# Parallel construction for model-adaptive convex bounding polyhedron

TANG Lei, SHI Kanle, YONG Junhai. etc

CG&CAD, School of Software, Tsinghua University

2014-01-22

# 目录

# 引言

包围体在计算机图形学和计算几何领域中应用广泛,
常用于加速几何求交、光线跟踪和碰撞检测等多种算法.

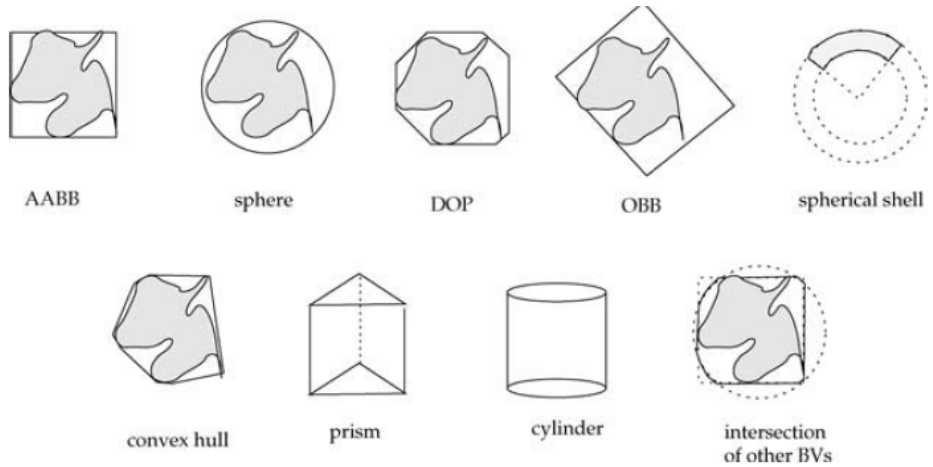



Figure 1: 各种各样的包围体[1]

# 引言

综合来看：

k-DOP[2] 方向固定且有限，不同模型其截面方向一致，不够紧致.

凸包 很(最)紧致，但面片数量太多，复杂度O(n log n).

本文目标：

紧致 能够自适应模型

快速 利用GPU加速

灵活 通过参数 k 调节简单性和紧致性

# 目录

# 问题的定义

由 k 个截面构成的凸包围多面体称为凸包围 k 面体(k-Convex Bounding Polytope, 简称 k-CBP), 可通过 k 个半空间定义:

$$\begin{cases} k\text{-}CBP = \bigcap_{i=1}^{k} H_i \\ H_i = \left\{ p \in \mathbb{R}^3 \middle| n_i \cdot p \leq w_i, w_i \in \mathbb{R} \right\}, \end{cases} \tag{1}$$

其中, $n_i$ 是半空间 $H_i$ 的法向, 方向指向包围体外部, $w_i$ 是输入点集中沿 $n_i$ 方向投影的最大值.

