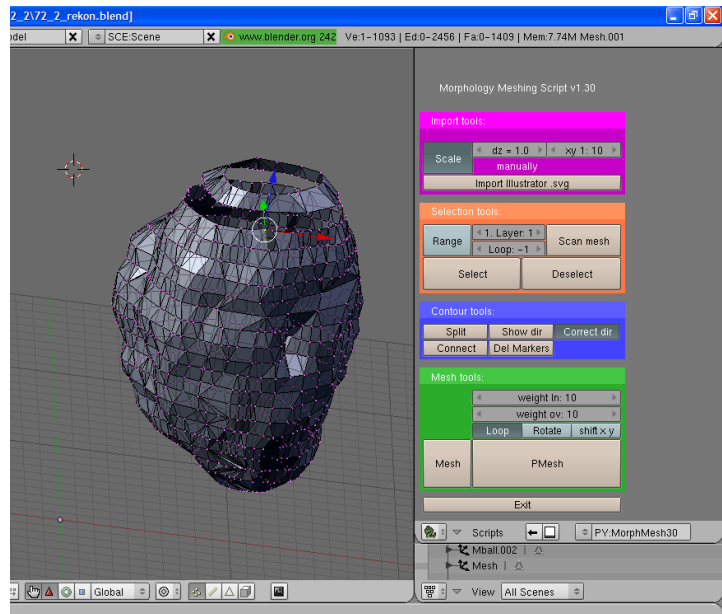


Dokumentation MorphMesh v1.32

MorphMesh ist ein Python Skript für Blender und soll ein Hilfsmittel für die 3D-Rekonstruktion aus Serienschritten sein. Es ist in der Lage eine netzartige Oberfläche zwischen einzelnen Konturen in einem Konturenstapel zu generieren. MorphMesh wurde in erster Line dazu entwickelt, Verbesserungen an mit automatischen Algorithmen (imod) erzeugten Oberflächenmodellen vorzunehmen. Bei diesen treten vielfach an komplizierten Stellen Fehlverbindungen auf, die nur manuell behoben werden können. Morphmesh bietet durch seine Tools und seine einstellbaren Algorithmen die Möglichkeit sehr komplexe Strukturen korrekt zu verbinden. Es ist daneben natürlich auch in der Lage komplette Oberflächenmodelle aus Konturstapeln zu erstellen.



Installation

MorphMesh läuft unter Blender 2.41 und Python 2.4. Wichtig ist, dass Python korrekt installiert ist und die Suchpfade für Moduldateien in den Umgebungsvariablen stimmen.

Unter Linux:

- Python-Konsole starten: `python`
- `>>> import sys`
- `>>> sys.path`
- die angezeigten Pfade werden in einer Konfigurationsdatei (z.B. `bashrc`) in die Umgebungsvariable `$PYTHONPATH` geschrieben (gleiches Format wie bei `$PATH`)
- unter Umständen muss auch noch `$PYTHONHOME` für ein alternatives Suchverzeichnis für Präfix-Dateien gesetzt werden, und zwar mit den Verzeichnissen die
- `>>> sys.prefix` ausgibt

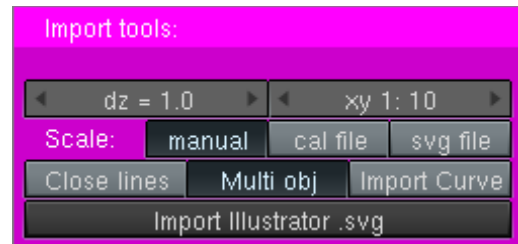
Die Datei `MorphMeshXXX.py` muss einfach nur in blenders Skript-Verzeichnis (`.../blender-2.41/.blender/scripts`) gelegt werden. In blender kann das Skript nun aus einem Scripts-Fenster unter `Scripts>Mesh>MorphMesh` oder aus einem 3D-View-Fenster im Edit Mode unter `Mesh>Scripts>MorphMesh` gestartet werden.

Importieren von Konturen

Blender unterscheidet bei Umrissen zwischen *Curve* (*Bezier*, *NURBS*, *Path*) und *Mesh*. Für die Oberflächenberechnung mit MorphMesh müssen die Konturen als *Mesh* vorliegen. Es besteht die Möglichkeit *Curves* in *Mesh* umzuwandeln. MorphMesh kann sowohl offene

als auch geschlossenen Konturen bearbeiten.

