

基于GPU的点在多面体内的快速鲁棒判定方法

李静*, 王文成. Fast and Robust GPU-based Point-in-polyhedron Determination.

Computer-Aided Design, Vol.87, p.20-28, 2017.

*e-mail: superredlark@163.com, *tel: 13693634172

概述

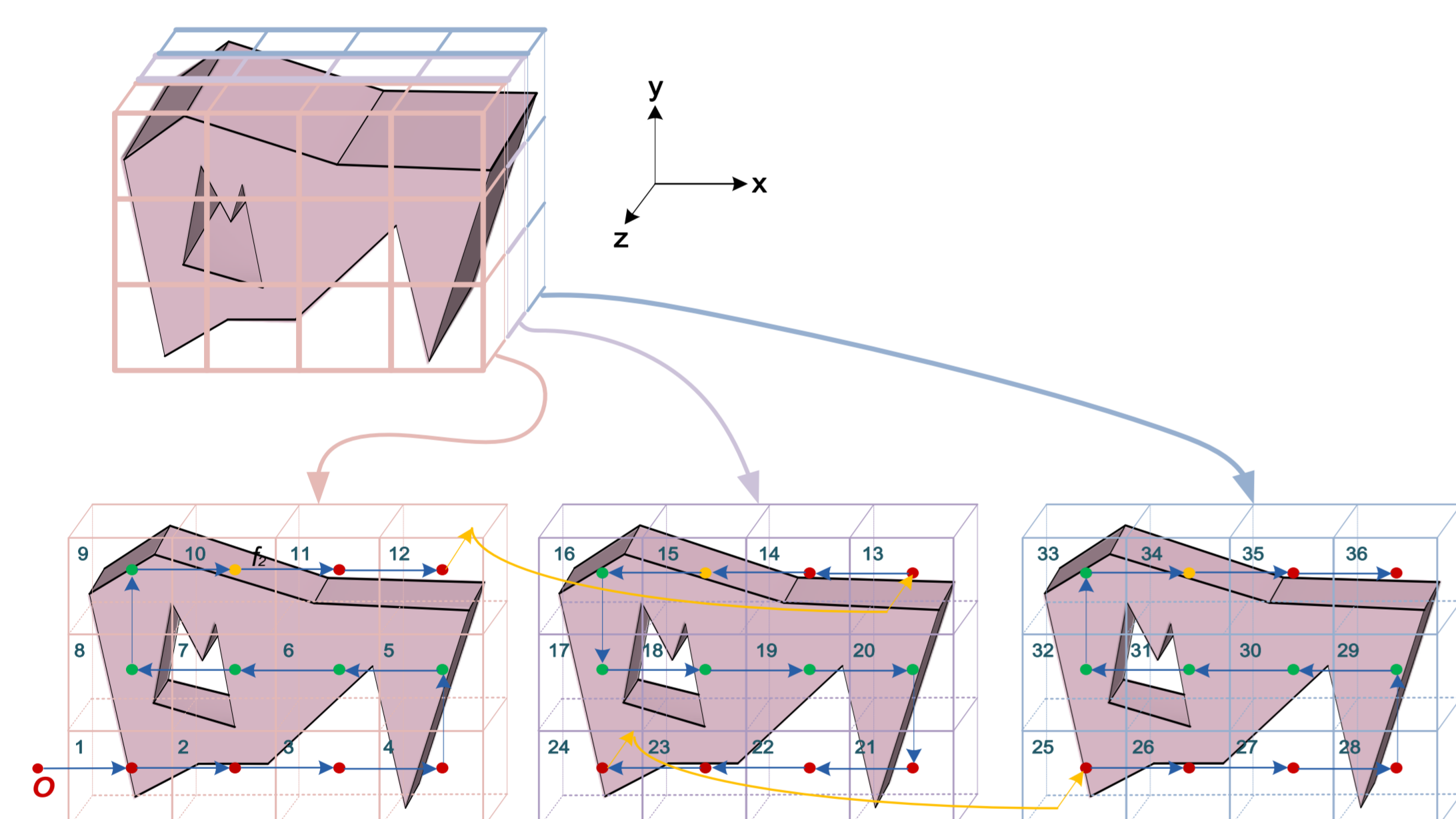
- 判定三维空间中的一个给定点是否位于给定多面体内是计算几何中的一个基本问题。
- 多面体点检测的方法主要有两类：全局化方法[1]和局部化方法[2,3]。
- 全局化方法，包括ray-crossing和基于四面体的方法[1]，均需要基于多面体所有的面计算某种度量特征来确定点的属性，复杂度为 $O(N)$ （ N 为多面体的面数），处理速度慢。
- 局部化方法[2,3]，通过建立某种结构对多面体的面进行有序组织，由此可进行一些局部化的度量检测获得判定结果，速度快。但组织结构通常以树形结构为主，不利于使用GPU的并行结构进行加速。
- GPU已被广泛应用于很多基础几何算法的加速计算中，但仍未被有效用于点在多面体内的检测问题。