# 算法流程

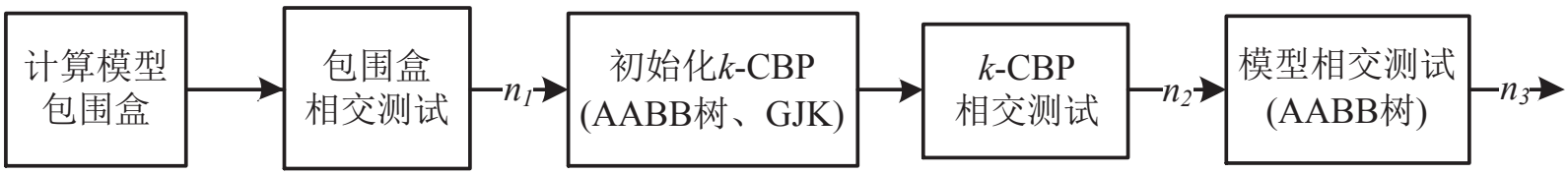



Figure 2: 算法流程图

法向 结合近似内凸包和 k-means

截面 GPU 中沿各法向搜索切点构造截面

交点 截面对偶映射求得交点

# 目录

# 近似凸包的构造

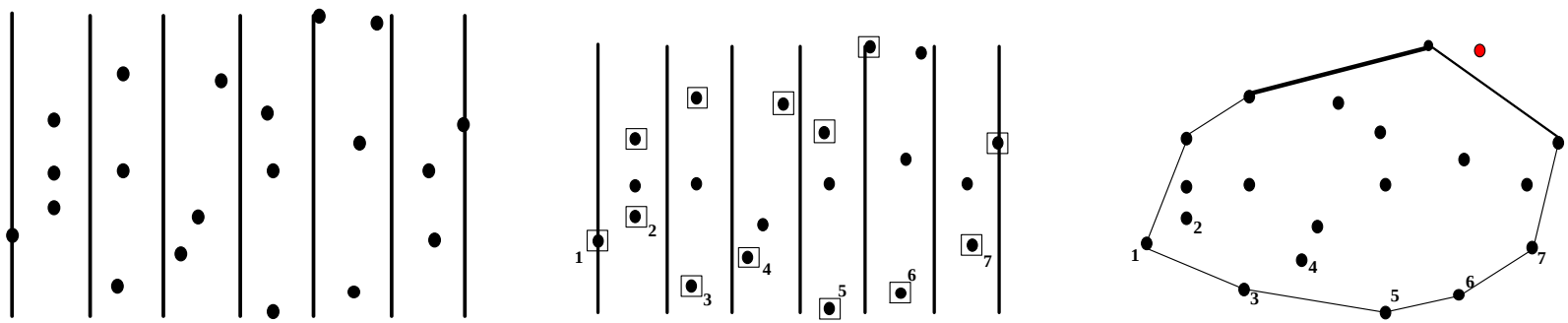
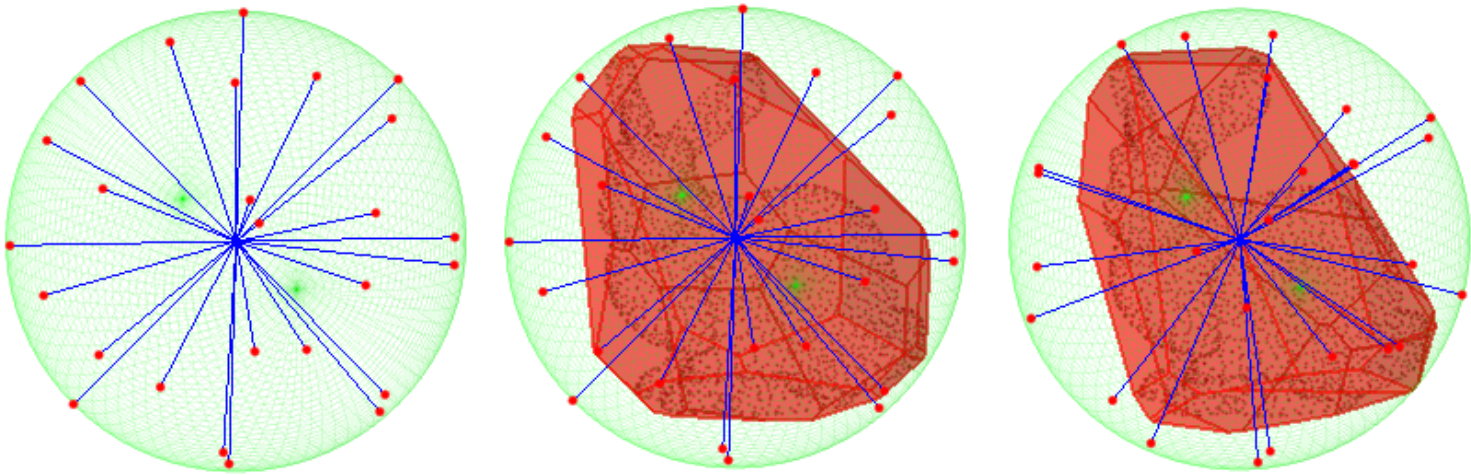



Figure 3: 二维近似内凸包的构造

构造近似内凸包[3]，算法复杂度为 $0(n+\xi)$,
扩展到三维为 $0(n+\xi^2 \log \xi)$，然后利用 k-means 聚类.