Die Konturen werden üblicherweise von einem alignierten Bilderstapel abgezeichnet. Dies kann erfolgen, indem die Bilder einzeln als Hintergrundbilder oder Texturen in Blender geladen und abgezeichnet werden. Blenders Zeichenfunktionen sind allerdings sehr beschränkt,



sodass dieser Weg in der Regel sehr beschwerlich ist. Meist ist es sinnvoller, diesen Schritt in einem 2D Vektorgrafikprogramm wie Adobe Illustrator oder Inkscape (geplant) zu erledigen. Der Konturenstapel (mit Ebenen) kann dann in Blender importiert werden. Als Austauschformat dient SVG. Im Prinzip kann die als .svg abgespeicherte Datei dann einfach importiert werden. Im Vektorgrafikprogramm können die Konturen in Polygonzüge umgewandelt werden (dies ist die sichere Methode). Dazu in Adobe Illustrator im Menü Objekt>Pfad vereinfachen anklicken und im sich öffnenden Fenster gerade Linien markieren. Die Datei als unkomprimiertes SVG speichern.

Es sind drei Methoden implementiert, um den Ebenenstapel beim Import zu skalieren, Manuell, mit einer Kalibrierungsdatei oder über die Datei direkt. Vor dem Importieren der Datei wird ausgewählt, welche der Methoden angewendet wird. Ist der Button manual gedrückt (Standardwert), werden die Skalierungswerte über die darüber liegenden Eingabefelder bestimmt. Ist der cal file Button gedrückt, kann im Feld darüber eine Kalibrierungsdatei ausgesucht werden. Bei der Option svg file werden die Werte aus der svg. Datei ausgelesen.

1. Manuell: Wenn die Ebenen immer den gleichen Abstand haben, können beim Import ein Skalierungsfaktor für die XY Dimensionen sowie der z-Abstand der Ebenen in Blendereinheiten angegeben werden. Beim Import mit xy 1:1 wird ein Illustrator point in eine Blendereinheit umgerechnet.

2. Mit Kalibrierungsdatei: Haben die Ebenen jeweils unterschiedliche Abstände, ist es sinnvoller eine Textdatei mit den Daten zu erstellen. Diese muss nach folgendem Schema aufgebaut sein: In der ersten Zeile steht xy, in der darauf folgenden der einem Pixel entsprechende Längenwert (z. B. in μm). Dann folgt eine Zeile mit dem Buchstaben z. Danach wird die z-Position für jede Konturebene in eine neue Zeile eingetragen (gleiche Längeneinheit wie xy-Wert!). Die Anzahl der z-Werte muß der Anzahl der Konturebenen entsprechen. In die letzte Zeile wird end

```
xy 1px(blenderU)= xxx (z.B.  $\mu\text{m}$ )
4.8
z height (z.B  $\mu\text{m}$ )
0
2
4
6
8
10
12
14
16
18
20
22
24
26
28
30
32
34
36
38
end
```

Beispiel für eine Kalibrierungsdatei

geschrieben.

3. Mit der .svg Datei: Diese Methode hat den Vorteil, dass die Kalibrierungsdaten direkt in der .svg Datei aufgehoben werden. Dies geschieht über die Ebenennamen und funktioniert so: Zunächst wird irgendwo in die Datei (am besten in die unterste Ebene) eine Linie (Gerade) als Maßstab eingefügt. Diese bekommt die Pfadbezeichnung „xy:100“, falls der Maßstab beispielsweise 100 µm entspricht. Die Ebenen bekommen dann ihre jeweilige absolute Höhe als Namen, also z.B. „z:2.34“ für einen Abstand von 2,34 µm von der untersten Ebene (die natürlich den Namen „z:0“ bekommt). Das Skript liest die jeweiligen Werte beim Importieren ein und legt die Ebenen auf die entsprechenden Höhen. Beim Eintragen der Ebenennamen ist auf die exakte Schreibweise zu achten (immer z: bzw. xy: und Dezimalbrüche immer mit . und nicht mit ,), da die Werte sonst nicht gelesen werden können.

