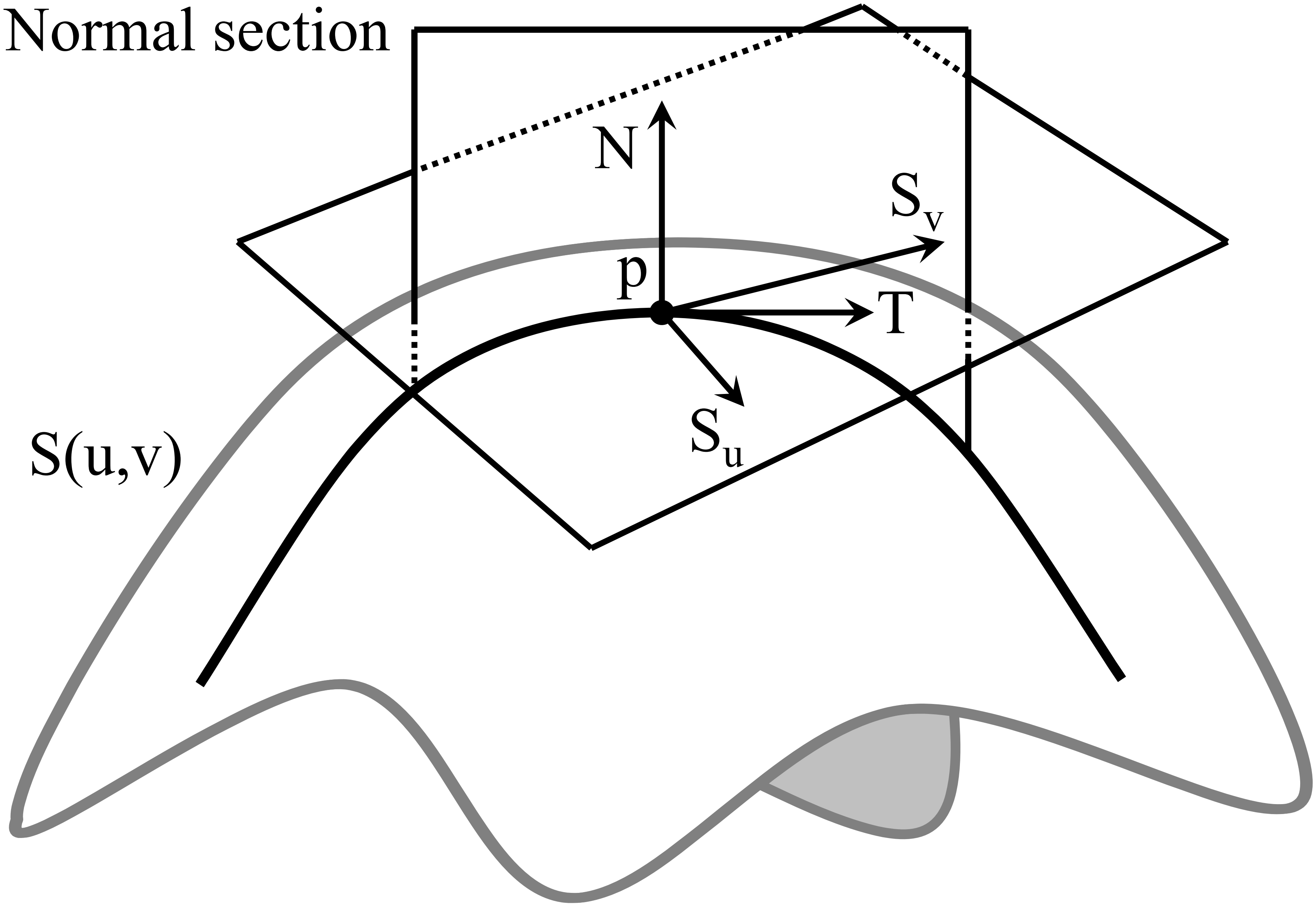
Differential Geometry of a Surface

Normal N

$$N = \frac{S_u \times S_v}{\left\| S_u \times S_v \right\|}$$

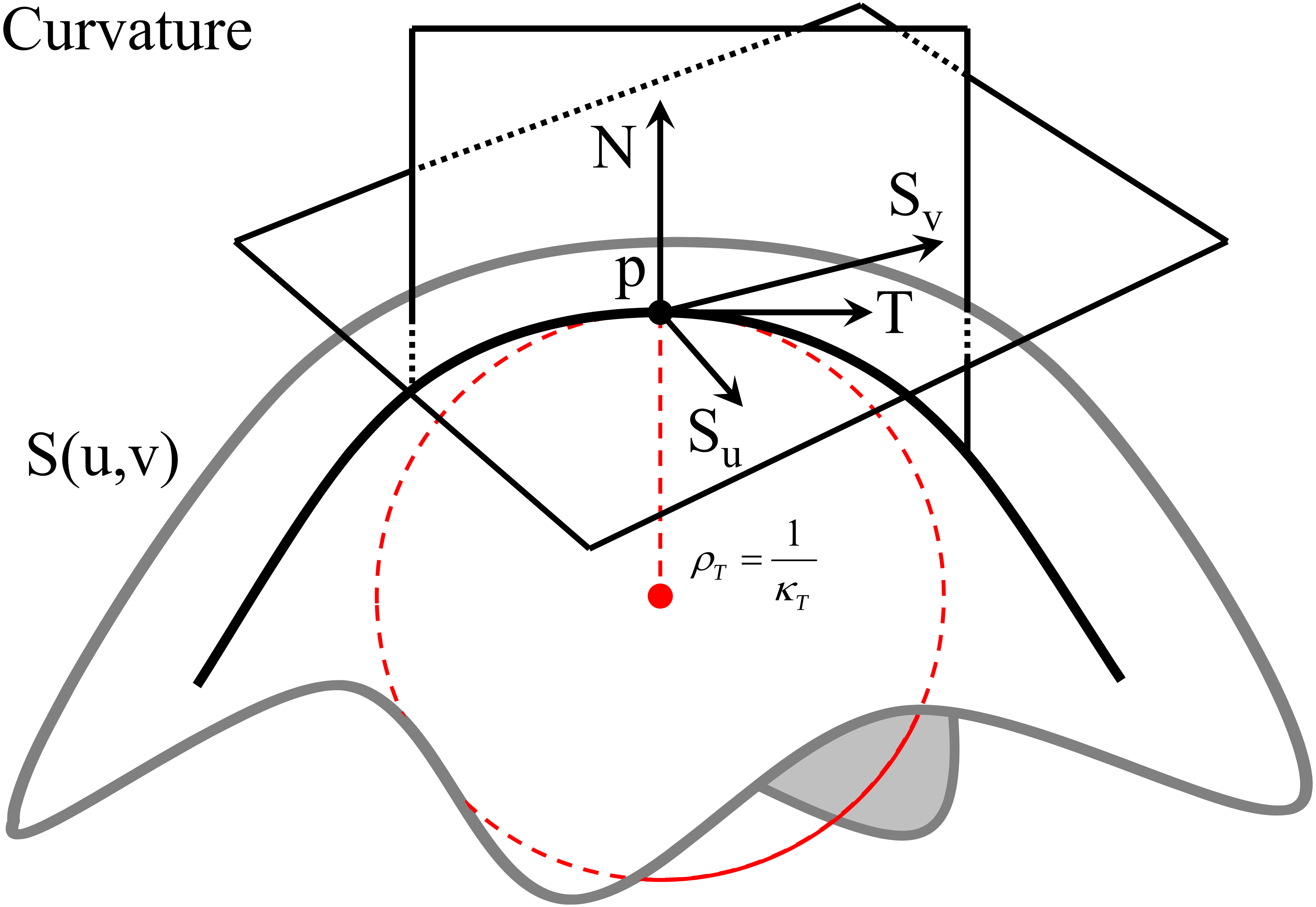


Differential Geometry of a Surface

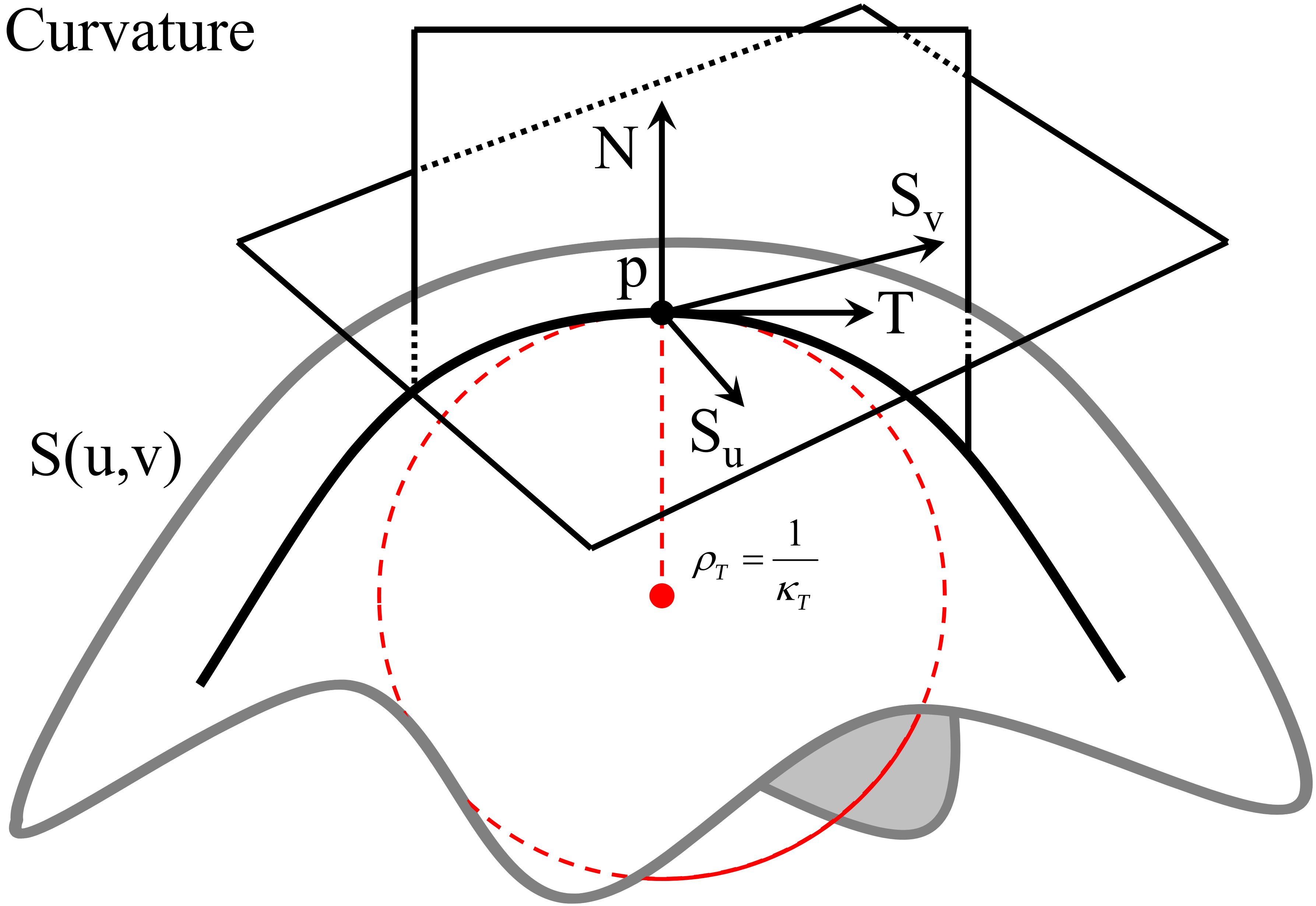


Differential Geometry of a Surface

Curvature

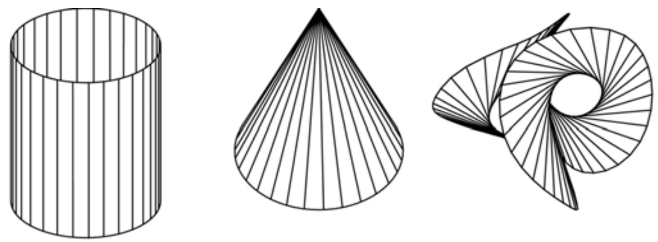


方向曲率：曲率是随着方向变化的