目标

- 提高大量点是否位于多面体内的判定速度

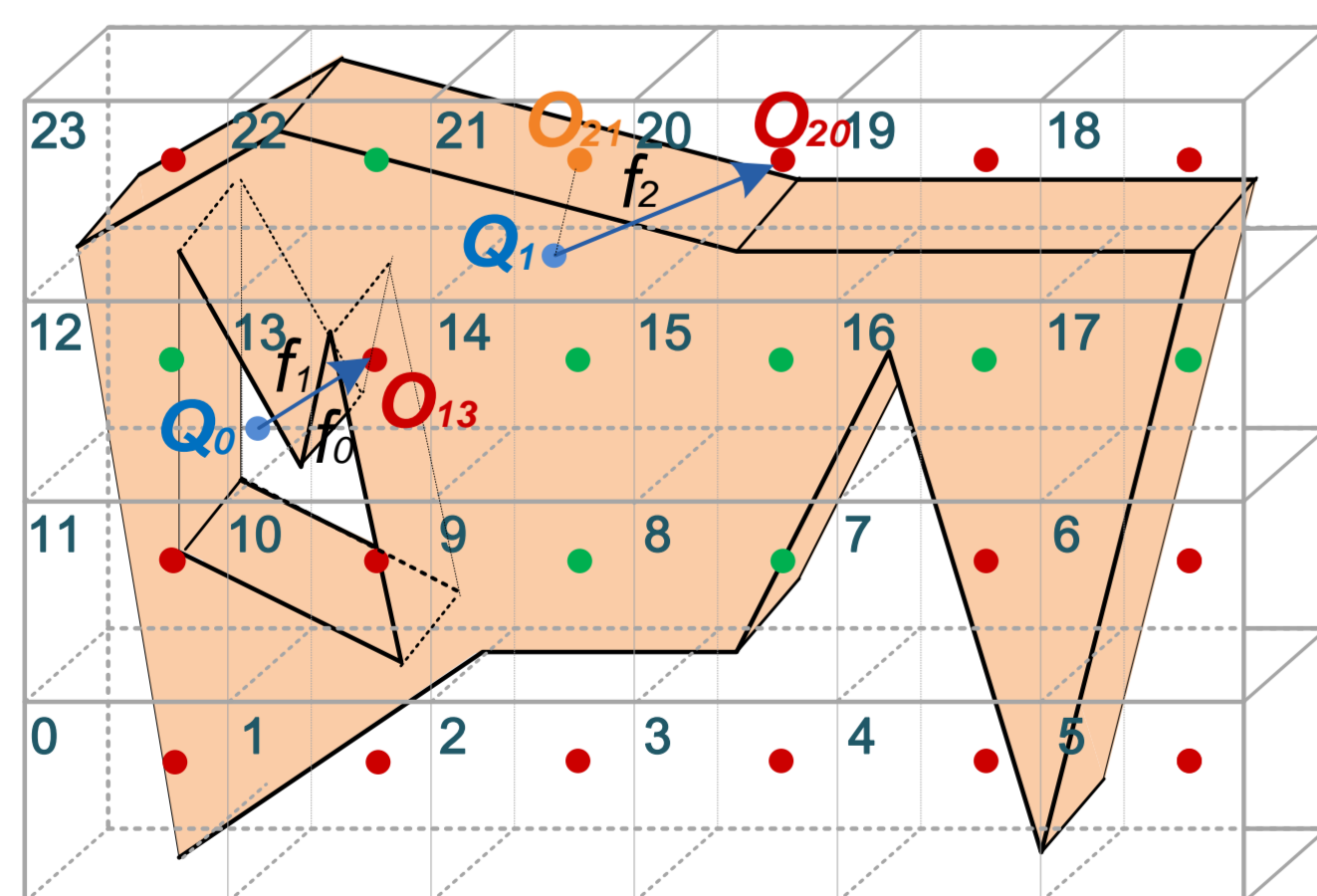
思路

- 建立均匀网格，预计算网格单元中心点的属性（多面体内/多面体外）。



Cell# 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36
S 0 0 0 0 -1 -1 -1 -1 -1 -1 0 0 0 0 -1 -1 -1 -1 0 0 -1 -1 -1 -1 0 0 -1 -1 -1 -1 -1 0 0
Property out