Es stehen noch zwei weitere nützliche Optionen zur Verfügung: `close lines` schließt beim Import alle offenen Konturen, was Überraschungen beim Meshen verhindern kann, da man im Illustrator manchmal leicht übersieht, ob man eine Kontur geschlossen hat. `Multi Obj` erstellt beim Import mehrere Objekte. Nur Konturen mit dem gleichen Farbwert werden in einem Objekt zusammen gefasst. Man kann also im Illustrator bereits Farben für bestimmte Objekte (z.B.) Zellen vergeben, die dann im Blender als getrennte Objekte erscheinen. Das macht gerade erhöht die Handhabbarkeit gerade komplizierterer Konturenstapel während des Rekonstruktionsprozesses deutlich.

Alternativ können Daten von einem anderen Rendering-Programm als VRML importiert werden. Ein Beispiel dazu wird in der [Kurzanleitung Imod](#) beschrieben. Beim Importieren einer VRML-Datei wird automatisch der Typ *Mesh* zugewiesen.

Anwählen der Konturen

MorphMesh bietet verschiedene Möglichkeiten die zu meshenden Konturen zu selektieren:

1. „manuell“: es wird ein jeweils ein Vertex auf den zu meshenden Konturen gewählt.
2. Selection tools: Mit Hilfe dieser Funktionen können bestimmte Konturen (Loops) selektiert bzw. deselektiert werden. Bevor diese Werkzeuge verwendet werden können ist es notwendig einmal alle Konturen im Mesh erkennen zu lassen. Dies ist ein sehr rechenzeitintensiver Schritt, der ansonsten immer nur an den Ebenen, an denen gerade gearbeitet wird ausgeführt wird. Durch Drücken der Taste Scan Mesh werden alle Konturen im Modell erfasst. Nun kann mit den Selektionswerkzeugen gearbeitet werden. Es gibt 2 Modi. Im normalen Modus (Range nicht gedrückt) wird mit dem oberen „Regler“ die Ebene gewählt, mit dem unteren die Kontur. Select selektiert Kontur x auf Ebene y. Im Range Modus können jeweils die ersten Konturen (Loop 0) von der im unteren Slider bis zu der im oberen Slider eingestellten Ebene selektiert werden.



Spezielle Vorbehandlungen: Contour tools

Bei der Rekonstruktion morphologischer Strukturen ist man oft mit dem sog. Tiling-Problem konfrontiert: Eine Struktur verzweigt sich von einer Ebene zur nächsten bzw. zwei Anschnitte verbinden sich zu einem einzigen.

Für die meisten in automatischen Rendering Programmen eingesetzten Algorithmen stellt dies ein Problem dar, da vielfach keine guten Kriterien angebracht werden können, welche Teile in der



einzelnen Struktur mit welchem Anschnitt in der nächsten Ebene verbunden werden sollen. Dem Bearbeiter ist dies allerdings meist ziemlich klar. Daher bietet MorphMesh zwei Tools für die manuelle Bearbeitung solcher Problemstellen an:

Das Split-Tool kann eine Kontur in zwei Konturen aufteilen. Diese können im Anschluss mit der jeweils korrespondierenden Kontur verbunden werden. Natürlich können die erhaltenen Konturen auch jeweils noch weiter zerteilt werden. Soll die zerteilte Struktur in der anderen Richtung verbunden werden und wieder als ganze Kontur erkannt werden, wird einfach ein remove doubles durchgeführt.

Das Connect-Tool geht das Problem genau umgekehrt an: es verbindet zwei Konturen zu einer, die dann mit einer auf der nächsten Ebene gemesht werden kann. Sollen die verbundenen Strukturen wieder als einzelne fungieren, müssen die Verbindungen manuell entfernt werden.

