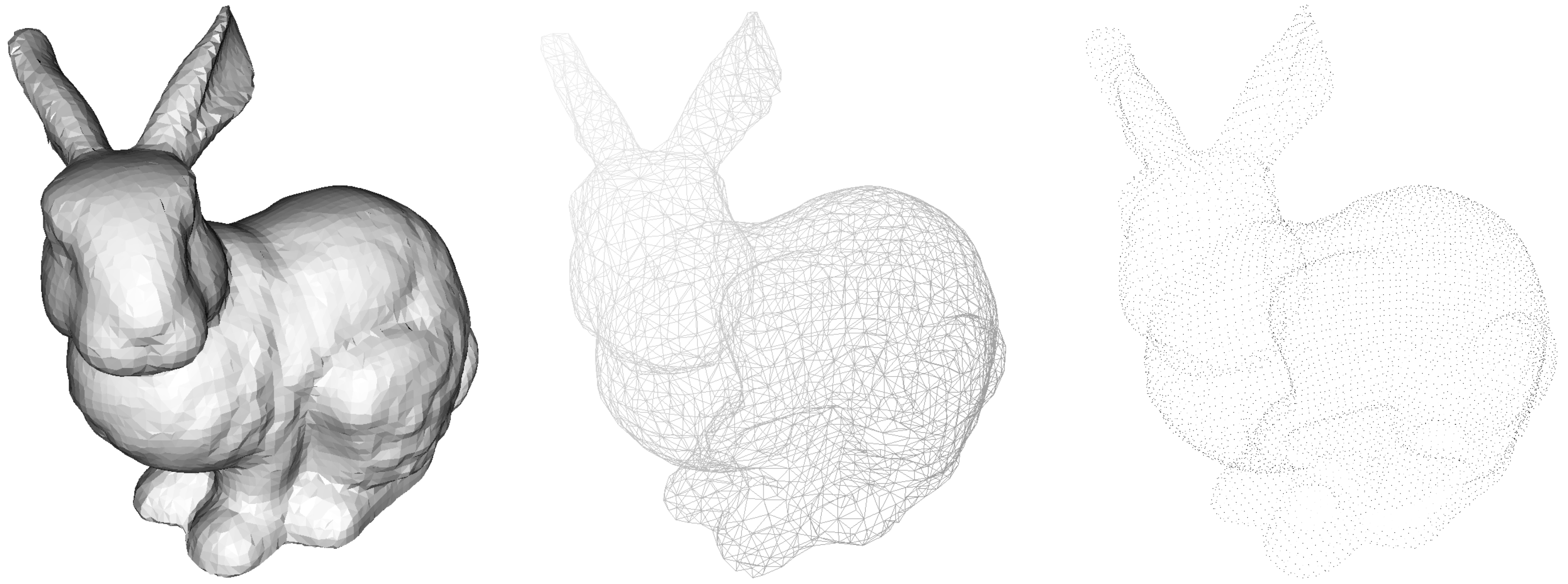
Laplacian Surface Editing

罗世通 元培学院 1600017413
2019 年 12 月

背景知识 Preliminaries



Mesh　　Graph　　Points (Nodes/Vertices)

Mesh 是什么？

- Mesh 是对三维空间中曲面的表示
- Mesh 是对完美的光滑曲面的近似：用一系列细小**三角形**的拼合近似一个曲面
- 因此，在计算机中，如果需要表示一个 Mesh，就需要一系列三维空间中的**点**（三角形的顶点），以及点之间的**关系**（哪三个点构成一个三角形）
- 从另一个角度看，Mesh 具有**图（Graph）**的结构（三角形的边视为定点之间的边）