# k-means 聚类



(a) (b) (c)

Figure 4: 通过聚类确定法向

聚类初始方向(均匀分布)． 距离度量(余弦)，
聚类更新中心点时将面片的面积作为权重即$c_i = \frac{\sum_{i=1}^{i=n} \omega_i \cdot n_i}{\sum_{i=1}^{i=n} \omega_i}$， 其中$c_i$ 为第 i 类的中心点，$\omega_i$ 为法向 $n_i$ 所在面片对应的面积.

# 目录

# 搜索截面

等效于寻找最大投影值，即对每个法向 $n_i$，
从输入模型的所有点中寻找最大投影值的点作为切点进而确定 $n_i$ 对应的截面.
时间复杂度为 $O(k \cdot n)$，其中 k 为法向数量，n 为模型所含点数.
各法向的计算相互独立，借助 GPU 并行加速.

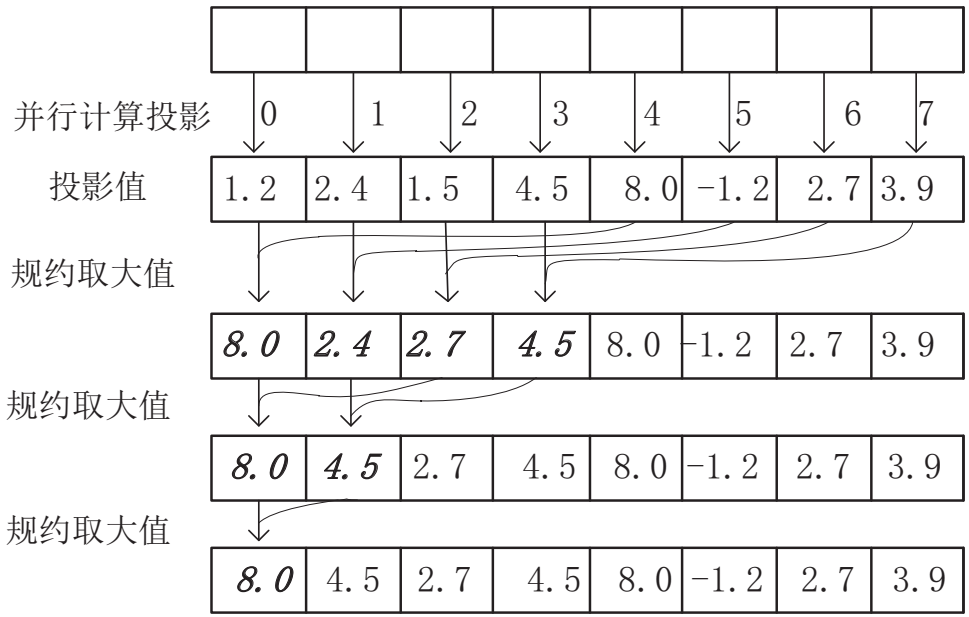



Figure 5: 并行规约求最大投影值

# 求交算法

法向 $n(a,b,c)$ 及平面上一点 $p(x_0,y_0,z_0)$ 确定，
转化为平面方程 $ax+by+cz=ax_0+by_0+cz_0=d,\ d\neq 0.$
对偶映射后的点为 $p'(a/d,b/d,c/d)$，对这 $k$ 个映射点求凸包，
凸包平面映射回原来的交点，时间复杂度为$O(k\log k)$.
亦可直接通过枚举所有每 3 个平面交于 1 点的情况，
然后排除在平面外部的交点，剩下的构成 k-CBP 的顶点，
时间复杂度为$O(k^3)$[4].