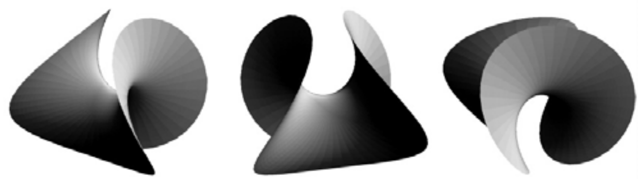


曲面的曲率

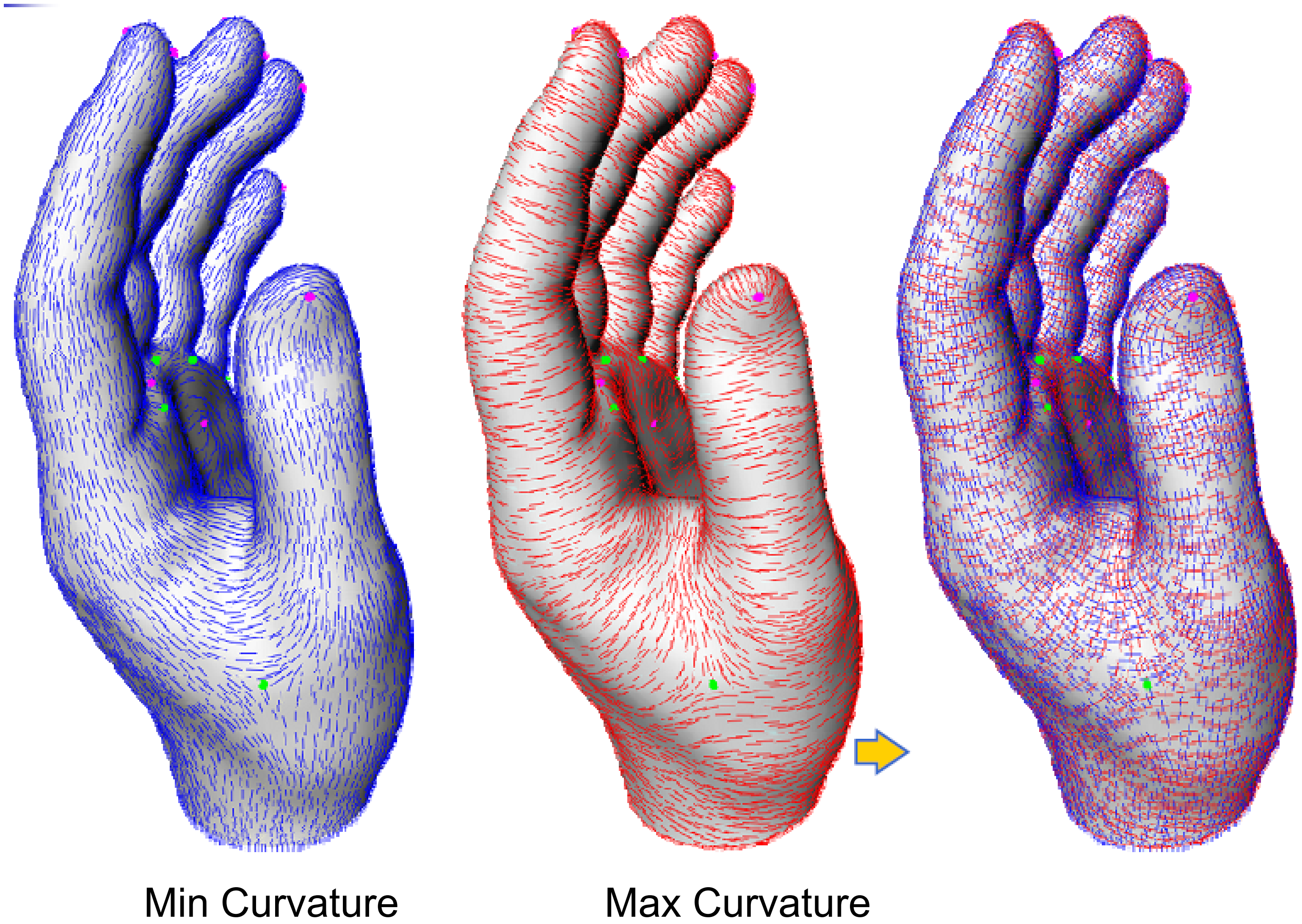
- 主曲率
  - 两个方向（正交）曲率：最大曲率κ_1和最小曲率κ_2
- 欧拉公式
  - 其他方向曲率$\kappa = \kappa_1 cos^2\theta + \kappa_2 sin^2\theta$
- 高斯曲率
  - $\kappa = \kappa_1 \kappa_2$
  - 等距变换不变量
  - 处处高斯曲率为0的曲面：可展曲面
  - 三类：柱面、锥面、切线面



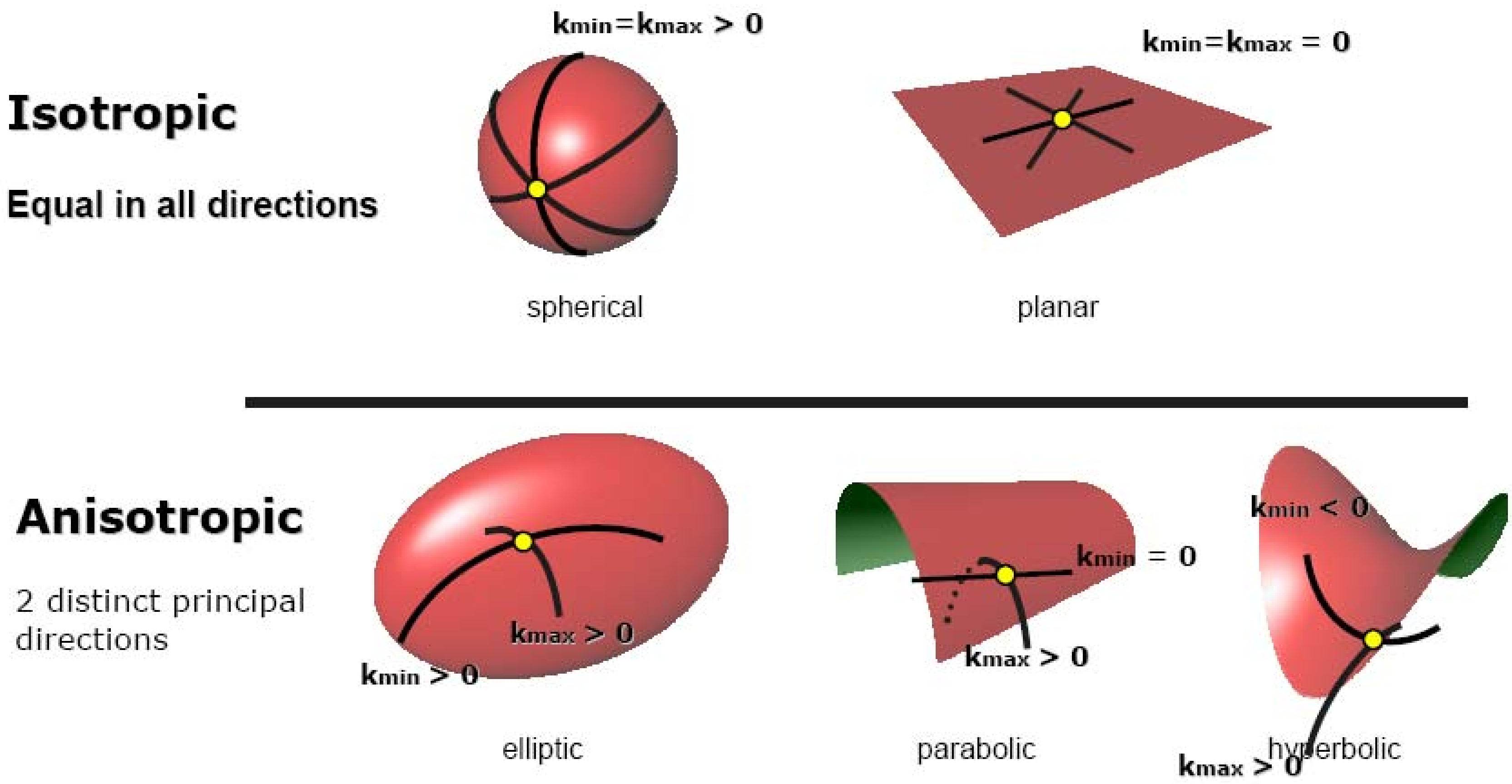
- 平均曲率
  - $\kappa = \frac{\kappa_1 + \kappa_2}{2}$
  - 处处平均曲率为0的曲面：极小曲面