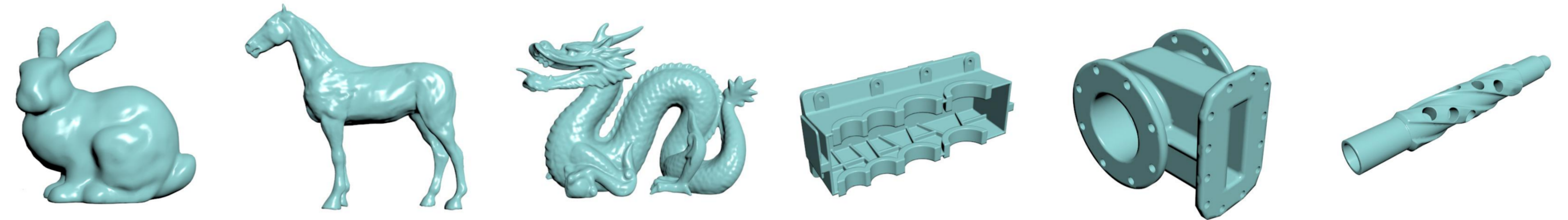
- 当给定一个被测点时，仅需要处理与被测点位于同一个网格单元内的面片，统计被测点与该单元中心点的连接线段与这些面片的交点数量（这可以局部化地运用ray-crossing方法获得）。



- 基于均匀网格中心点的局部化操作避免了大量求交计算，降低了计算复杂度，也便于使用GPU的并行构架进行加速计算。
- 给出了歧义情况的鲁棒性处理方法，确保正确的判定。
- 对于应用中的常见模型，本文方法的判定时间复杂度为 $O(1)$ ，能够以极快的速度完成大量点的判定任务。

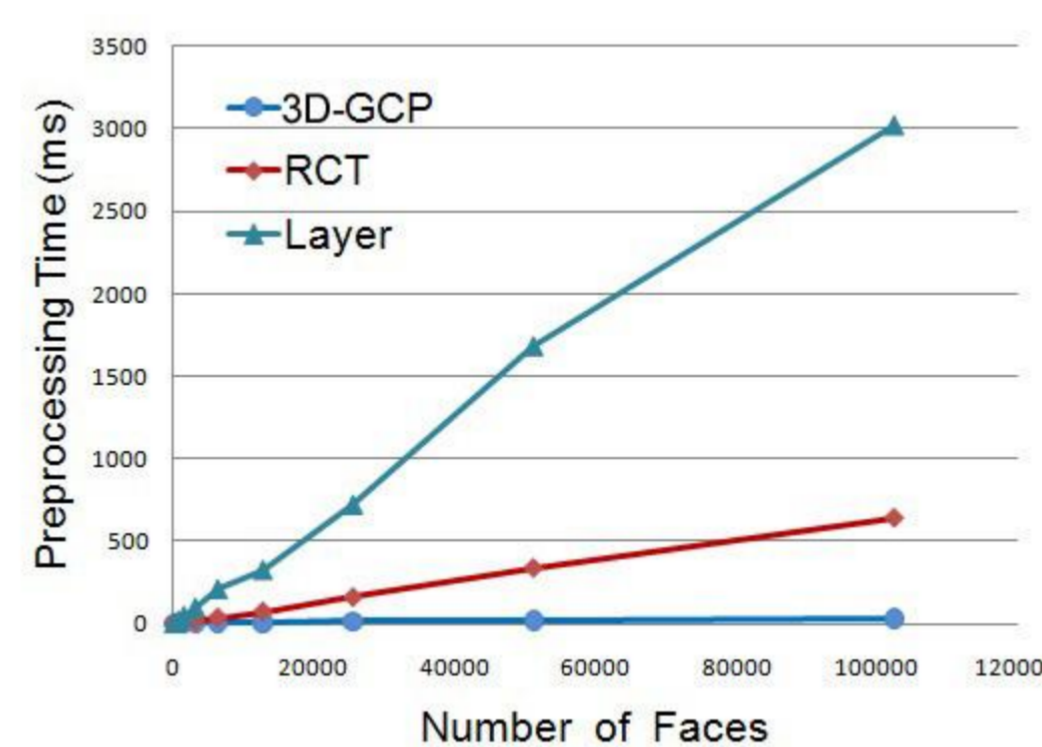
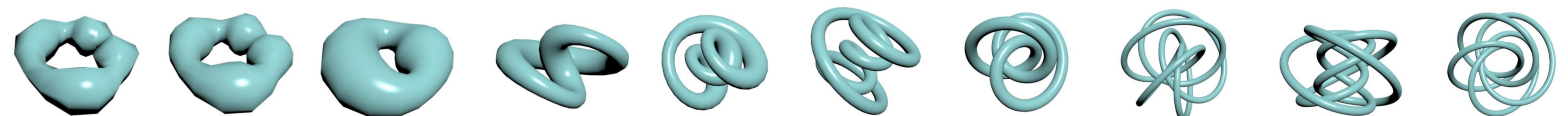
对比实验结果