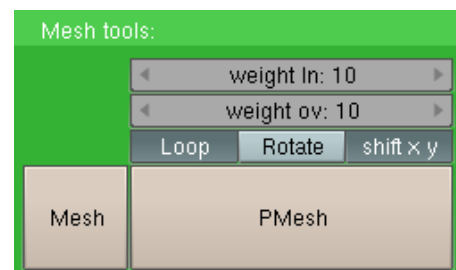
Show direction

Hierzu ist es notwendig etwas in die Datenstrukturen von Blender und MorphMesh einzutauchen. Blender betrachtet den gesamten Konturenstapel als ein Mesh. Eine Kontur ist nichts anderes als eine Liste von Punkten, die miteinander verbunden sind. Je nachdem in welcher Reihenfolge die Punkte in der Liste stehen, kann eine Kontur rechts- oder linksdrehend sein. MorphMesh versucht standardmäßig alle Konturen auf rechtsdrehend zu vereinheitlichen. Dies kann allerdings bei komplizierten Umrissen fehlschlagen, so dass einzelne Konturen linksdrehend sind. Die Meshalgorithmen sind, wie im weiter unten beschrieben wird, richtungsabhängig, d.h. der Versuch eine rechts- und eine linksdrehende Kontur zu verbinden schlägt in aller Regel fehl. Das Showdirection tool zeigt die Richtung der markierten Konturen an, indem es die ersten beiden Punkte selektiert und dort einen Richtungspfeil erzeugt. Die Pfeile können mit del Markers wieder aus der Szene entfernt werden. Weist eine Kontur nicht die gewünschte Richtung auf, kann diese mit invert dir umgekehrt werden. Hierzu muss natürlich die Autodetect funktion deaktiviert werden. Nach Invertieren wird die neue Drehrichtung mit einem Richtungspfeil angezeigt.

Mesh tools

MorphMesh bietet zwei Algorithmen zur Oberflächenberechnung an: Mesh und PMesh.

Mesh ist ein relativ einfacher, wenig rechenintensiver Algorithmus, der recht gut für offene, nicht zu komplexe Konturen mit ähnlicher Punktzahl funktioniert, wie sie z.B. das Programm Imod erzeugt.



PMesh ist der Algorithmus mit dem sehr komplexe Konturen miteinander verbunden werden können. PMesh verwendet drei gewichtete Parameter um Auszuwählen, welche Punkte von zwei Konturen miteinander verbunden werden.

Abstand zwischen den Punkten auf beiden Konturen: PMesh sucht von einem Punkt aus den nächsten Punkt auf der anderen Kontur. Dazu wird die Distanz zwischen dem Startpunkt und allen noch nicht verbundenen Punkten auf der anderen Kontur berechnet.

Abstand zwischen dem letzten und dem neu zu verbindenden Punkt: MorphMesh kann beim Verbinden Punkte überspringen. Dies ermöglicht, dass Bereiche ausgeglichen werden in denen eine Kontur wenige und die andere Kontur sehr viele Punkte enthält. Die übersprungenen Punkte werden später mit einem Mesh verbunden. Damit der Algorithmus beim Überspringen von Punkten nicht ganze Konturbereiche auslässt, weil z. B. zwei Punkte auf gegenüber liegenden Seiten einer Kontur sehr nah beieinander liegen, wird die Distanz zwischen dem letzten und dem neu zu verbindenden Punkt berechnet. Der Regler `weight ln` stellt das Gewicht dieses Wertes in der Gesamtberechnung ein.

Unterschied in der Richtung der Kontur: Bei Überschneidungen von Konturen können Punkte von gegenüberliegenden Konturseiten sehr nah beieinander liegen. Durch einen gewichteten Parameter, der die Richtung des Konturverlaufs berechnet kann eine solche Fehlverbindung verhindert auch bei dicht aufeinander folgenden Punkten verhindert werden. Der Regler `weight ov` stellt das Gewicht dieses Wertes in der Gesamtberechnung ein.

Die beiden Regler geben jeweils das Gewicht des Parameters in Prozent an. Das Gewicht des ersten Parameters (Abstand zwischen den Punkten auf beiden Konturen) ist die Differenz von 100 minus der Prozentwerte der anderen Parameter.

Der Schalter `Loop` bestimmt, ob die Konturen als geschlossene Loops betrachtet werden sollen. Derzeit ist nur diese Einstellung sinnvoll, offene Flächen werden noch nicht richtig gemesht.

Mit `Rotate` wird bestimmt, ob ein markierter Punkt auf jeder Kontur als Startpunkt zum meshen genutzt werden soll, oder jeweil die zwei Punkte mit dem geringsten Abstand als Startpunkte ausgewählt werden.

Der Schalter `shift x y` schiebt die markierten Konturen temporär übereinander, so dass Konturen, die aufgrund des Alignments gegeneinander verschoben sind, besser gemesht werden.