# 目录

## 凸包围多面体的生成速度

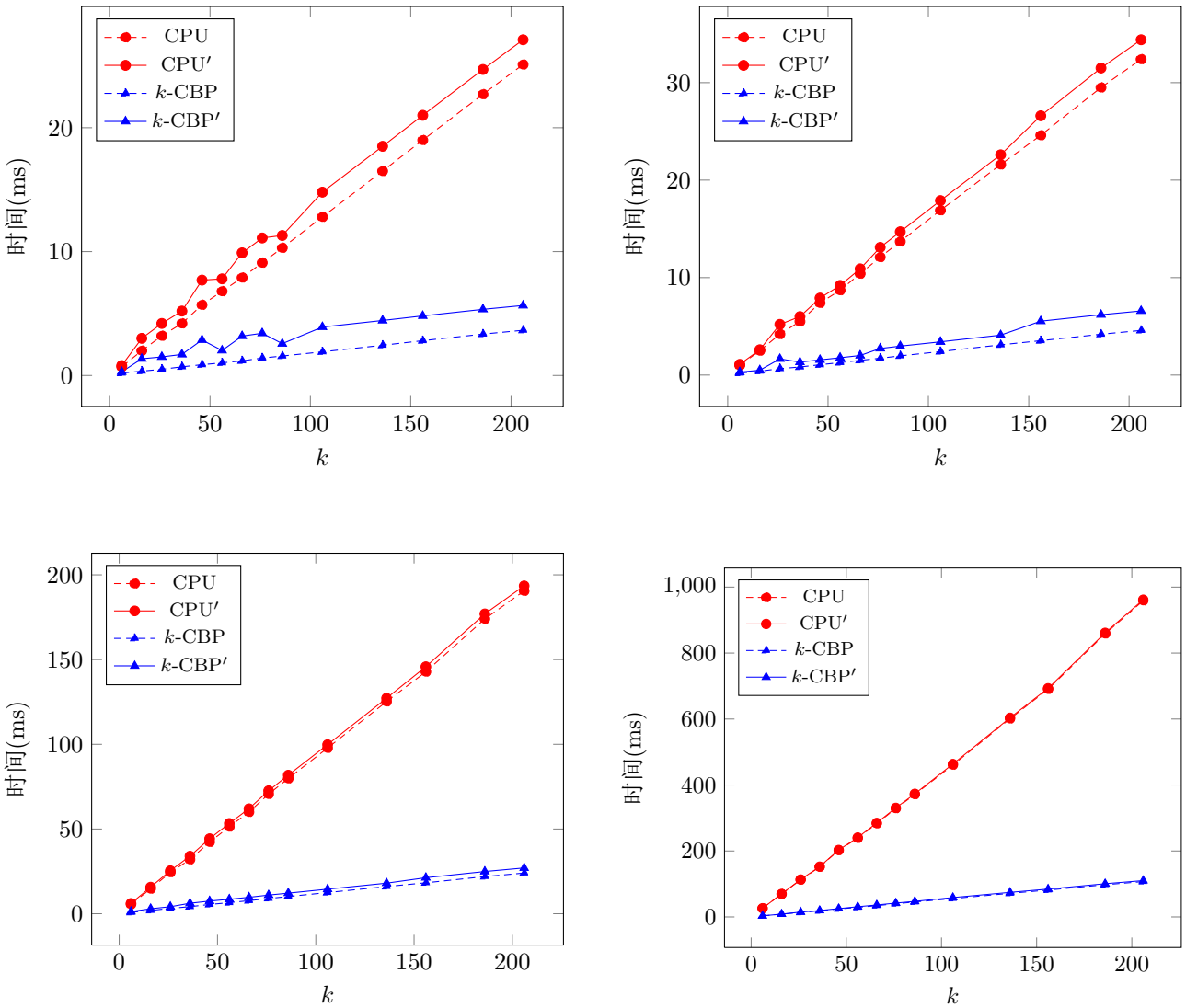



Figure 6: 本文算法与 CPU 算法对比Budda(31k),Dinosaur(40k),Alice(224k), Bugatti(1011k)

图 6 中横纵坐标分别代表多面体面数和运行时间,
其中虚线代表搜索截面的过程,
实线为构造凸包围多面体总体耗时. 当模型点数量较大时,
搜索截面的过程占据了算法绝大多数时间,
且随着凸包围多面体的面数 k 值增加而线性增长,
这与搜索截面时间复杂度($O(k \cdot n)$)一致,
截面求交过程的时间复杂度为$O(k \log k)$, 当点数量极大时,
实线虚线几乎重合即求交等步骤耗时相比整体算法而言几乎可忽略.