点在空间中位置的表示

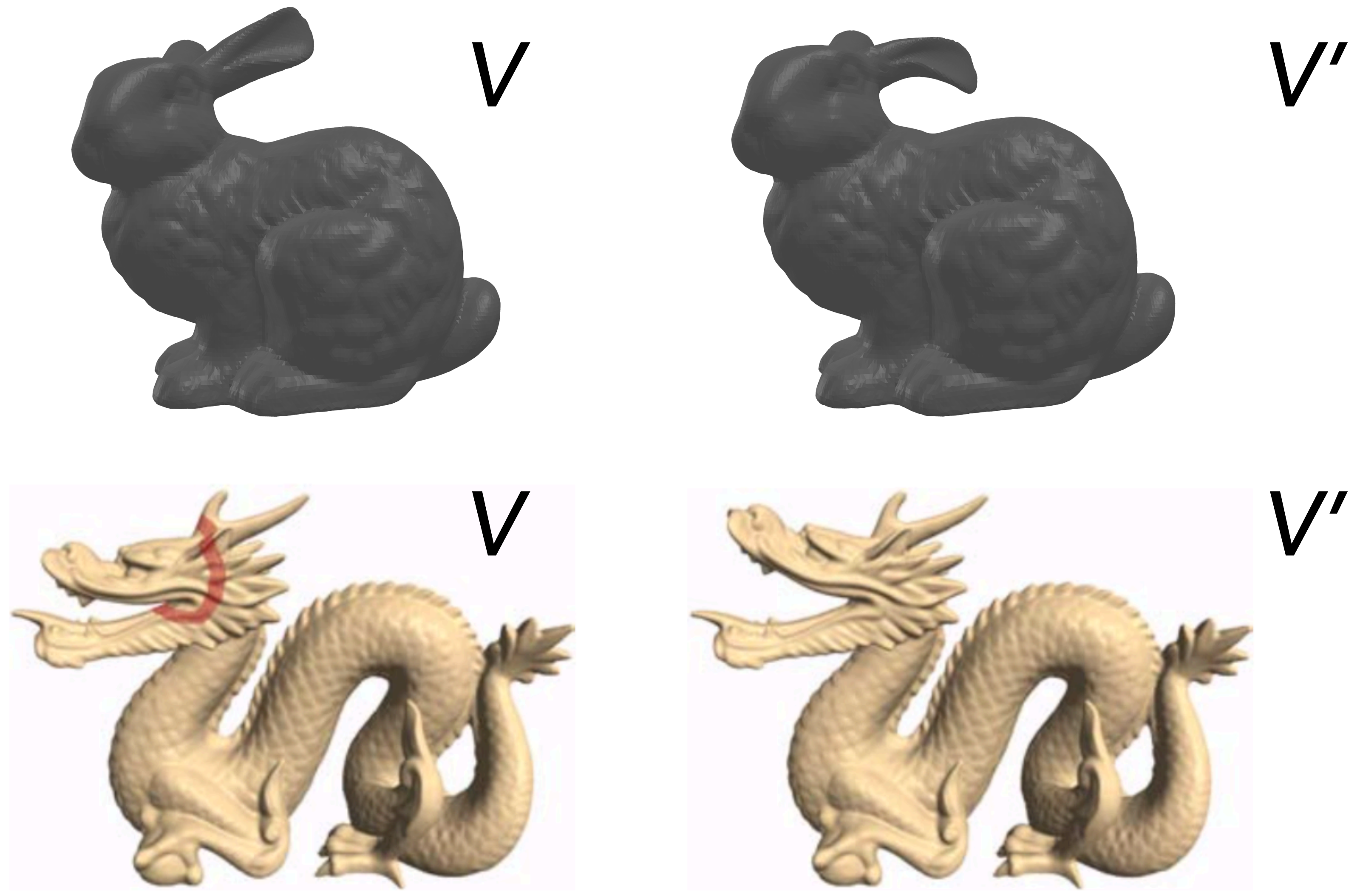
- **绝对坐标**
- **相对坐标**
  - 基于结点与其邻域结点的相对位置关系
  - 无论 Mesh 在空间中如何移动，结点的相对坐标是不变的，因此这种表示是**内在的（intrinsic）**
  - 本文使用的表示方式：首先求出邻域结点绝对坐标的平均值，然后用当前结点的绝对坐标减去这个均值，得到相对坐标