Principal Directions

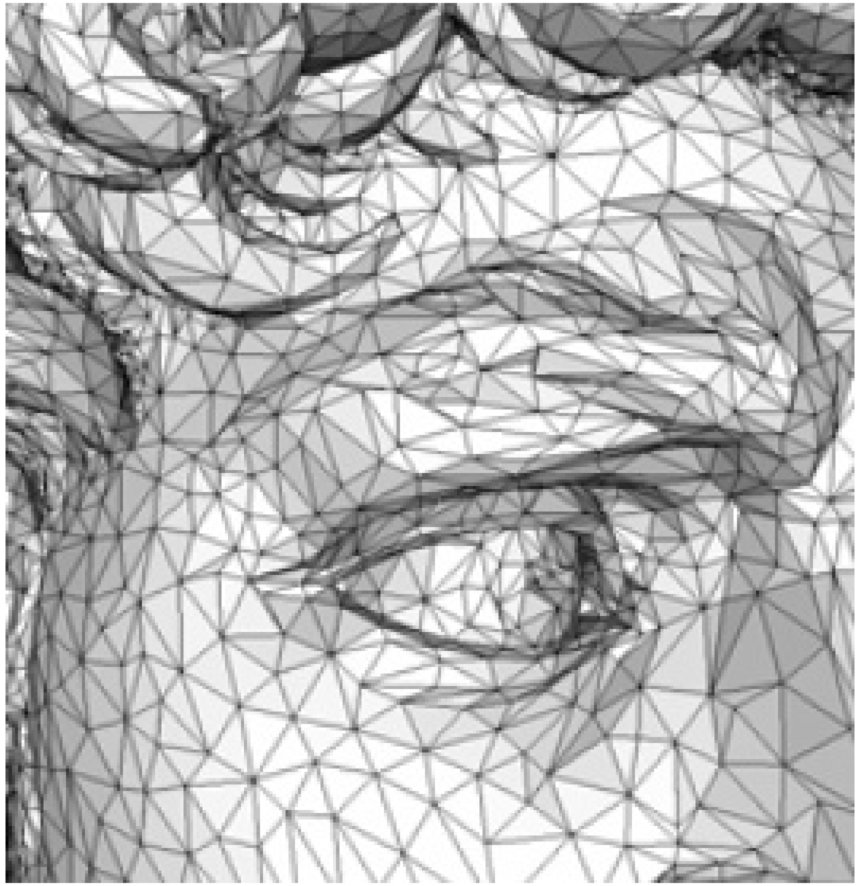


Surface Curvature



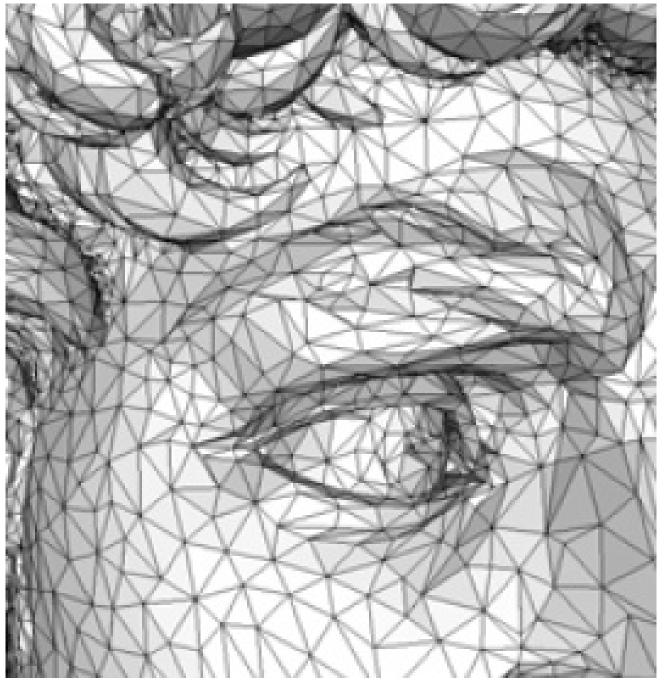
离散微分几何

三角网格曲面的光滑性?



However, meshes are only C^0

- Meshes are piecewise linear surfaces
  - Infinitely continuous on triangles
  - C^0 at edges and vertices



Discrete Differential Geometry

- How to apply the traditional differential geometry on discrete mesh surfaces?
  - Normal estimation
  - Curvature estimation
  - Principal curvature directions
  - …

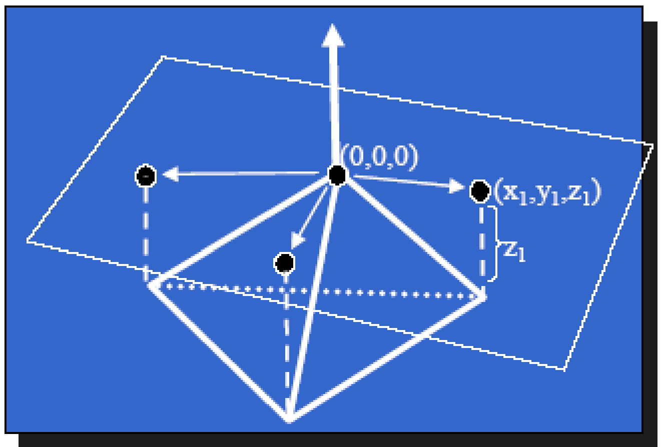
Estimation of Differential Measures

- Approximate the (unknown) underlying surface
  - Continuous approximation
    - Approximate the surface & compute continuous differential measures (normal, curvature)
  - Discrete approximation
    - Approximate differential measures for mesh

Continuous Approximation

Quadratic Approximation

- Approximate surface by quadric
- At each mesh vertex (use surrounding triangles)
  - Compute normal at vertex
    - Typically average face normals
  - Compute tangent plane & local coordinate system
    - (node = (0,0,0))
  - For each neighbor vertex compute location in local system
    - relative to node and tangent plane



Quadratic Approximation (2)

- Find quadric function approximating vertices

$$F(x, y, z) = ax^2 + bxy + cy^2 - z = 0$$

- To find coefficients use least squares fit

$$\min \sum_i (ax_i^2 + bx_iy_i + cy_i^2 - z_i)$$

Quadratic Approximation (3)

Finding the quadric function approximating points
$F(x,y,z) = ax^2 + bxy + cy^2 - z = 0$

To find coefficients use least square fit to find minimum: $min\sum_i \left(ax_i^2 + bx_iy_i + cy_i^2 - z_i\right)$

$$\begin{pmatrix} x_1^2 & x_1y_1 & y_1^2 \\ \dots & \dots & \dots \\ x_n^2 & x_ny_n & y_n^2 \end{pmatrix}\begin{pmatrix} a \\ b \\ c \end{pmatrix} = \begin{pmatrix} z_1 \\ \dots \\ z_n \end{pmatrix} \quad A = \begin{pmatrix} x_1^2 & x_1y_1 & y_1^2 \\ \dots & \dots & \dots \\ x_n^2 & x_ny_n & y_n^2 \end{pmatrix},\ X = \begin{pmatrix} a \\ b \\ c \end{pmatrix},\ b = \begin{pmatrix} z_1 \\ \dots \\ z_n \end{pmatrix}$$

Approximation can be found by: $\tilde{X} = (A^T A)^{-1} A^T b$

Quadratic Approximation (4)

- Given surface F its principal curvatures k_{min} and k_{max} are real roots of:

$$\boxed{k^2 - (a + c)k + ac - b^2 = 0}$$

- *Mean curvature*: $H = (k_{min} + k_{max})/2$

- *Gaussian curvature:* $K = k_{min} k_{max}$

Other approximation

- Cubic approximation
  - J. Goldfeather and V. Interrante. A novel cubic-order algorithm for approximating principal direction vectors. ACM Transactions on Graphics 23, 1 (2004), 45–63.
- Implicit surface approximation
  - Yutaka Ohtake et al. Multi-level partition of unity implicits. Siggraph 2003.
- Many others…

Discrete Approximation

Normal Estimation

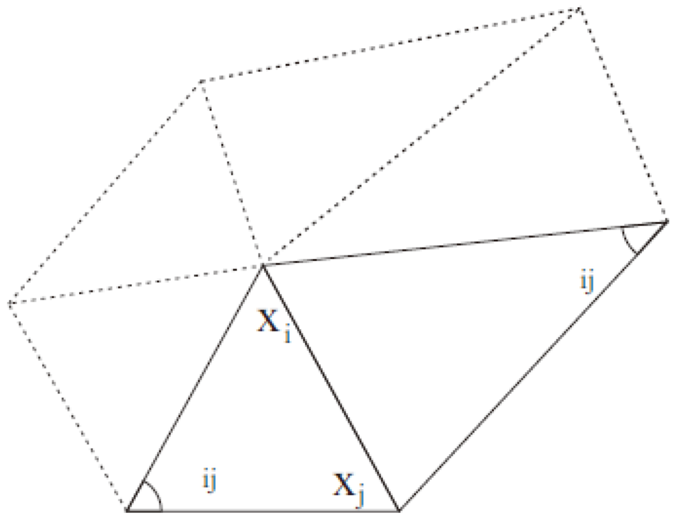
- Normal estimation on vertices
  - Defined for each face
  - Average face normals
    - Weighted: face areas, angles at vertex

- What happen at edges/creases?

Mean Curvature

- 由Laplace-Beltrami定理：

$$\boldsymbol{K}(\boldsymbol{x}_i) = \frac{1}{2\mathcal{A}_M} \sum_{j \in N_1(i)} (\cot \alpha_{ij} + \cot \beta_{ij})(\boldsymbol{x}_i - \boldsymbol{x}_j)$$

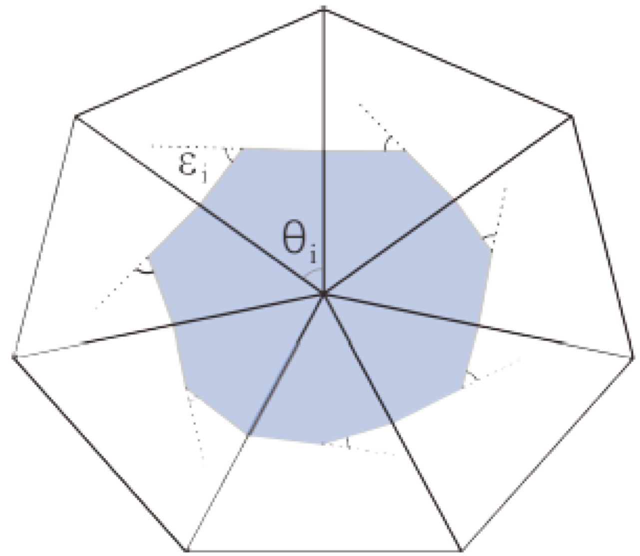


Gauss Curvature

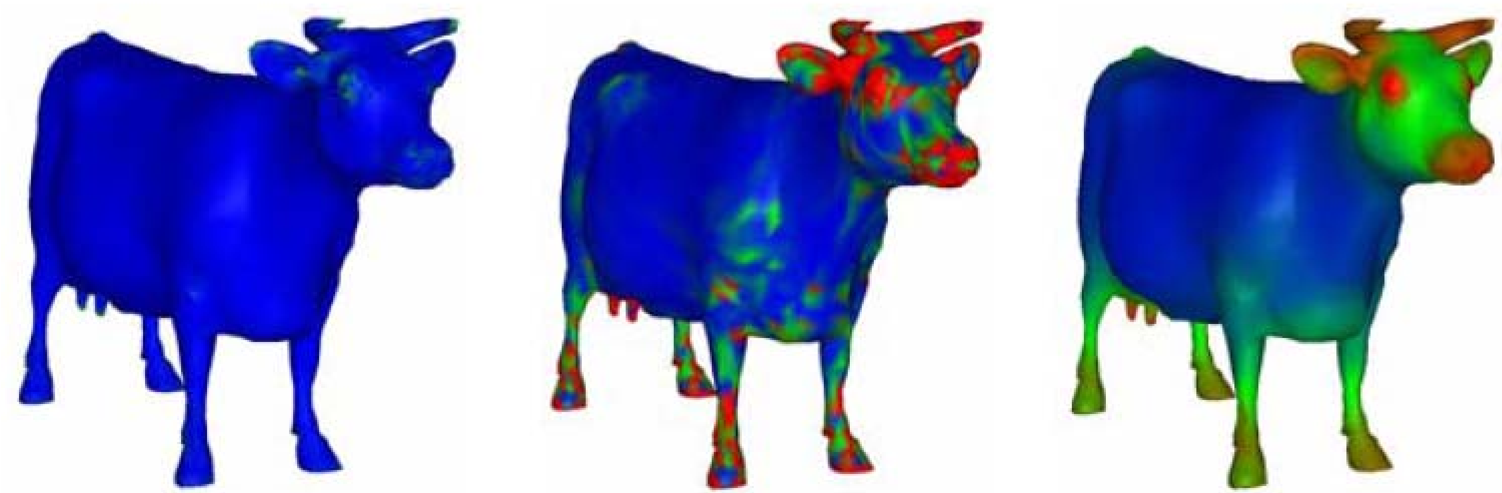
- 由Gauss-Bonnet定理：

$$\iint_{\mathcal{A}_M} \kappa_G dA = 2\pi - \sum_j \epsilon_j = 2\pi - \sum_{j=1}^{\#f} \theta_j$$

$$\kappa_G(\boldsymbol{x}_i) = (2\pi - \sum_{j=1}^{\#f} \theta_j)/\mathcal{A}_M$$