- 1,000,000个点的判定时间对比（单位：毫秒）

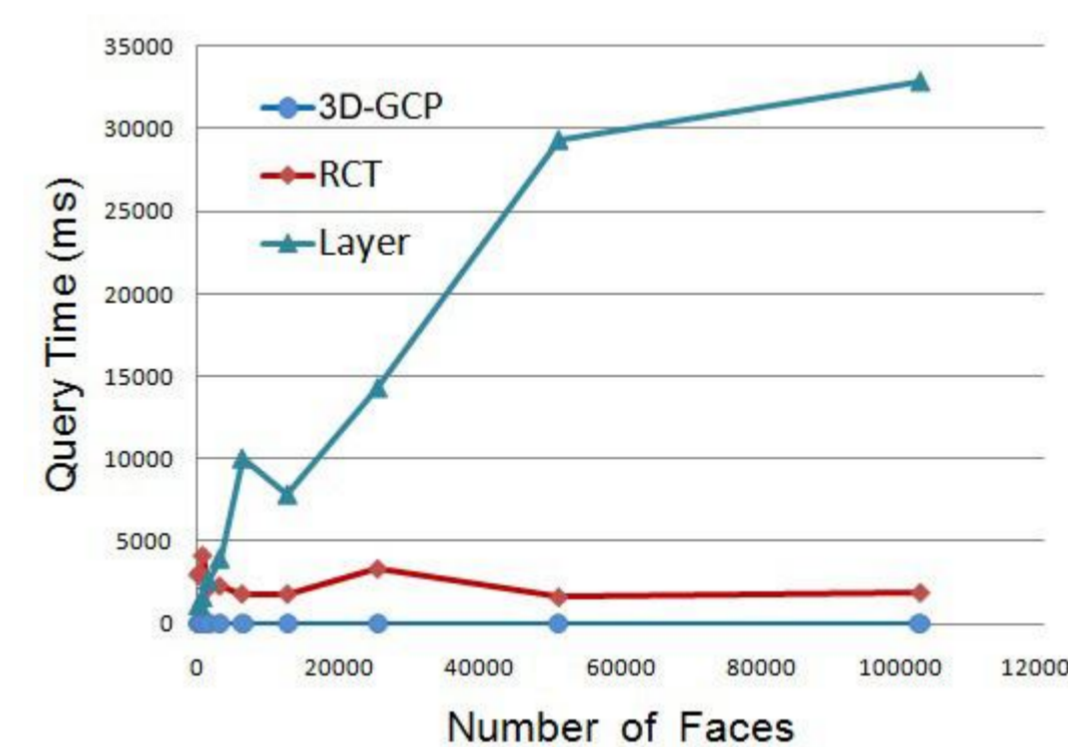


Method	Rabbit	Horse	Dragon	Mechanical Part	Pipe	Long Object
ray-crossing(并行)	9800	52645	245360	100371	35873	49642
Tetra[1](并行)	18921	94590	430512	260156	83908	101349
Layer[2](串行)	12471	26986	85013	49686	28500	33938
RCT[3](串行)	1625	1349	3119	1568	1930	3259
3D-GCP(本文方法, 并行)	7	8	8	8	8	9