Table 1: 本文算法与文献[5]算法对比

| k | Apple(8118 points) | | | Bugatti(1010815 points) | | |
|---|---|---|---|---|---|---|
| | SSE[5](ms) | k-CBP(ms) | Speedup | SSE(ms) | k-CBP(ms) | Speedup |
| 6 | 0.4 | 0.12 | 3.20 | 24.2 | 3.20 | 7.56 |
| 16 | 0.9 | 0.26 | 3.43 | 44.5 | 8.44 | 5.27 |
| 26 | 1.4 | 0.41 | 3.38 | 66.5 | 13.65 | 4.87 |
| 36 | 1.9 | 0.52 | 3.65 | 91.1 | 18.34 | 4.97 |
| 46 | 2.5 | 0.67 | 3.74 | 119.5 | 24.13 | 4.95 |
| 56 | 2.9 | 0.79 | 3.66 | 138.4 | 28.86 | 4.80 |
| 66 | 3.5 | 0.95 | 3.69 | 170.6 | 34.10 | 5.00 |
| 76 | 4.0 | 1.08 | 3.70 | 197.1 | 39.85 | 4.95 |
| 86 | 4.5 | 1.22 | 3.69 | 219.8 | 45.08 | 4.88 |
| 106 | 5.4 | 1.49 | 3.62 | 267.8 | 55.52 | 4.82 |
| 136 | 6.8 | 1.92 | 3.54 | 342.9 | 71.24 | 4.81 |
| 156 | 7.7 | 2.17 | 3.55 | 411.3 | 81.18 | 5.07 |
| 186 | 9.3 | 2.60 | 3.58 | 479.4 | 97.39 | 4.92 |
| 206 | 10.5 | 2.85 | 3.68 | 523.0 | 106.87 | 4.89 |

点数量较小时, 能够提高 3-4 倍速度,模型变大, 加速比更大, Bugatti 模型的提速达到 4-8 倍.