$$\delta_i = \mathbb{L}\left(\mathbf{v}_i\right) = \mathbf{v}_i - \frac{1}{d_i}\sum_{j\in\mathbb{N}_i}\mathbf{v}_j$$

$$\Delta = (I - D^{-1}A)V$$

  - 上式中的 I-D^{-1}A 称为 Laplacian 矩阵（一种变体，称为 Random Walk Laplacian）
- **两种表示之间的转换**
  - **绝对坐标 => 相对坐标**：直接计算
  - **相对坐标 => 绝对坐标**：求解线性方程组，这是一个欠定问题，需要固定一个点的绝对坐标才有解，欠定是因为相对坐标不能表示 Mesh 具体在空间中的什么位置

Problem Statement & Algorithm



要做什么？What to do?

- 对 Mesh 变形，同时保证 Mesh 整体形状发生变化的同时，细节形状保持不变；
- 形式化地说，就是给定 Mesh 所有结点的**相对坐标**以及若干**控制点**，推算出绝对坐标
  - **控制点**：一方面确定 Mesh 在空间中的位置，另一方面确定 Mesh 的形态，例如以兔子耳朵的顶端的一个结点作为控制点，移动该控制点即可导致耳朵形态的改变

怎么做？How to do?

- **给定**一个 Mesh，记其结点的绝对坐标为 ***V***，相对坐标为 ***Δ***
- **给定**若干控制点，及这些控制点的目标位置坐标
- **设**变形后的 Mesh 的结点的绝对坐标为 ***V'***
- 我们的目标是**求解 *V'***
- **[约束1]** 为了保证 Mesh 细节不被改变，要让 ***V'*** 的相对坐标与原 Mesh 的相对坐标 ***Δ*** 尽可能一致

$$\Delta - \mathbb{L}(V') = 0$$

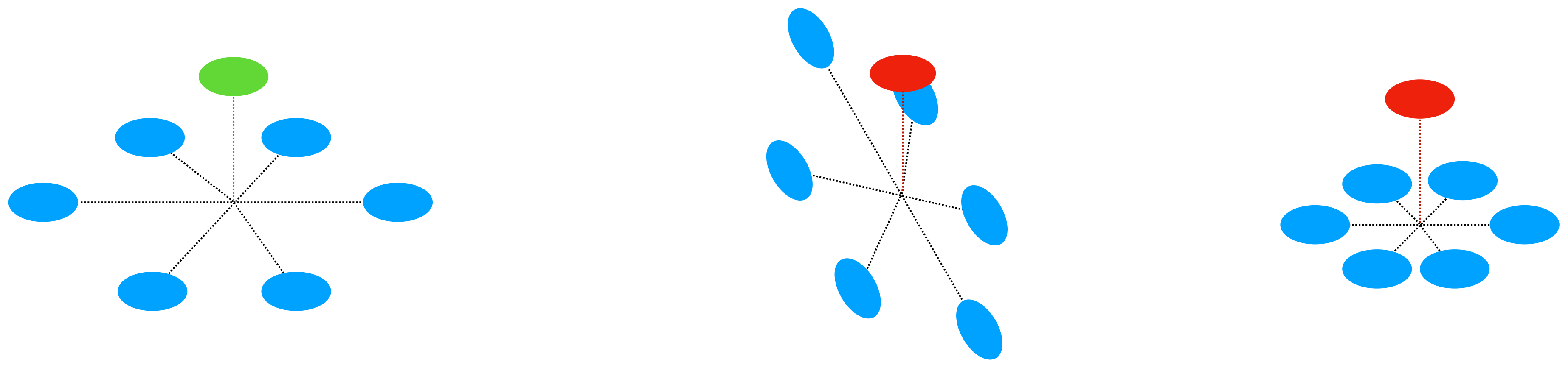
- **[约束2]** 要让控制点在变形后处于目标位置

$$\mathbf{v}_i' - \mathbf{u}_i = 0 \quad i \in P_{\text{ControlPoints}}$$

  - 其中 i 是控制点的索引，u_i 是控制点的目标位置

- 可以看到，这是一个**超定**（给定一个控制点为适定，多个控制点为超定）解线性方程组的问题，可以使用**最小二乘法**求解

Problem Statement & Algorithm (Cont'd)