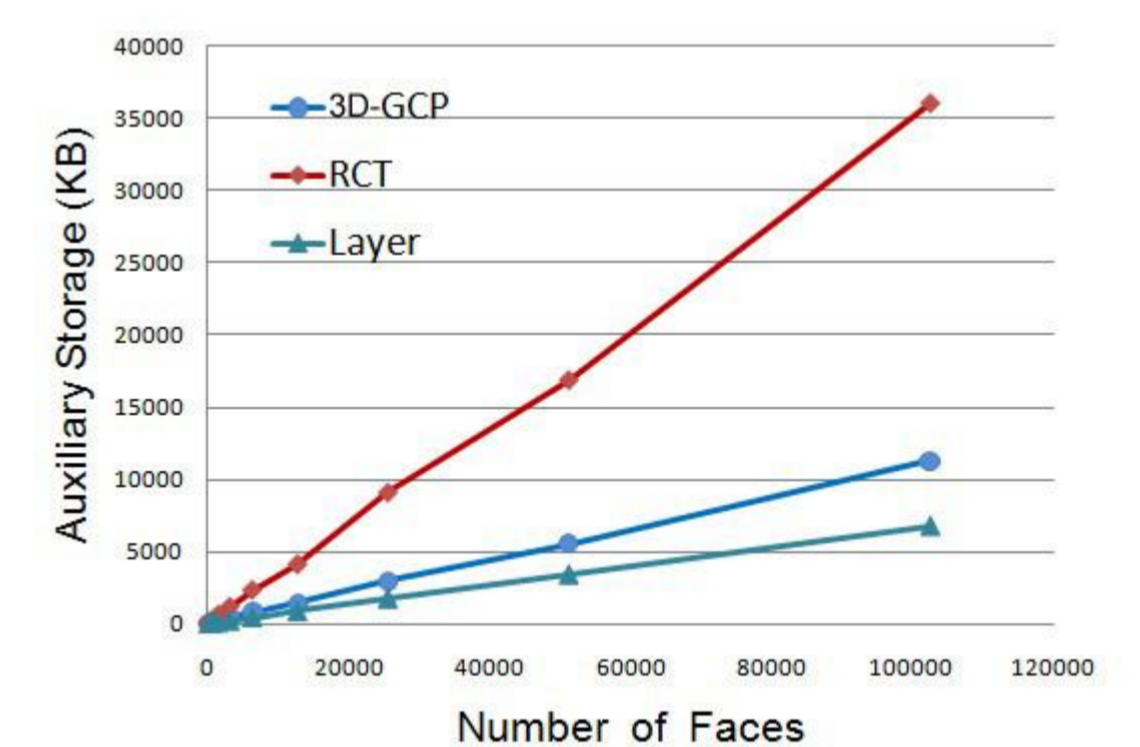
- 性能随多面体规模的变化曲线



(a) 预处理时间

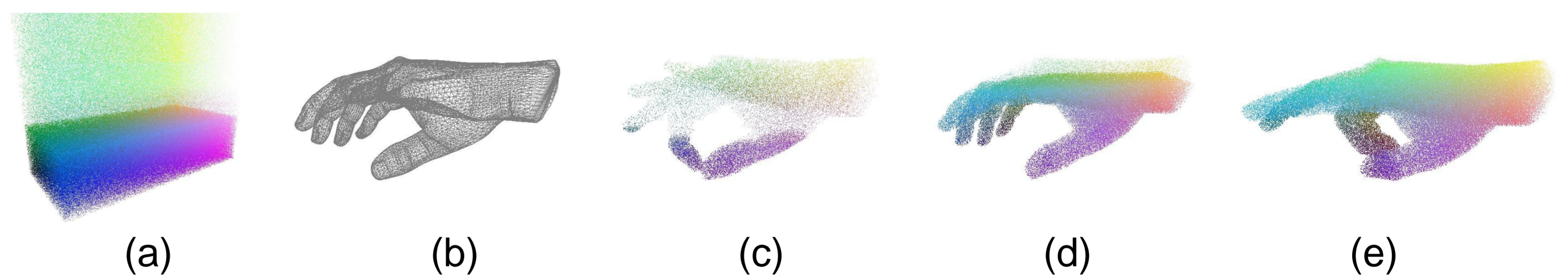


(b) 1M个点的总判定时间



(c) 空间

- 动画中的应用。1百万个点(a)自上而下从包含15K面片的动态改变的手部模型(b)中穿过，并自模型底部逐步累积。实时判定点是否位于动态变化的模型内部，且仅显示位于模型内部的点(c)-(e)。在配置有NVIDIA Quadro 6000 GPU的台式机上，屏幕分辨率为512*512时，平均帧速率为40fps。



结论

- 显著提高点在多面体内的检测效率，便于海量的点的判断、以及模型动态改变的应用。

参考文献

- [1] Rueda AJ, Ortega L, Geometric algorithms on CUDA. In: Proceedings of GRAPP 2008. Paper No. 59.
- [2] Wang W, Li J, Sun H, Wu E. Layer-based representation of polyhedrons for point containment tests. *IEEE Trans Vis Comput Graphics* 2008;14(1):73–83.
- [3] Liu JF, Chen YQ, Maisog JM, Luta G. A new point containment test algorithm based on preprocessing and determining triangles. *Comput-Aided Design* 2010; 42(12): 1143 – 50.