# 目录

凸包围多面体的紧致程度

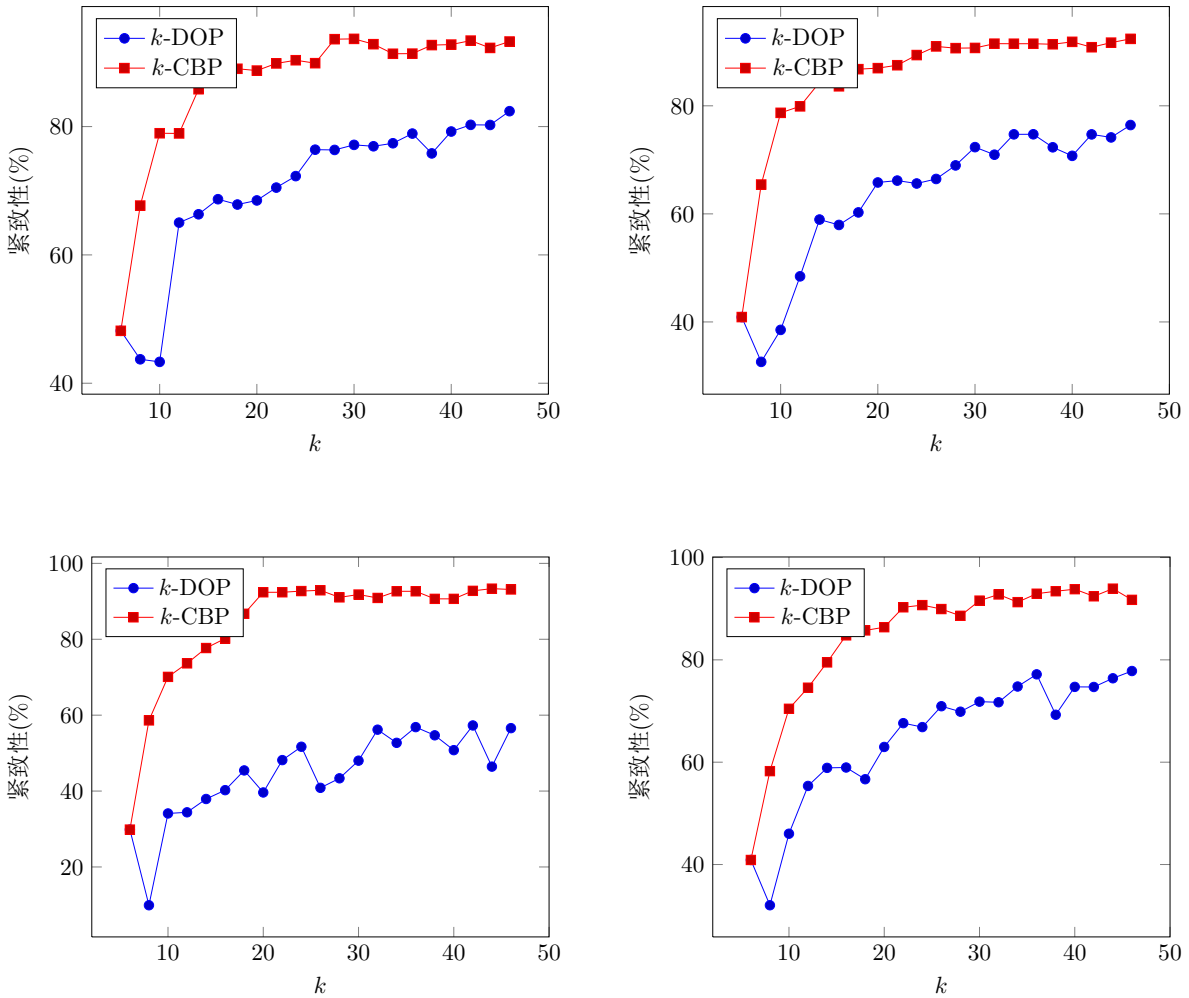



Figure 7: 紧致程度对比: k-DOP v.s k-CBP