旋转、缩放不变性问题

- 对 **[约束1]** 稍加修改，从 $\delta_i - \mathbb{L}(\mathbf{v}_i') = 0$ 改为

$$T_i\delta_i - \mathbb{L}(\mathbf{v}_i') = 0$$

- 其中 T_i 是**旋转缩放矩阵**（线性近似），满足以下条件：

$$V_{i\cup N(i)}T_i^{\mathsf{T}} - V'_{i\cup N(i)} = 0$$

- 用最小二乘法解出 T_i 代入 **[约束1]**，仍然是一个线性方程组，因此依然可以用最小二乘法求解

Formulation

- **给定**一个 Mesh，记其结点的绝对坐标为 ***V***，相对坐标为 ***Δ***
- **给定**若干控制点，及这些控制点的目标位置坐标 ***u***
- **设**变形后的 Mesh 的结点的绝对坐标为 ***V′***
- 我们的目标是**求解 *V′***

- **[约束1]** 为了保证 Mesh 细节不被改变，要让 ***V′*** 的相对坐标与原 Mesh 的相对坐标 ***Δ*** 尽可能一致

$$T_i\delta_i - \mathbb{L}(\mathbf{v}_i') = 0 \quad i = 1\ldots n$$

- **[约束2]** 要让控制点在变形后处于目标位置

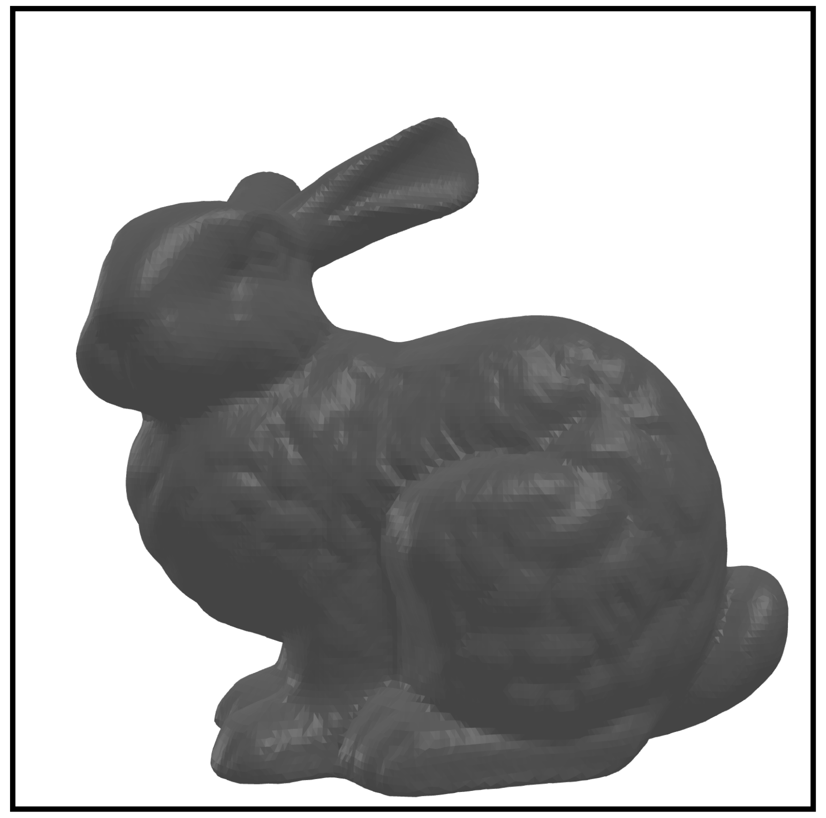
$$\mathbf{v}_i' - \mathbf{u}_i = 0 \quad i \in V_{\mathrm{ControlPoints}}$$