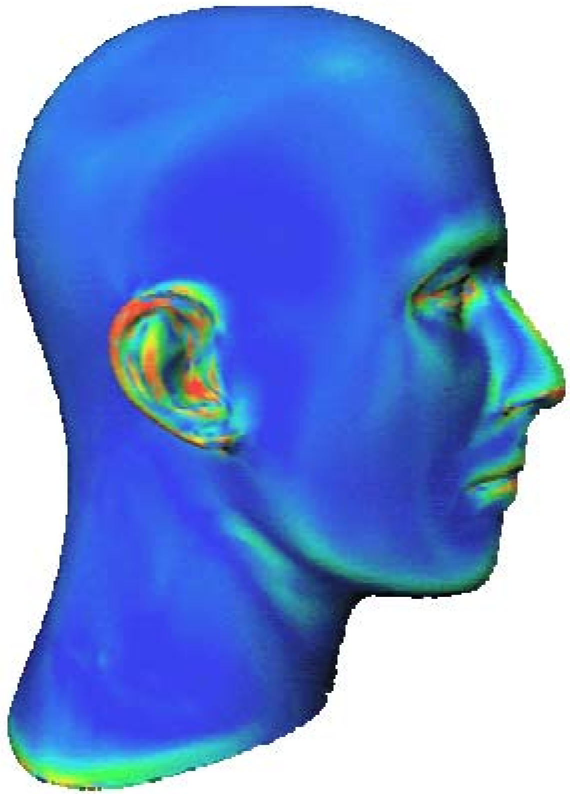


Gaussian Curvature Estimate – Example

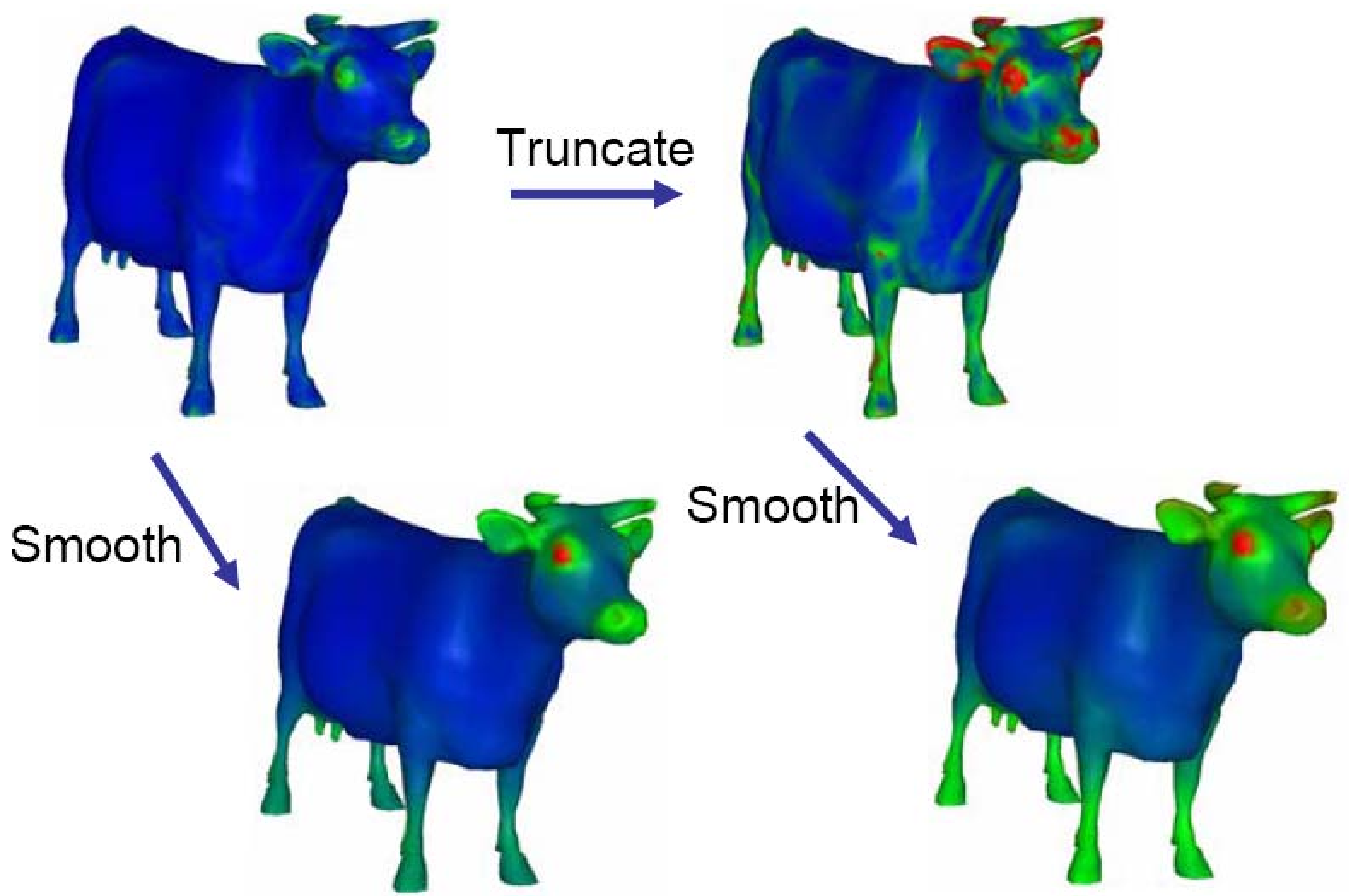


- Approximation always results in some noise
- Solution
  - Truncate extreme values
    - Can come for instance from division by very small area
  - Smooth
    - More later

Mean Curvature Estimate – Example



Mean Curvature



More…

- MEYER M., DESBRUN M., SCHRÖDER P., BARR A.: Discrete differential-geometry operators for triangulated 2-manifolds. In Visualization and Mathematics III, Hege H.-C., Polthier K., (Eds.). Springer, 2003, pp. 35–58. (PDF)

