Der Ball-Pivoting Algorithmus (BPA)

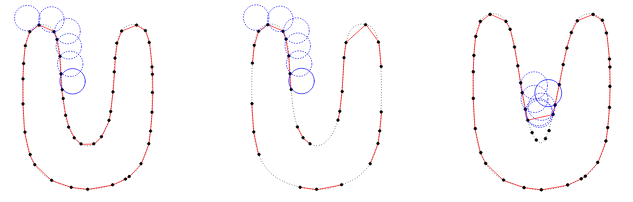
Der Ball-Pivoting Algorithmus (BPA) ist ein Algorithmus zur Oberflächen Rekonstruktion, bei dem man eine Kugel über eine Menge von Punkten rollt, und so die korrekten Faces des Mesh herstellt.

Man beginnt mit einem bekannten Dreieck, in welchem die Kugel liegt. Dabei muss die Kugel alle drei Punkte berühren. Nun wird begonnen die Kugel über eine der Kanten zu rollen. Wenn die Kugel auf einen anderen Punkt trifft, also die Kugel wieder auf allen drei Punkten liegt aber mit keinem anderen Punkt kontakt hat, wurde ein neues Face gefunden. Dieser Vorgang wird für alle Kanten der bekannten Faces wiederholt. Wenn bei einer Kannte kein Punkt gefunden wurde, ist diese eine Rand-Kante und wird entsprechend markiert.

Da das Überprüfen, ob sich ein Punkt mit der Kugel schneidet, bei einem Durchlauf mit besonders vielen Punkten sehr aufwändig wird, werden im BPA die Punkte in Voxel vorsortiert. Da wir in unserem Fall mit einer Menge an Punkten rechnen, die voraussichtlich kleiner als 100 ist, haben wir diesen Teil des Algorithmus nicht implementiert, da der Aufwand nicht gerechtfertigt wird.

Der Vorteil dieses Algorithmus ist, dass er auch komplexere Meshes wiederherstellen kann. Da die Form eines Kreuzgewölbes relativ einfach ist, dürfte das Ergebnis also dementsprechend fehlerfrei ausfallen.

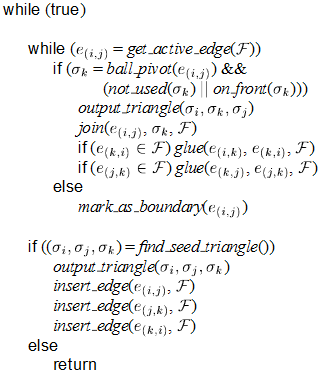
Ein Problem des Algorithmus ist es, den richtigen Radius für die Kugel festzulegen, da wenn dieser zu klein ist, die Kugel mögliche Punkte nicht erreicht. Wenn die Kugel jedoch zu groß ist, kann es sein, dass sie stecken bleibt.



Um Fehler zu vermeiden, wird bei komplexeren Meshes empfohlen, den Algorithmus öfter mit verschiedenen Kugelradien und Startpositionen laufen zu lassen. Was wiederum in unserem Anwendungsfall nicht notwendig sein wird.

Der Ball Pivot Algorithmus ist wie er von IBM erstellt wurde recht umfangreich und undurchsichtig. Zudem gibt es wenige oder nur unzureichende Quellen zu diesem Algorithmus, was eine Implementierung schwierig macht. Aufgrund dessen wurde entschieden, dass der Algorithmus den Umfang des Projektes übersteigt und davon abgesehen diesen zu implementieren. Um den eigentlichen Algorithmus zu testen wurden Testdaten in Form von einer Liste von Dreiecken generiert, was da wir keine realen Testdaten haben sowieso die sinnvollere Methode ist.

Der Pseudocode von IBM:



Quelle: <http://www.research.ibm.com/vistechnology/pdf/bpa_tvcg.pdf>, 22.05.17