

Gingolds “A Direct Texture Placement and Editing Interface”

Katerina Tzikas

Proseminar Mensch-Computer-Interaktion, Universität Ulm

Katerina.Tzikas@uni-ulm.de

Abstract. Die 3D-Computergrafik bekommt in zahlreichen Bereichen der Medien eine immer tragendere Rolle zugesprochen. Hierfür verwendete Software hat somit ein hohes Potential weiter erforscht und –entwickelt zu werden.

Ich gehe im Folgenden zusammenfassend auf den aktuellen Stand gebräuchlicher 3D-Software ein und stelle auf dieser Basis kürzlich veröffentlichte Forschungsergebnisse zum Thema „Grafische Benutzerschnittstellen“ von Yotam I. Gingold vor.

Gingolds Beitrag „A Direct Texture Placement and Editing Interface“ behandelt das Anlegen und Modellieren von Texturen auf dreidimensionalen Oberflächen mit Fokus auf Eingabemedium und Benutzerschnittstelle.

Die Software zielt auf direktes Anlegen und Bearbeiten von Texturen auf 3D-Objekte und ist kompatibel zu mehrhändigen Eingabemedien.

1 Grundlagen und Begriffe der 3D-Computergrafik

Die Erstellung eines dreidimensionalen Objekts erfolgt in der Regel dreistufig. Die erste Stufe beinhaltet das Formen eines Objekts, die zweite das Hinzufügen von Oberflächeninformationen und die dritte die Umrechnung der dreidimensionalen Bilddaten in eine Rastergrafik.

1.1 Erstellen und Formen eines Objekts

Zur Erstellung eines neuen Objekts werden in der Regel (mehrere) geometrische Figuren als Grundlage, meist Kugeln oder Würfel, verwendet. Diese Figuren liegen unmittelbar nach ihrer Erstellung in *Parameterdarstellung* vor. Durch Ändern der Parameterwerte des Objekts lassen sich Eigenschaften wie Höhe und Radius editieren. Möchte man nun die dem Objekt zugrunde liegende Form weitergehend verändern, muss es von der mathematisch eindeutigen Parameterdarstellung in *Polygonform* gebracht werden. Dies bedeutet, das Objekt wird nicht mehr durch eine mathematisch eindeutige Gleichung, sondern durch viele einzelne Punkte beschrieben, welche derart miteinander verbunden sind, dass ihre Kanten Polygone (meist Rechtecke oder Dreiecke) bilden [3]. Diesen Vorgang nennt man speziell bei Dreiecken *Triangulation*. Bei der vom Programm automatisch durchgeführten Umrechnung ist

2 Katerina Tzikas

es wichtig, dass die Polygone *planar* sind, das heißt die Punkte eines jeden Polygons müssen für sich genommen auf einer Ebene liegen. Dem Benutzer erscheint die Oberfläche des Objekts nun wie ein dreidimensionales Gitternetz (engl. *mesh*). Durch Anwählen von Polygonen und mit Hilfe verschiedener Werkzeuge kann der Benutzer nun die Lage der Polygone zueinander verändern, das Objekt also frei deformieren. Dies ist möglich, da die Polygone intern unabhängig voneinander dargestellt werden. Das heißt, jedes Polygon hat sein eigenes dreidimensionales Koordinatensystem. Da die Polygone planar sind, reichen zur Beschreibung der einzelnen Polygone zwei Achsen aus. Die dritte Achse ist einer Normalen vorbehalten, die angibt welche Seite des Polygons sichtbar ist, also „oben“ liegt [3].

Geläufige Werkzeuge zur Deformation bewirken zum Beispiel Skalierung, Spiegelung, Verschmelzung und Wölbung.

1.2 Texture-Mapping

Unter dem Begriff *Texture-Mapping* versteht man in der 3D-Computergrafik das Anbringen der Oberflächenstruktur an ein dreidimensionales Objekt. Es gibt verschiedene spezielle Formen des Texture-Mappings wie zB. *Bumpmapping*, *Normalmapping* und *Displacement Mapping*.

Beim Bumpmapping werden dem Objekt die Oberflächeninformationen in Form eines Bildes „aufgeklebt“. Eine Oberflächenstruktur wird somit nur suggeriert. Diese Form des Mappings erfordert wenig Rechenaufwand [2].

Beim Displacement Mapping hingegen wird die Form des Objekts tatsächlich verändert. Hierfür muss eine sogenannte *Height-Map* angefertigt werden, bestehend aus Graustufenwerten. Diese Height-Map wird über die Oberfläche des Objekts gezogen. Der jeweilige Graustufenwert gibt dabei an wie weit ein Punkt auf der Oberfläche hervorstehen soll [2].

