# Convexe Hülle in 3D

Jonas Sorgenfrei – MINF101767/JSO

**Algorithmus** KonvexeHülle(*P*)

*Eingabe*: Menge *P* aus *n* Punkte (x,y,z)

*Ausgabe*: Konvexe Hülle *CH*(*P*) der Punkte in *P*

1. Finde 4 Punkte *p1, p2, p3, p4* in *P* die ein Tetraedern beschreiben

2. C 🡨 *CH*({*p1, p2, p3, p4*})

3. Erstelle eine zufällige Permutation *p5, p6, …, pn* aus den verbleibenden Punkten

4. Initialisiere den Konflikt-Graphen *G* mit allen sichtbaren paaren (*pt, f*), wobei *f* eine Fläche

von C ist und *t* > 4

5. **for** *r* = 5; *r* < *n*; *r*++

6. **do** // Füge *pr* in *C*:

7. **if** *Fconflict(pr)* ist nicht leer //wenn *pr* außerhalb von *C* liegt)

8. **then** Lösche alle Flächen in *Fconflict(pr)* von *C*

9. Erstelle den Horizon, in dem alle Kanten um die Außenseite der sichtbaren Regionen von *pr* (Fläche in *Fconflict(pr)*) in Reihenfolge in eine Liste *L* eingefügt werden

10. **for** alle Kanten in der Horizon Liste (*e ∈ L*)

11. **do** verbinde *e* mit *pr* mittels neuer Dreiecks-Flächen *f*

12. **if** *f* ist co-planar mit seiner Nachbar Fläche *f‘*

auf der anderen Seite von *e* ist

13. **then** merge *f* und *f‘* in eine Fläche, dessen Konflikt-Liste dieselbe ist wie die von *f‘*

14. **else** // Konflikte für *f:* finden

15. Erzeuge neuen Knoten für *f* in *G*.

16. Wobei *f1* und *f2* die Flächen sind, die an der Kante e liegen in der alten konvexen Hülle

17. *P(e)* 🡨 *Pconflict(f1)* U *Pconflict(f2)*

18. **for** alle Punkte *p* *∈* *P(e)*

19. **do** falls *f* sichtbar ist von *p*,

füge *(p,f)* zum Graphen *G* hinzu

20. Lösche den Knoten pr sowie die Knoten der Flächen in *Fconflict(pr)* aus *G*, zusammen mit ihren Konflikt- Verbindungen.

21. **return** C

Horizon – Linie zwischen Sichtbaren und nicht sichtbaren Flächen von Punkt *pr*

Verweise:

Mark de Berg / Otfried Cheong / Marc van Kreveld / Mark Overmars: Computational Geometry, Algorithms and Applications Springer 2008 (3. Aufl.), ISBN 978-3-540-77973-5

https://github.com/jonassorgenfrei/convexHull3D

# Sprachguide

|  |  |
| --- | --- |
| Englisch | Deutsch |
| edge | Kante |
| facet | Fläche(-abschnitt) |
| sphere | Kugel |
| halfspace | Halbraum (durch eine Hypereben begrenzte Teilmenge eines Raums) |
| perpendicular | rechtwinklig |
| halfEdge | Halbkante (kante eines Graph die in eine bestimmte Richtung verläuft, mit der entgegengesetzten Halbkante bilden diese eine Verbindung 2er vertices) |