Apple(8k), Budda(31k), Dinosaur(40k), Alice(224k)

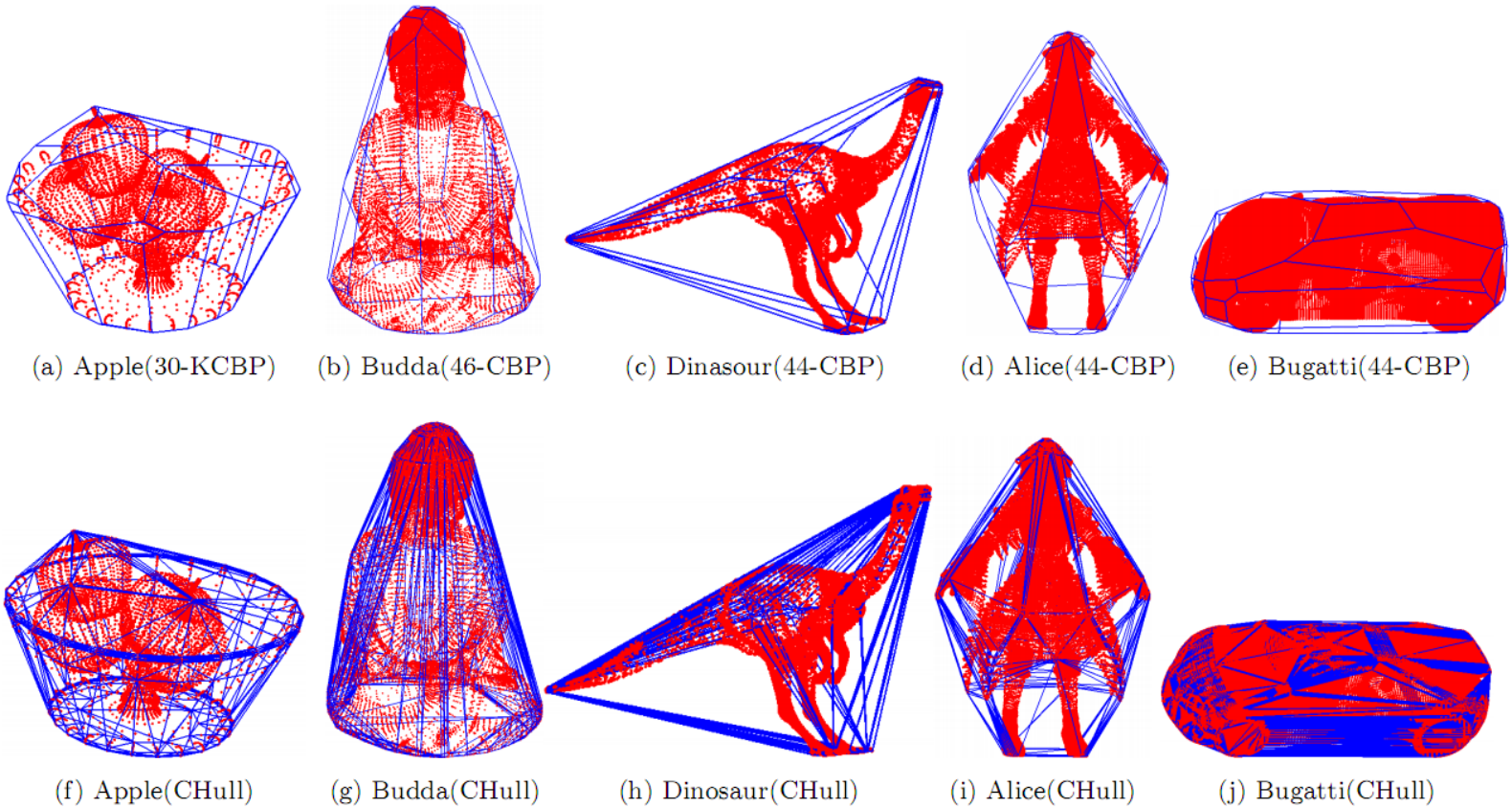
紧致程度用凸包与凸包围多面体的体积之比来量化.