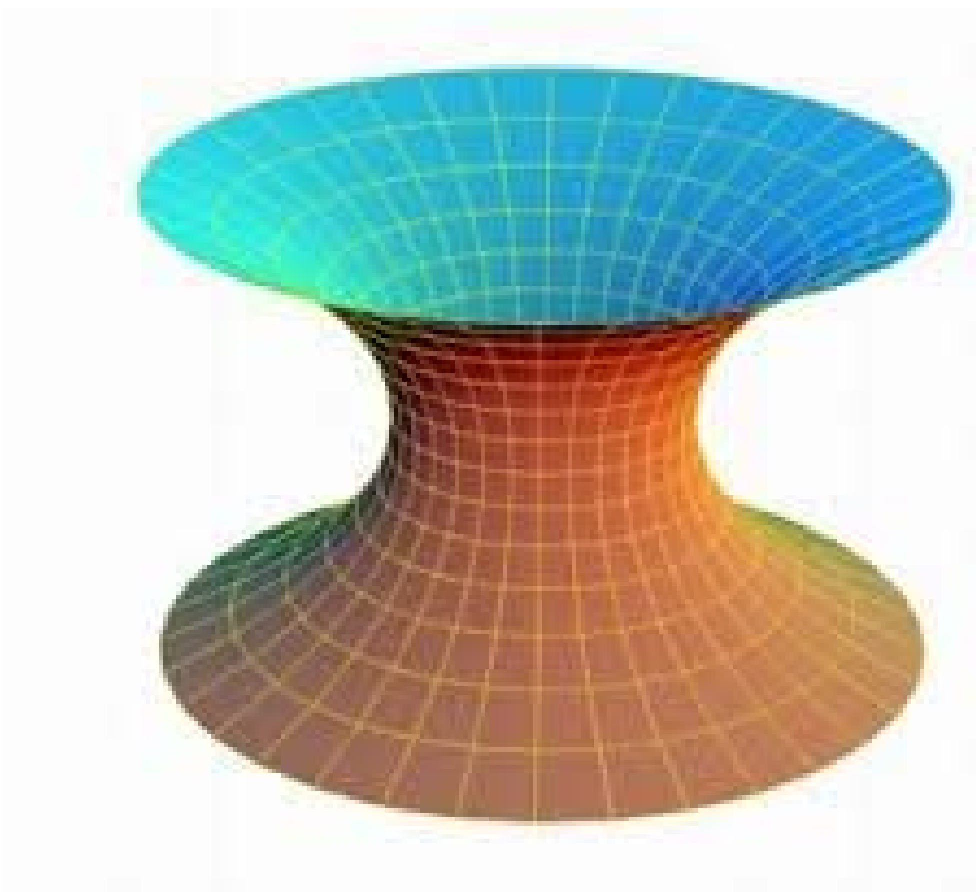
References

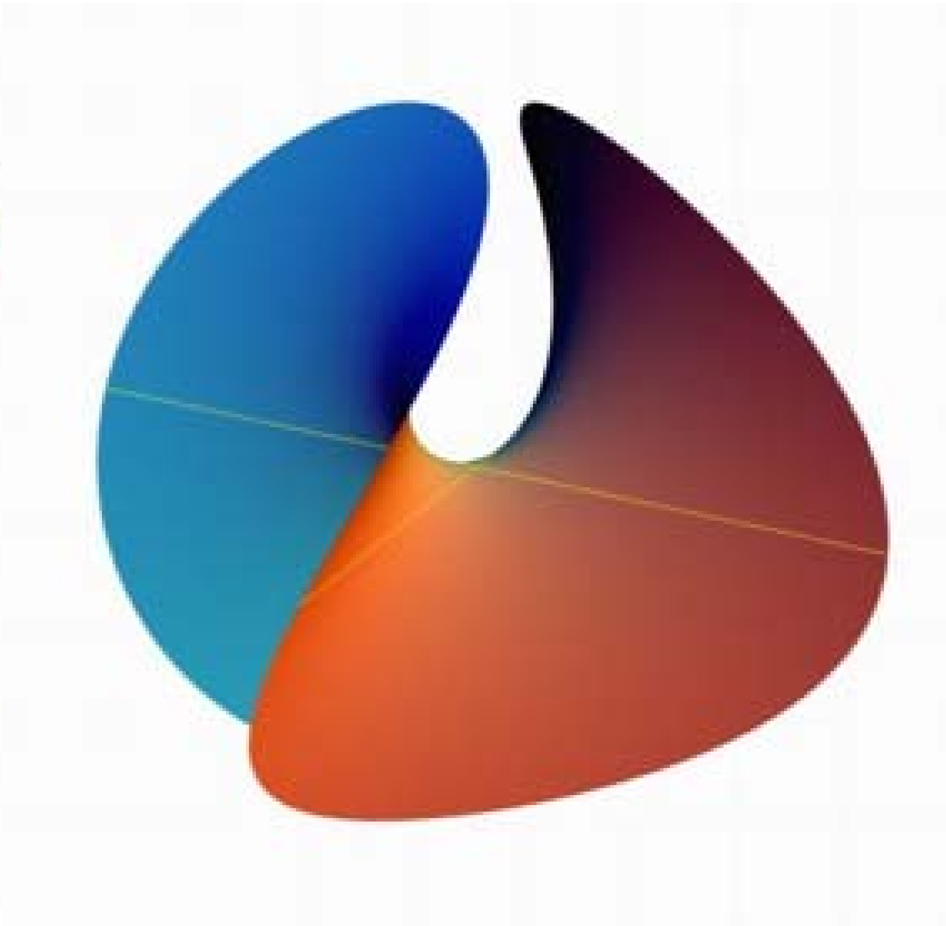
- TAUBIN G.: Estimating the tensor of curvature of a surface from a polyhedral approximation. In Proc. International Conference on Computer Vision (1995), pp. 902–907.
- MEYER M., DESBRUN M., SCHRÖDER P., BARR A.: Discrete differential-geometry operators for triangulated 2-manifolds. In Visualization and Mathematics III, Hege H.-C., Polthier K., (Eds.). Springer, 2003, pp. 35–58.
- CAZALS F., POUGET M.: Estimating differential quantities using polynomial fitting of osculating jets. In Eurographics Symposium on Geometry Processing (2003), pp. 177–187.
- COHEN-STEINER D., MORVAN J.: Restricted delaunay triangulations and normal cycle. In Proc. ACM Symposium on Computational Geometry (2003), pp. 312–321.
- GOLDFEATHER J., INTERRANTE V.: A novel cubic-order algorithm for approximating principal direction vectors. ACM Transactions on Graphics 23, 1 (2004), 45–63.
- MARTIN R. R.: Estimation of principal curvatures from range data. International Journal of Shape Modeling 4, 1 (1998), 99–109.
- OHTAKE Y., BELYAEV A., SEIDEL H.-P.: Ridge-valley lines on meshes via implicit surface fitting. ACM Transactions on Graphics 23, 3 (2004), 609–612. (Proc. SIGGRAPH'2004).
- PAGE D., SUN Y., KOSCHAN A., PAIK J., ABIDI M.: Normal vector voting: Crease detection and curvature extimation on large, noisy meshes. Graphical Models 64, 3-4 (2002), 199–229.