  - 其中 i 是控制点的索引，u_i 是控制点的目标位置

- 使用**最小二乘法**求解以上线性系统
- 由于一个结点的邻域较小，因此这个线性系统是**稀疏**的

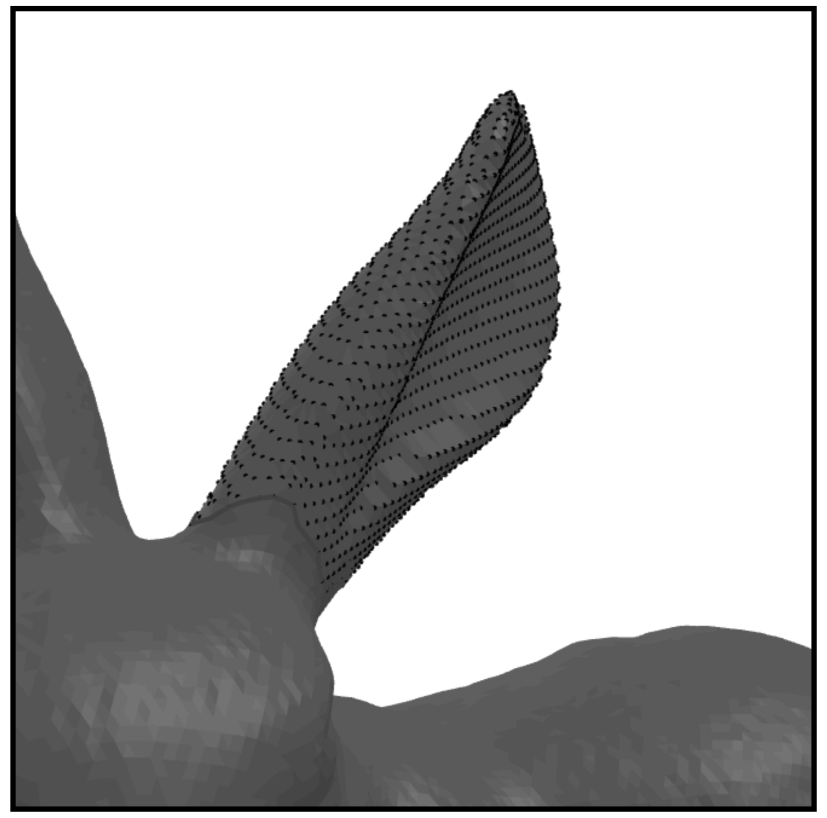
Experiments

给定一个 Mesh

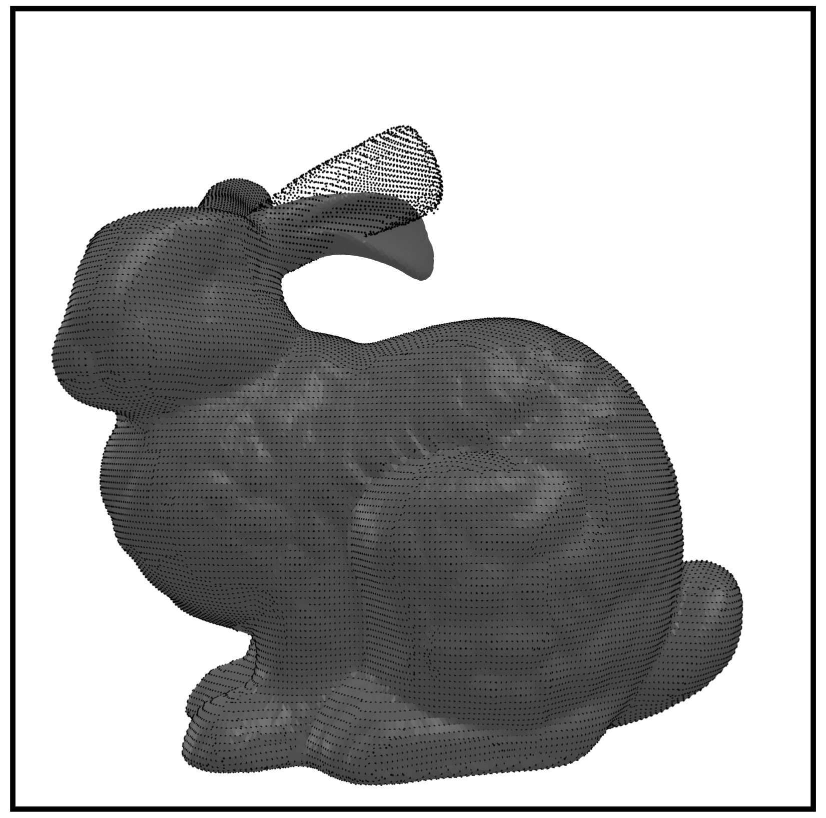


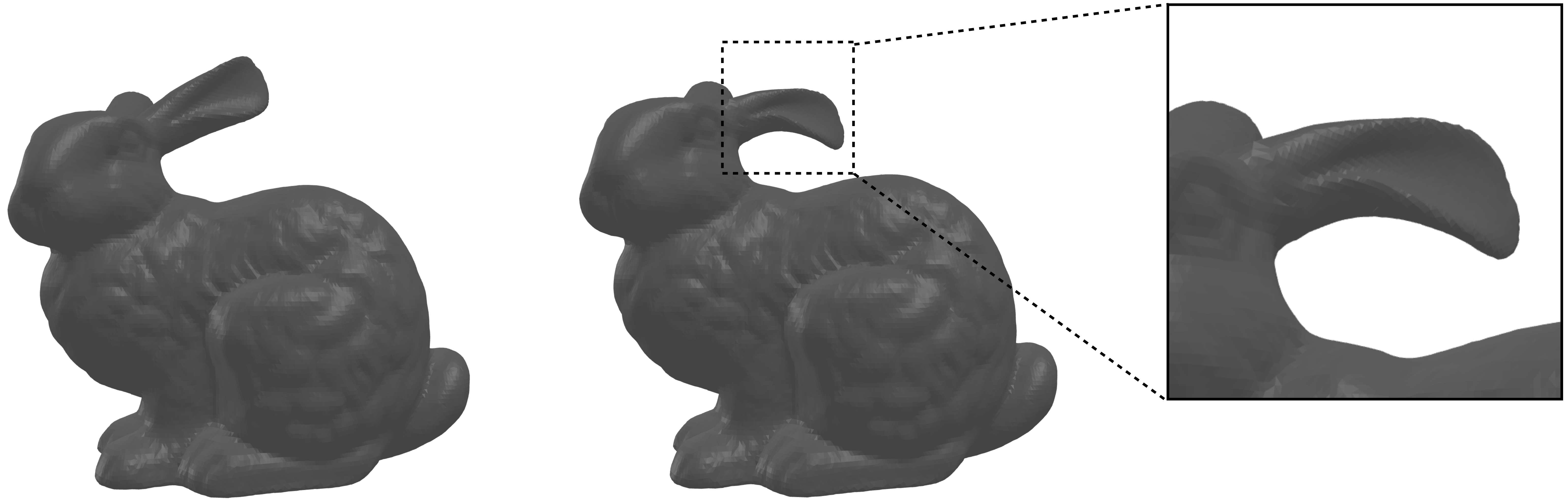
选定需要编辑的区域以及控制点

控制点包括区域的边界以及可移动的点



建立并求解线性模型





Code is available at https://github.com/luost26/laplacian-surface-editing

Conclusions & List of References

- 绝对坐标 <=> 相对坐标转换问题
- 绝对坐标：决定 Mesh 的总体形态、位置
- 相对坐标：决定 Mesh 的细节形态
- 相对坐标向绝对坐标的转换是一个稀疏线性最小二乘问题，可以较为高效地求解

[List of References]

- Sorkine, Olga, et al. "Laplacian surface editing." *Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing*. ACM, 2004.
- The Stanford 3D Scanning Repository. http://graphics.stanford.edu/data/3Dscanrep/

Code is available at *https://github.com/luost26/laplacian-surface-editing*

谢谢！