Table 2: k-CBP 与 QuickHull 凸包算法比较

| Model | f(CHull) | f(k-CBP) | $\tau$ (k-CBP) | t(CHull(ms)) | t(k-CBP(ms)) |
|---|---|---|---|---|---|
| Apple | 499 | 30 | 93.67% | 5.5 | 1.30 |
| Budda | 1608 | 46 | 92.39% | 21.3 | 2.86 |
| Dinosaur | 1240 | 44 | 93.34% | 22.6 | 1.99 |
| Alice | 1332 | 44 | 93.92% | 85.8 | 8.47 |
| Bugatti | 24654 | 44 | 95.06% | 688.7 | 25.41 |

$\tau$(k-CBP) 为凸包围多面体的紧致程度. 与凸包相比,
本文算法在大大简化包围体平面数量的同时能保持较好的紧致程度,
下图为可视化结果.

凸包围多面体的紧致程度



(a) Apple(30-KCBP) (b) Budda(46-CBP) (c) Dinasour(44-CBP) (d) Alice(44-CBP) (e) Bugatti(44-CBP)

(f) Apple(CHull) (g) Budda(CHull) (h) Dinosaur(CHull) (i) Alice(CHull) (j) Bugatti(CHull)

Figure 8: k-CBP 与凸包对比

# 目录

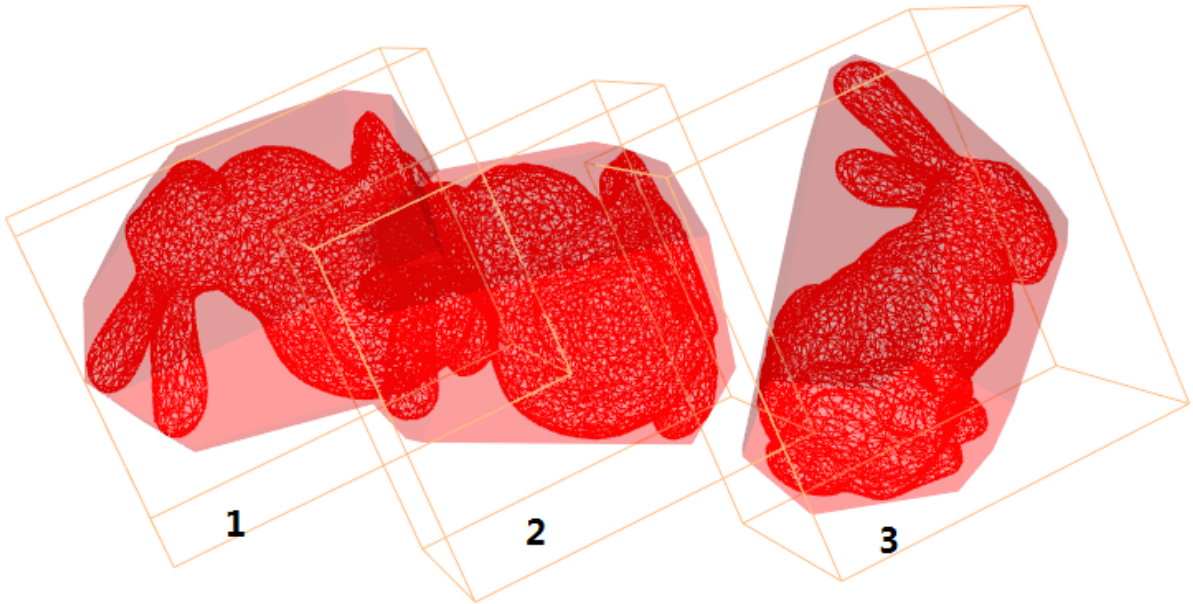



Figure 9: k—CBP 应用于碰撞检测示例

图中模型 1 与 2、2 与 3 的包围盒分别相交,
而其 16-CBP 仅 1 与 2 相交, 实际模型仅 1 与 2 相交.
不同数量的模型(模型位置和旋转角度随机生成)测试结果如下表所示.

Table 3: k-CBP 和包围盒应用于碰撞检测结果对比

| n | c(Box) | c(16-CBP) | t(Box) | t(16-CBP) | r(Box) | r(k-CBP) | n(Model) |
|---|---|---|---|---|---|---|---|
| 10 | 0.1 | 1.8 | 26.0 | 0.1 | 0.00 % | 100.00% | 0 |
| 30 | 0.2 | 2.9 | 134.0 | 70.0 | 45.45% | 83.33% | 5 |
| 50 | 0.5 | 4.8 | 506.0 | 255.2 | 46.34% | 86.36% | 19 |
| 70 | 0.4 | 4.8 | 901.1 | 492.5 | 44.16% | 80.95% | 34 |
| 90 | 0.7 | 5.7 | 1324.0 | 734.7 | 41.82% | 73.02% | 46 |
| 100 | 0.7 | 7.8 | 1481.0 | 870.7 | 43.31% | 75.34% | 55 |
| 150 | 1.0 | 9.8 | 4153.1 | 2473.0 | 42.98% | 70.75% | 150 |
| 200 | 1.6 | 12.8 | 8049.3 | 4430.9 | 41.02% | 71.32% | 281 |

其中 r(Box), r(16-CBP)分别表示包围盒、16-CBP 的命中率即用实际模型相交的数量除以包围体检测出来相交的数量. 模型和凸包围多面体是否相交都采用了普通 AABB 树的方式进行判断.

# 主要参考文献 I

[1] M. Teschner, S. Kimmerle, B. Heidelberger, G. Zachmann, L. Raghupathi, A. Fuhrmann, M.-P. Cani, F. Faure, N. Magnenat-Thalmann, W. Strasser, et al., "Collision detection for deformable objects," in Computer Graphics Forum, Wiley Online Library, vol. 24, 2005, pp. 61–81.

[2] J. T. Klosowski, M. Held, J. S. Mitchell, H. Sowizral, and K. Zikan, "Efficient collision detection using bounding volume hierarchies of k-dops," IEEE Transactions on Visualization and Computer Graphics, vol. 4, no. 1, pp. 21–36, 1998.

# 主要参考文献 II

[3] J. L. Bentley, F. P. Preparata, and M. G. Faust, "Approximation algorithms for convex hulls," Communications of the ACM, vol. 25, no. 1, pp. 64－68, 1982.

[4] C. Ericson, Real-time collision detection. San Francisco, CA: Morgan Kaufmann Publishers, 2005.

[5] M. Karlsson, O. Winberg, and T. Larsson, "Parallel construction of bounding volumes," in The Annual Swedish Computer Graphics Association Conference(SIGRAD), 2010, pp. 65－69.

# FAQ

Thank you!