极小曲面

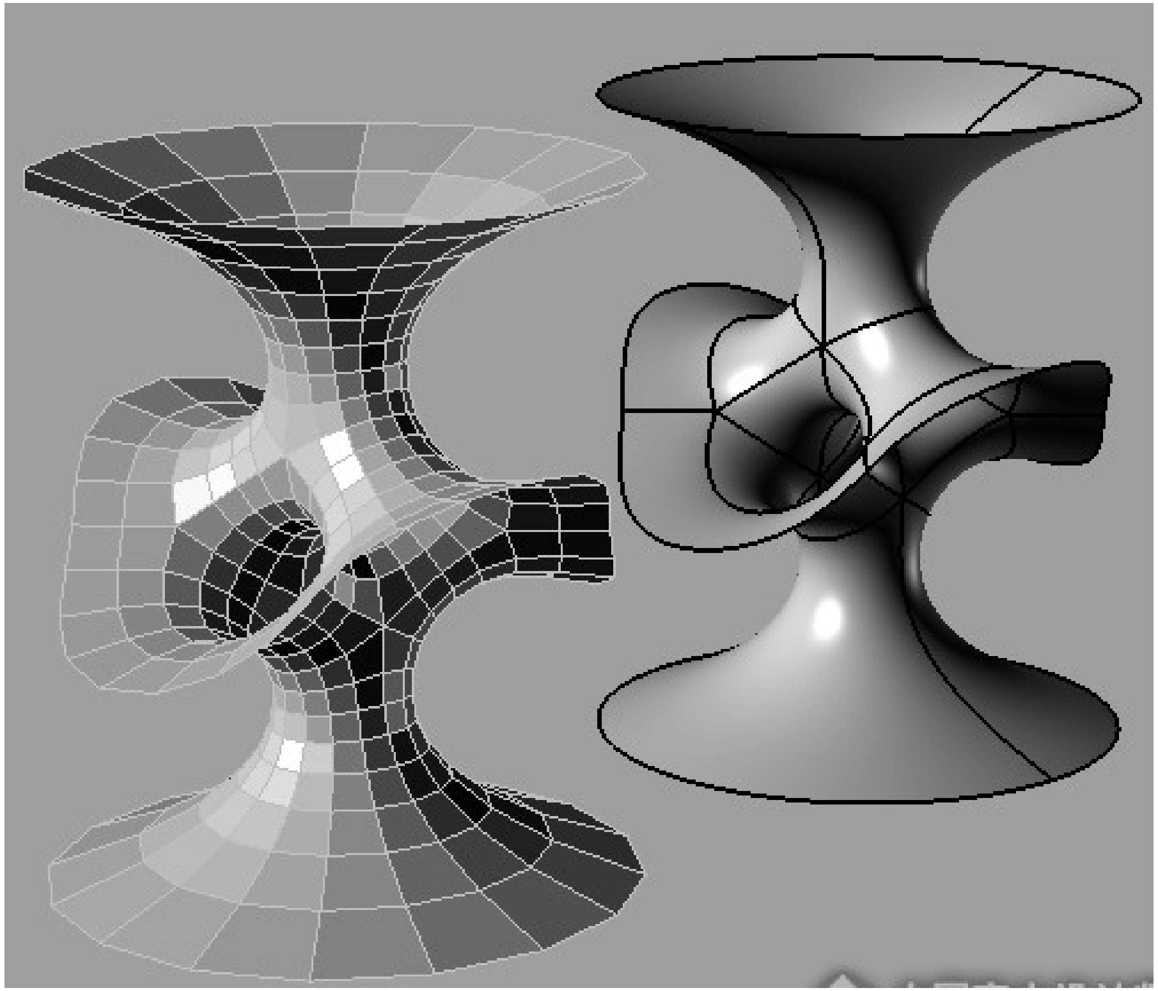
极小曲面

- 平均曲率处处为0的曲面

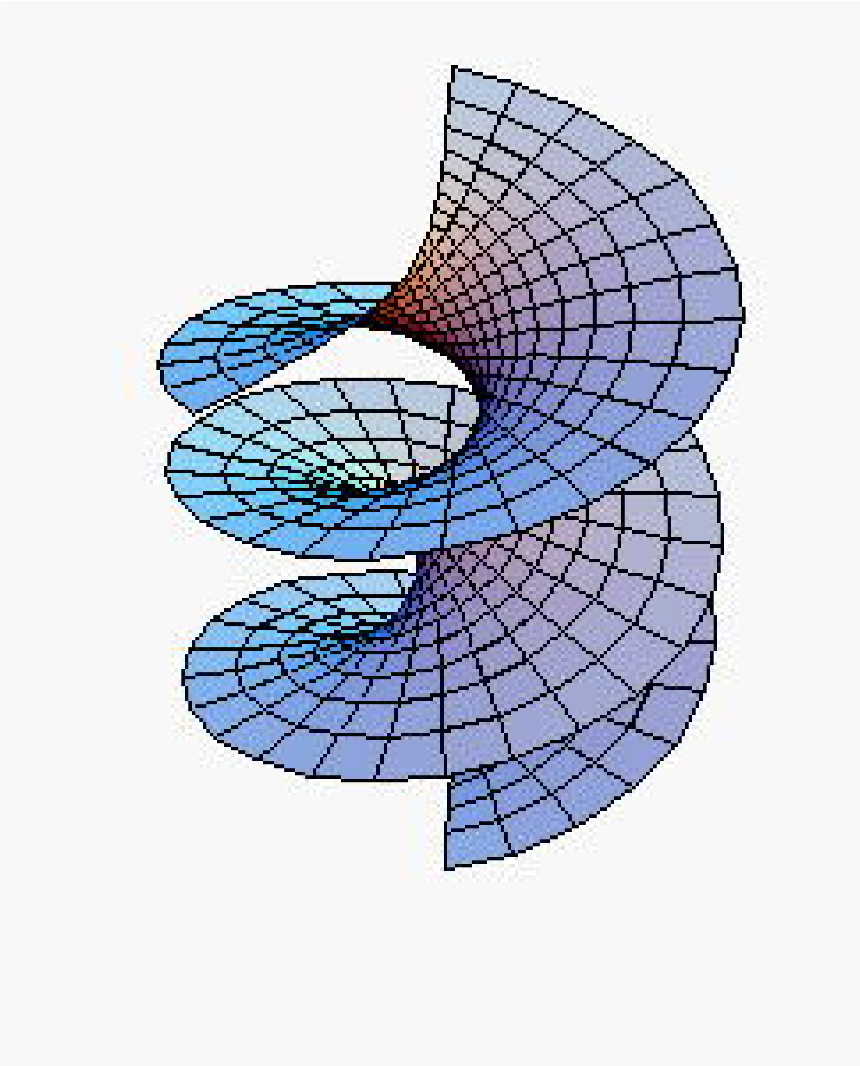


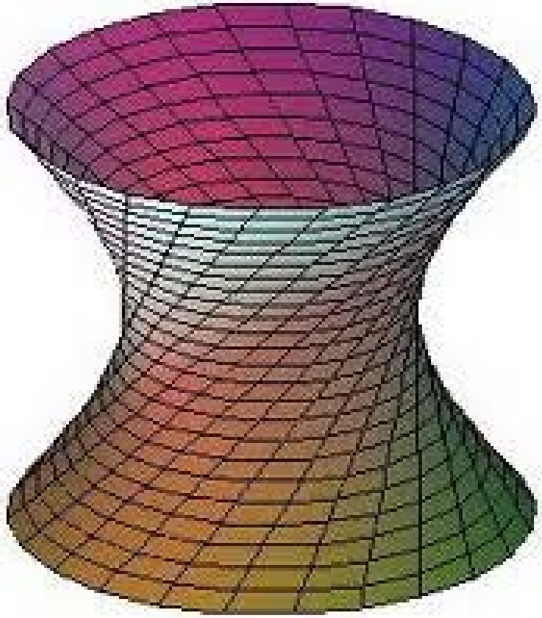


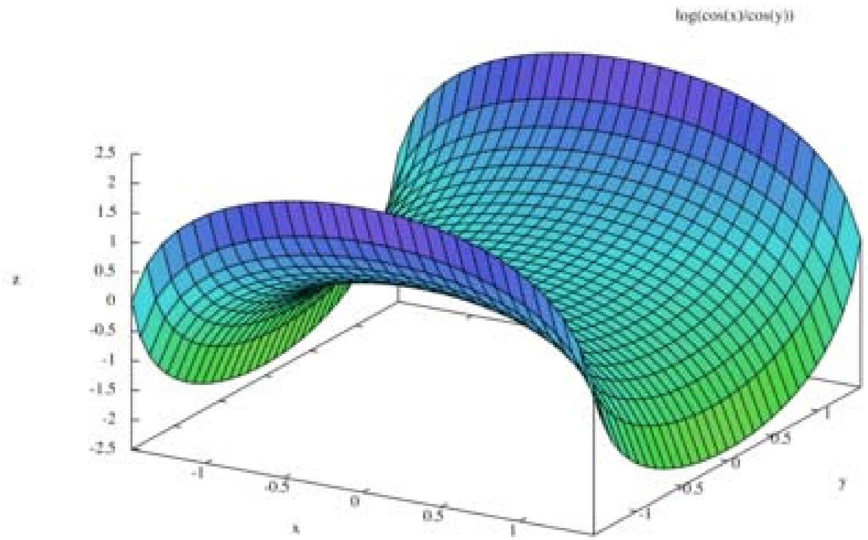
极小曲面



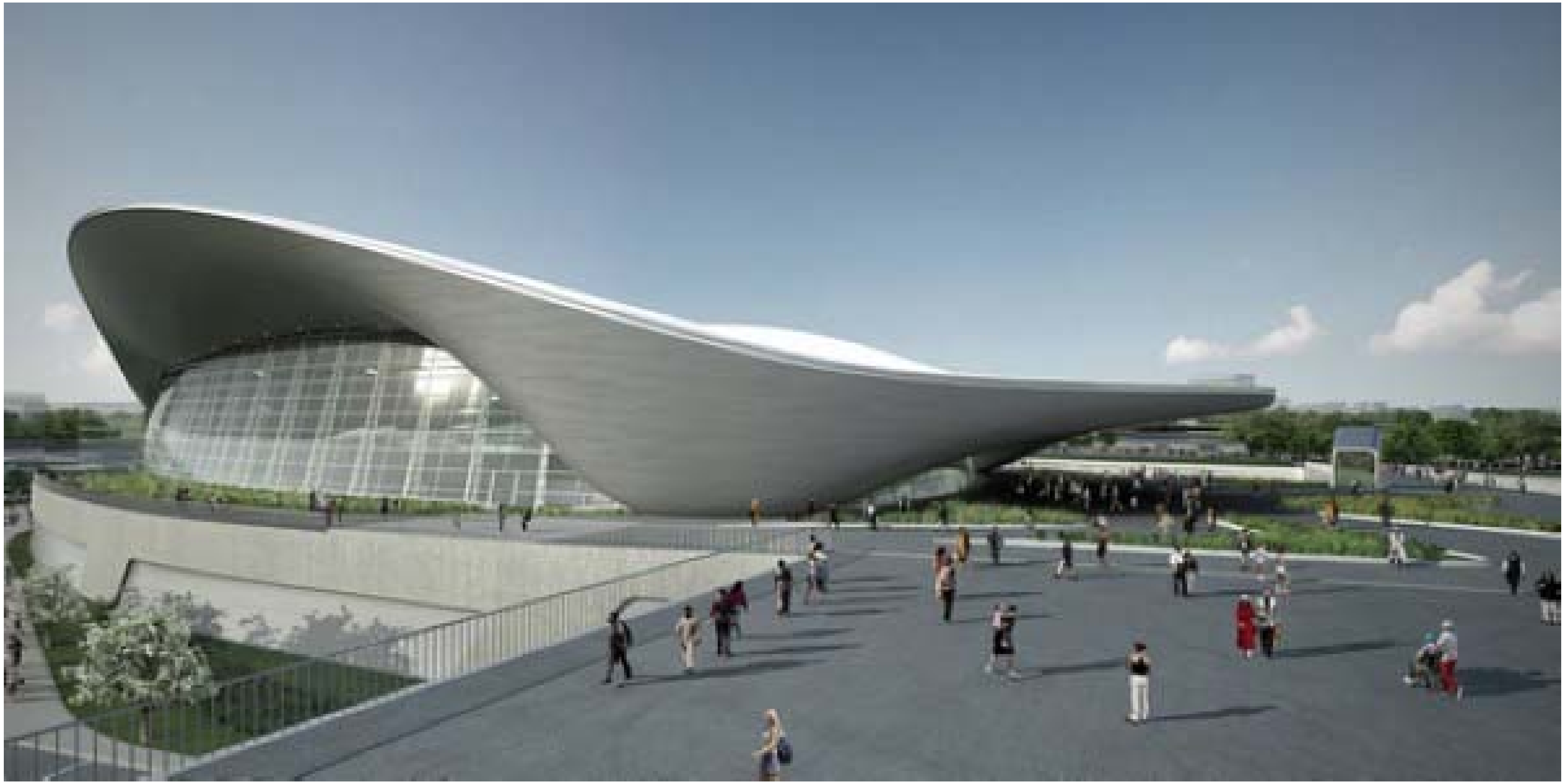
极小曲面的例子







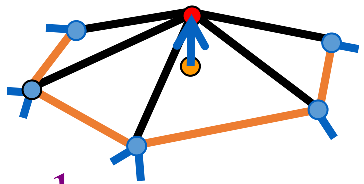
建筑中的极小曲面：膜结构

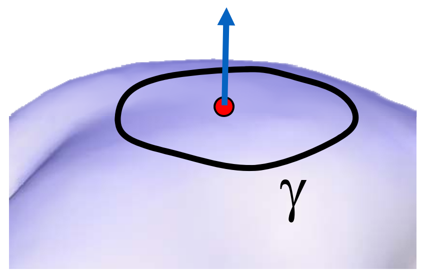


极小曲面及平均曲率流

- 平均曲率处处为0

$$\boxed{H(\mathbf{v}_i)=0, \qquad \forall i}$$





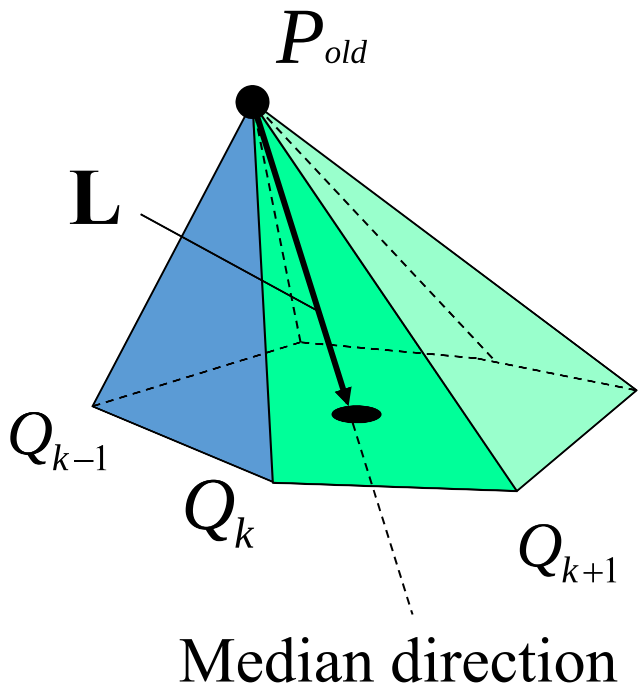
$$\boldsymbol{\delta}_i=\frac{1}{d_i}\sum_{\mathbf{v}\in N(i)}(\mathbf{v}_i-\mathbf{v})$$

$$\frac{1}{len(\gamma)}\int_{\mathbf{v}\in\gamma}(\mathbf{v}_i-\mathbf{v})\,ds$$

$$\lim_{len(\gamma)\to 0}\frac{1}{len(\gamma)}\int_{\mathbf{v}\in\gamma}(\mathbf{v}_i-\mathbf{v})\,ds = H(\mathbf{v}_i)\mathbf{n}_i$$

Laplace Operator (Umbrella Operator)

$$L(P) = \frac{1}{n}\sum_{i=1}^{n}\overrightarrow{PQ_i} = \frac{1}{n}\sum_{i=1}^{n}Q_i - P$$

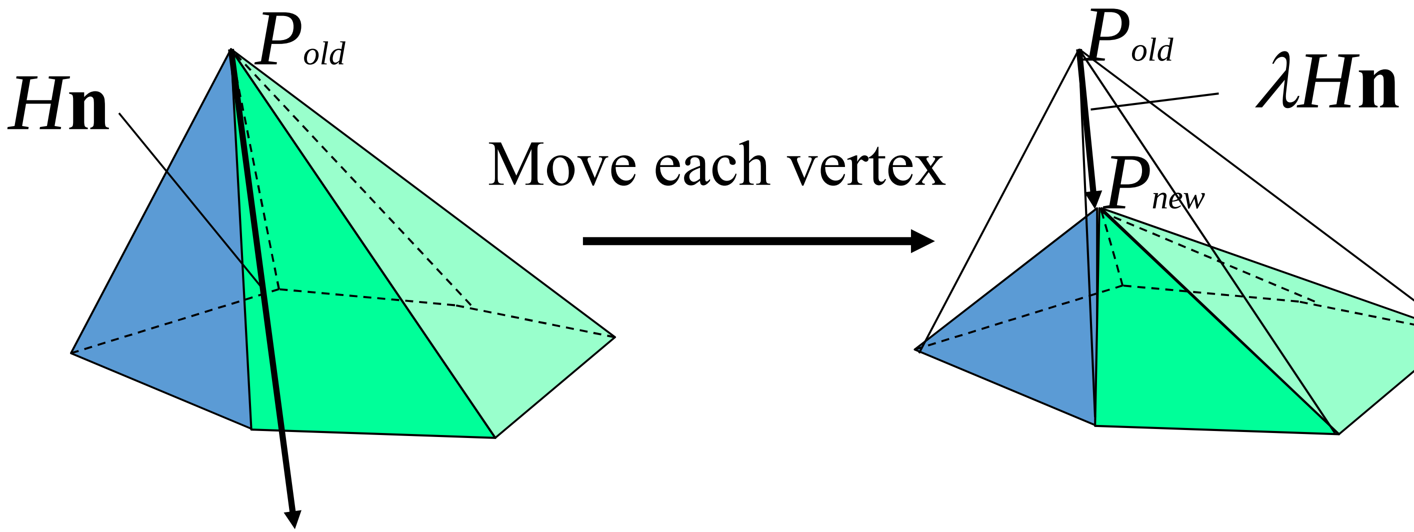


离散平均曲率流

$$P_{new} \leftarrow P_{old} + \lambda \boxed{H(P_{old})}\,\boxed{\mathbf{n}(P_{old})}$$

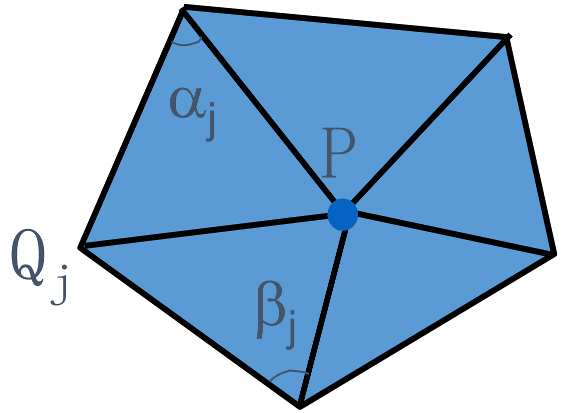
Speed = discrete mean curvature

Direction = normal



Discrete Mean Curvature

$$H\mathbf{n} = \frac{\nabla_P \mathrm{A}}{2\mathrm{A}}$$



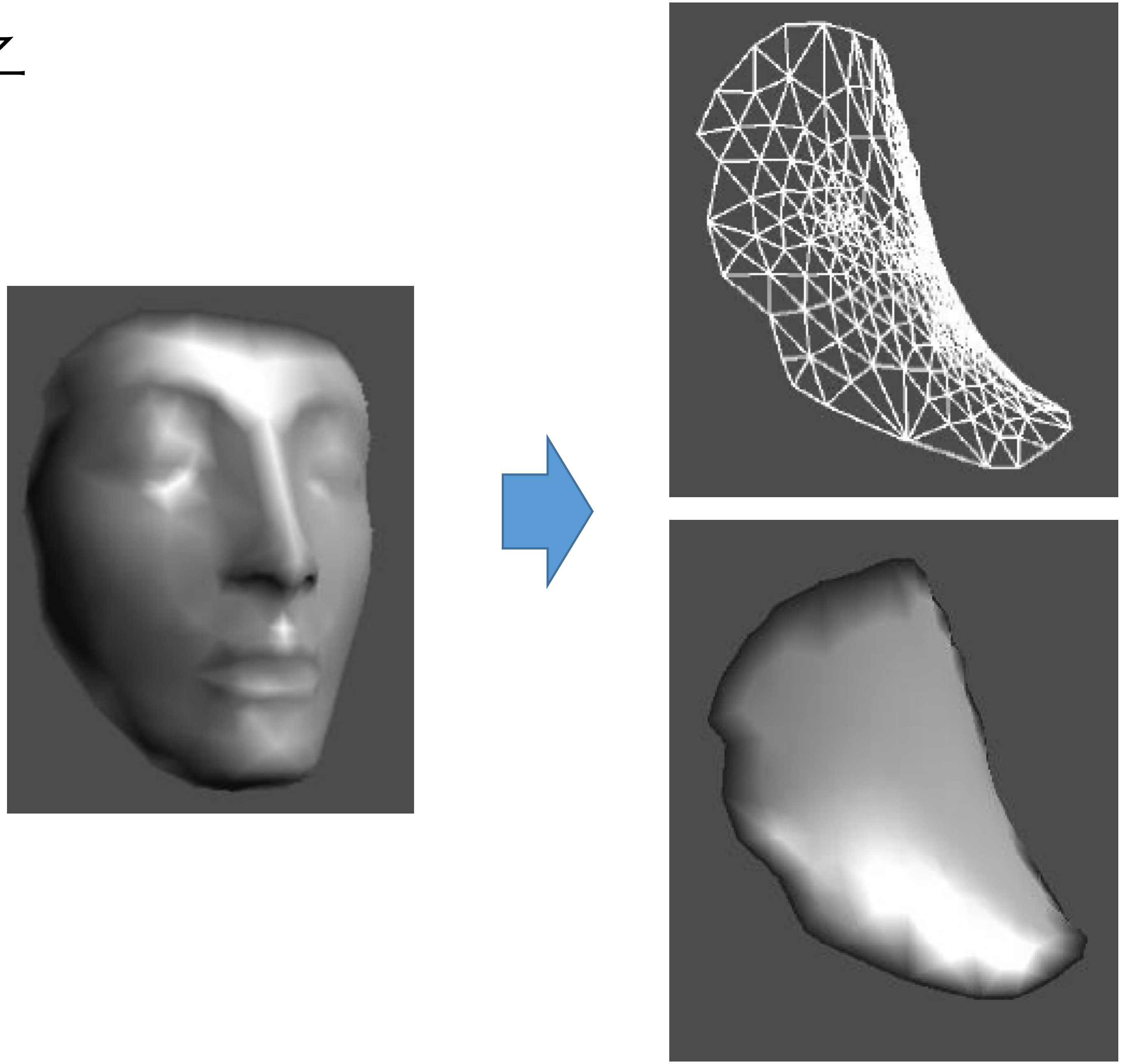
$$H\mathbf{n} = \frac{1}{4\mathrm{A}} \sum_j (\cot \alpha_j + \cot \beta_j)(\mathbf{P} - \mathbf{Q}_j)$$

离散极小曲面的局部迭代法

- 找到边界
- 固定边界顶点
- 对每个内部顶点
  - 找顶点1-邻域
  - 更新其坐标
- 迭代
- 更新所有顶点法向

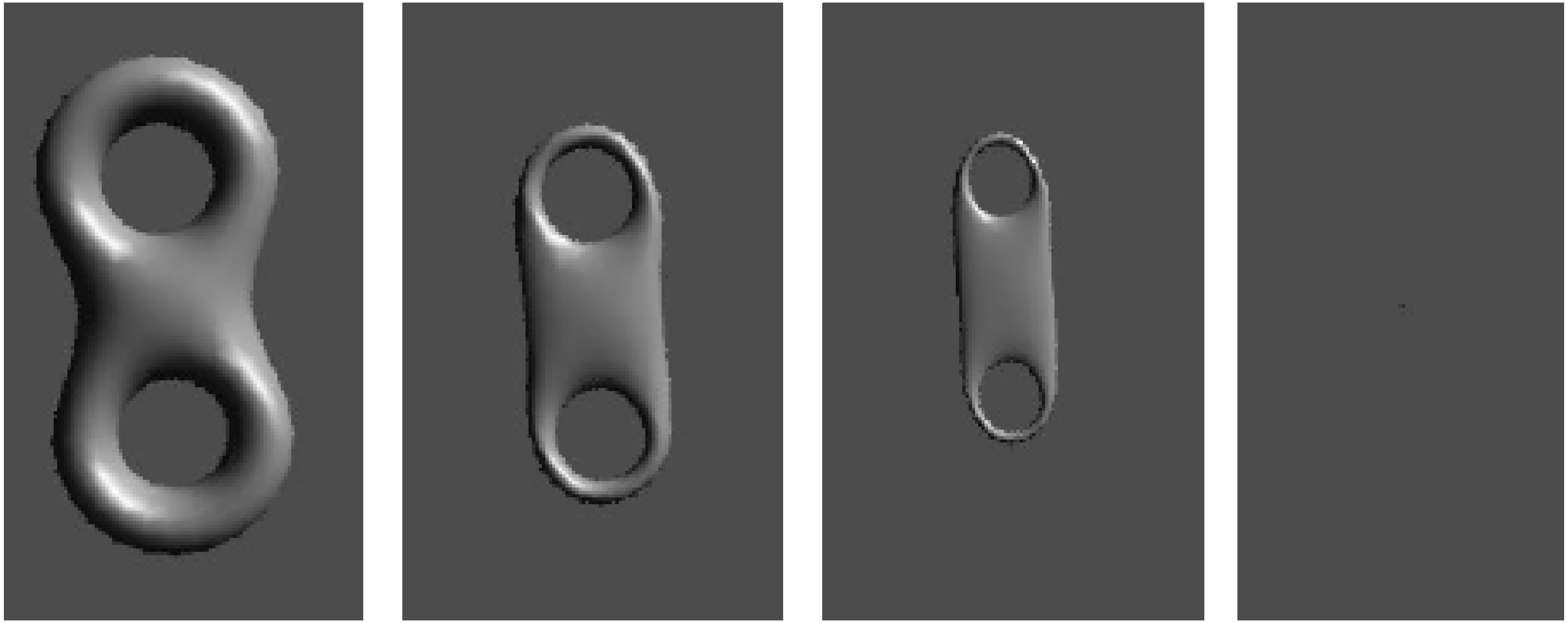
- 【注】
  - 只能对非封闭曲面（带一条边界）操作
  - 更新坐标需要用老的顶点坐标
  - 尝试试验不同的参数λ（$\lambda = 0.1$）

例子



封闭曲面

- 不固定任何顶点
- 迭代结果如何?

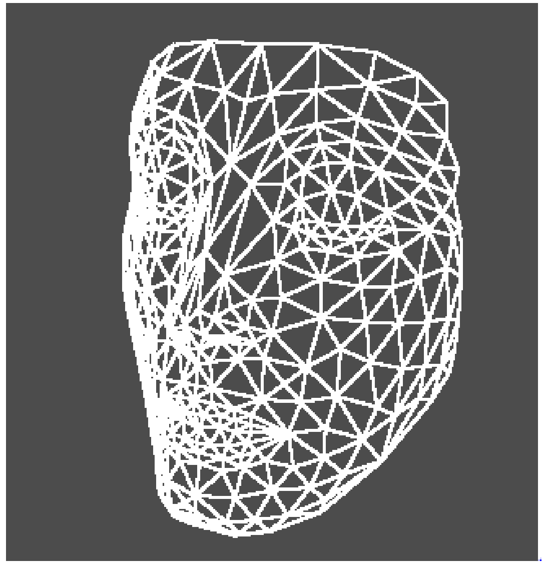


作业6

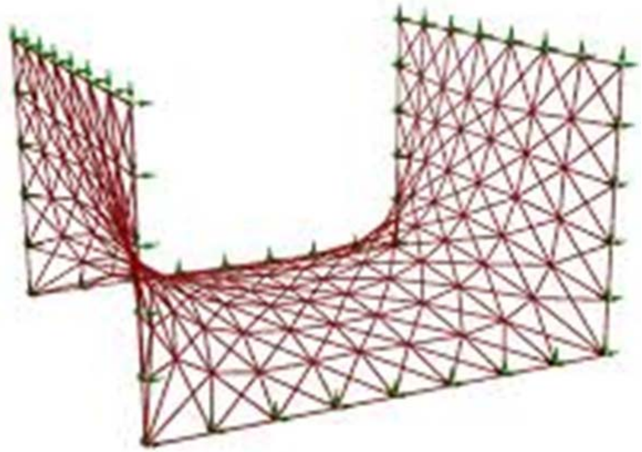
- 任务：实现极小曲面的局部法
  - 寻找非封闭三角网格曲面的边界
  - 每个顶点更新坐标
  - 迭代给定次数 或 迭代至收敛
- 目的
  - 学习三角网格的半边数据结构及操作
- 框架：
  - Utopia （推荐）
  - 其他
- 【可选】计算三角网格顶点的离散高斯曲率和平均曲率并用颜色进行可视化
- Deadline：2020 年12 月5日晚

附加：如何构造曲面边界？

- 3D空间封闭曲线
  - 已知的非封闭曲面
  - 自己构造：平面曲线变形

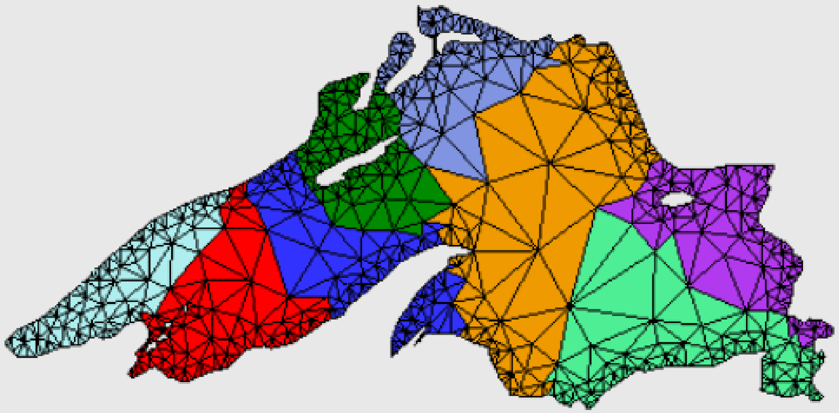


- 初始网格
  - 自己构造
    - 平面：Delaunay三角化（学习Triangle库）
    - 空间变形: Warping



Triangle

http://www.cs.cmu.edu/~quake/triangle.html



Triangle

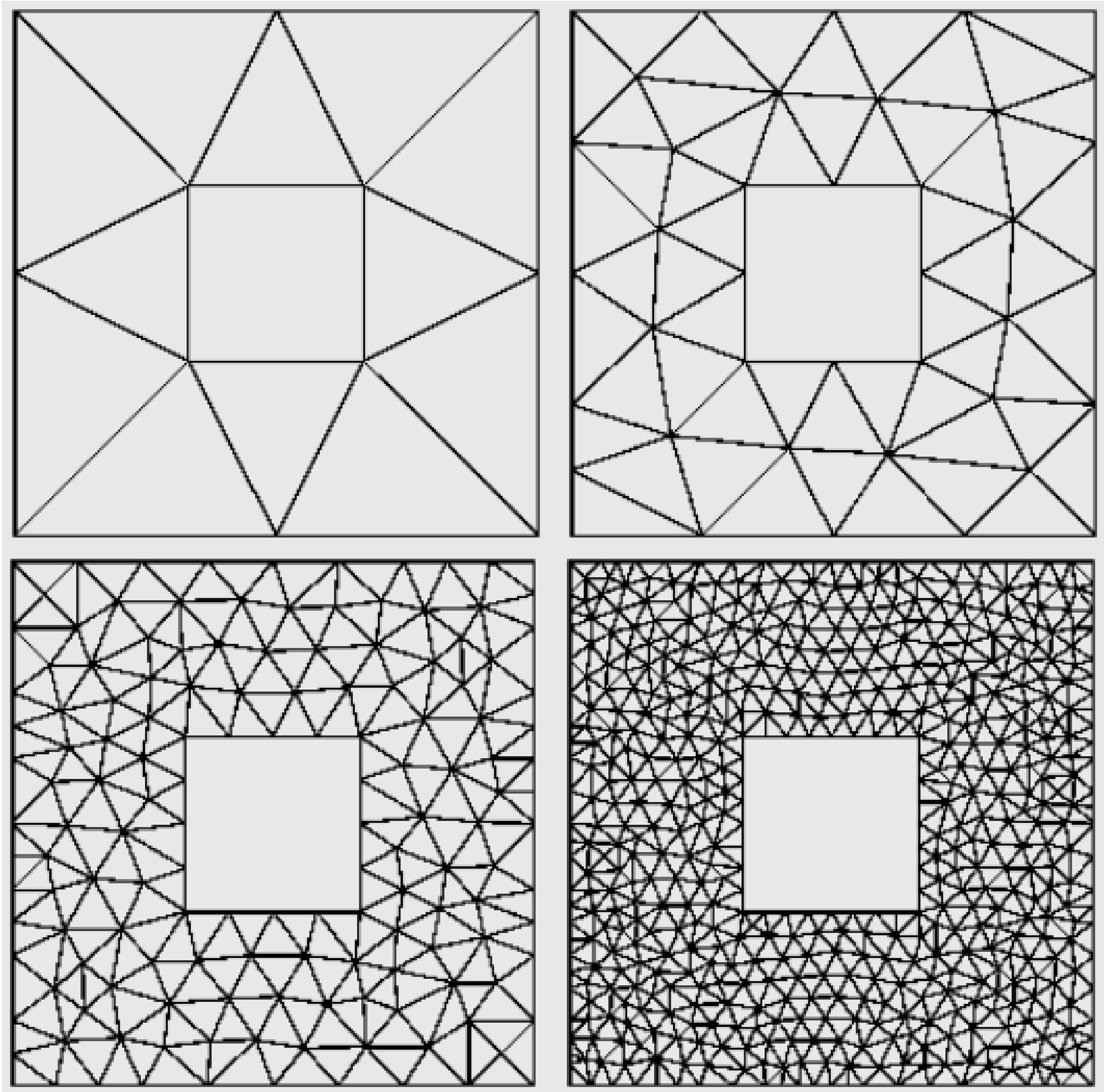
A Two-Dimensional Quality Mesh Generator and Delaunay Triangulator.

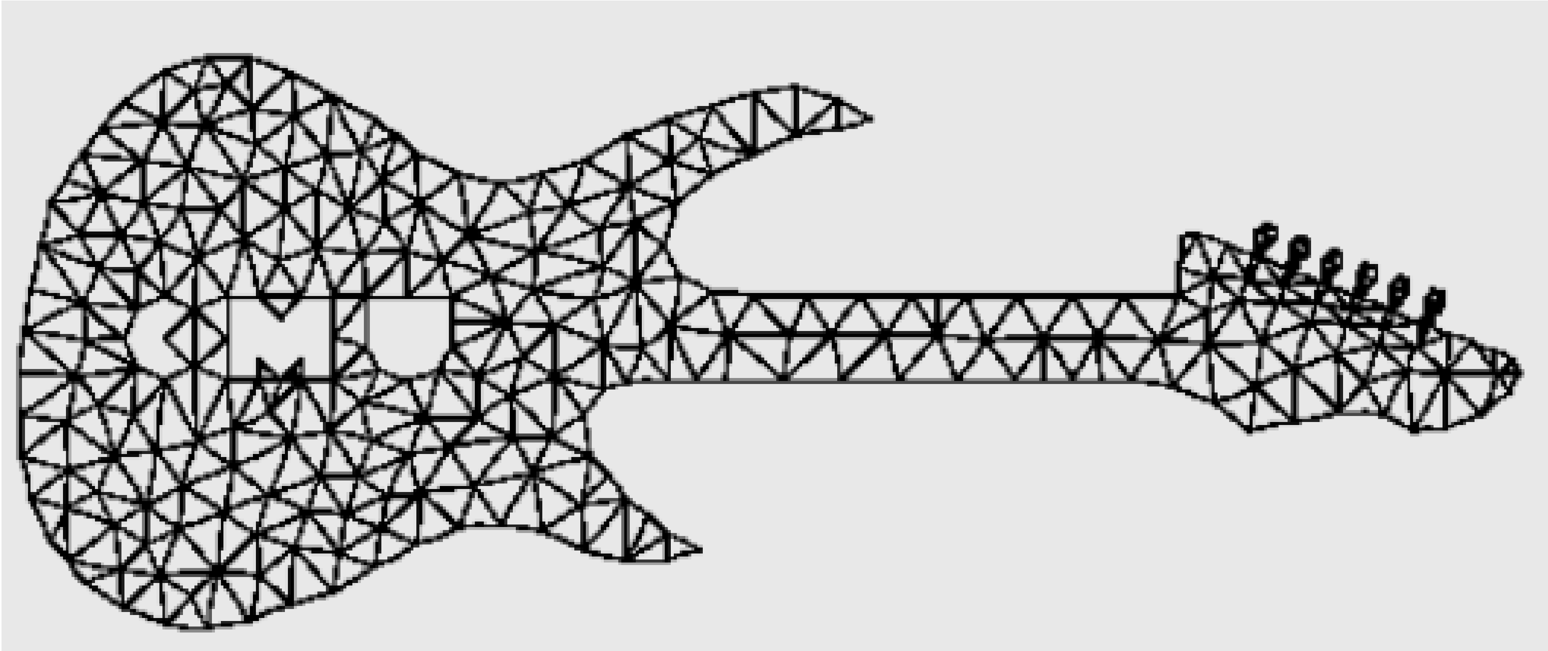
Jonathan Richard Shewchuk
Computer Science Division
University of California at Berkeley
Berkeley, California 94720-1776
jrs@cs.berkeley.edu

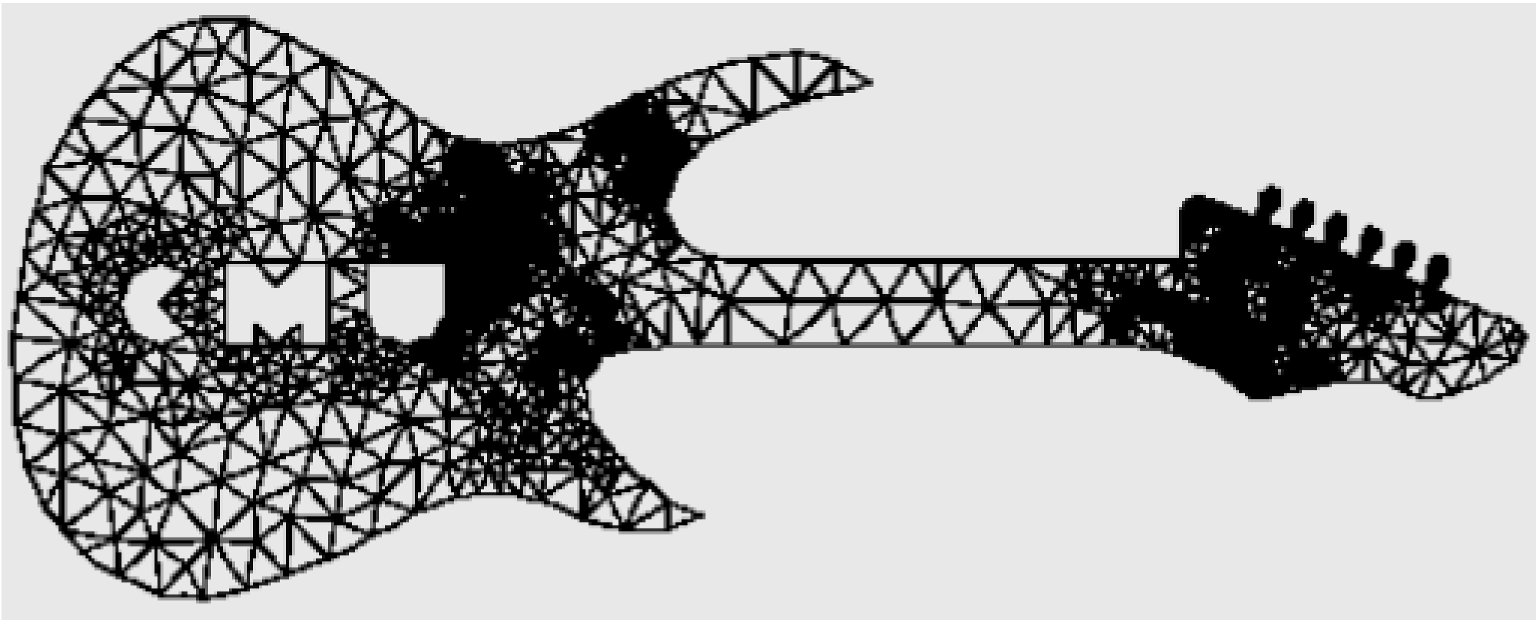
Winner of the 2003 James Hardy Wilkinson Prize in Numerical Software.
Created at Carnegie Mellon University as part of the Quake project (tools for large-scale earthquake simulation).
Supported by an NSERC 1967 Science and Engineering Scholarship and NSF Grant CMS-9318163.

Triangle generates exact Delaunay triangulations, constrained Delaunay triangulations, conforming Delaunay triangulations,
quality triangular meshes. The latter can be generated with no small or large angles, and are thus suitable for finite elem

Triangle (version 1.6, with Show Me version 1.6) is available as a .zip file (159K) or as a .shar file (829K) (extract with
voronoi directory. Please note that although Triangle is freely available, it is copyrighted by the author and may not be s
commercial products without a license.







中国科学技术大学
University of Science and Technology of China

谢　谢　！