Beim Normalmapping werden vom Benutzer zwei Modelle erstellt, ein detailreiches mit vielen Informationen und ein detailarmes, das dementsprechend eine niedrigere Auflösung hat. Die Informationen vom hochauflösenden Objekt werden auf das detailarme Objekt mittels einer *Textur* übertragen, sodass dieses optisch wie das hochauflösende Objekt wirkt, jedoch weniger Rechenleistung in Anspruch nimmt [2]. Der Hauptteil dieser Ausarbeitung beschäftigt sich in erster Linie mit der interaktiven Anpassung einer vorhandenen Textur an ein dreidimensionales Objekt.

1.3 Rendering

Ist ein dreidimensionales Objekt modelliert, muss es zur Fertigstellung gerendert werden. Der Begriff *Rendering* bezeichnet einen mathematischen Prozess, der vom Programm mit gegebenen Daten automatisch durchgeführt wird.

Allgemein lässt sich sagen, dass beim Rendering aus den dreidimensionalen Bilddaten eine *Rastergrafik* erstellt wird. Das Bild verliert dabei seine stereoskopische Wirkung und ist somit nicht mehr „wirklich dreidimensional“. Zum Rendering zählen Verdeckungsrechnung, also welcher Teil eines Objekts durch ein zweites verdeckt wird, Licht- und Schattenberechnung, sowie Texture-Mapping [3].

2 „A Direct Texture Placement and Editing Interface“

Wie weiter oben schon beschrieben, erfordert die Modellierung eines dreidimensionalen Objekts das Formen, die Angabe von Oberflächeneigenschaften und die Berechnung derselbigen. Wir konzentrieren uns im Folgenden auf das Hinzufügen von Informationen mittels eines zweidimensionalen Bildes.

Hierzu muss der Benutzer parallel zu der 3D-Objekt-Erstellung mittels eines Zeichenprogramms ein zweidimensionales Bild erstellen, welches dann dem Objekt als Oberflächeneigenschaft hinzugefügt werden kann.

Für den Benutzer bedeutet dies in der Regel einen Wechsel zwischen zwei Ansichten: dem dreidimensionalen und dem zweidimensionalen Raum.

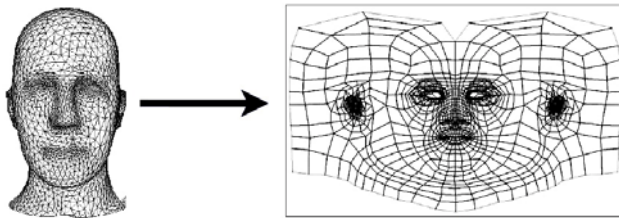


Fig. 1. Umdenken vom 3D-Objekt zur 2D-Ansicht [1]

Die „Plättung“ eines 3D-Objekts beinhaltet zwangsläufig einen gewissen Grad an Verzerrung (Fig.1). Der Benutzer muss sich beim Anfertigen seines Bildes also vorstellen wie dieses als Überzug auf seinem 3D-Objekt aussehen würde und es dementsprechend anfertigen, um beim Endergebnis sichtbare Verzerrungen auf der Oberfläche zu vermeiden.

Dies erfordert eine gewisse Vorstellungsgabe und Erfahrung im Umgang mit der Software und ist somit nicht intuitiv.

Gingold ist von dieser Problematik ausgegangen und hat eine neue Eingabemethode entwickelt. Die Idee ist, direkt am 3D-Modell arbeiten zu können, das als Textur fungierende Bild also auf das Objekt auftragen und es direkt auf dem Objekt anpassen und modellieren zu können.

Ich gehe nun genauer auf die einzelnen Eigenschaften der von Gingold entwickelten Software ein, beginnend mit dem Eingabemedium, fortfahrend mit den jeweiligen Modellierungsmodi des Programms.

2.1 Eingabemedium

Gingolds Programm wurde auf einem sogenannten *Drafting Table* implementiert, eine Art drucksensitivem Berührungsbildschirm (engl. *Touch-Screen*). Dieses wird als *Multi-Touch-Interface* klassifiziert, was heißt, dass man darauf mit mehreren Händen gleichzeitig arbeiten kann.

Das *Drafting Table* ist intuitiv nutzbar wie ein herkömmliches Zeichenbrett. Es registriert jede Berührung sowie die Stärke des Drucks, der auf den Berührungspunkt ausgeübt wird. Die daraus resultierenden Daten sind essentiell für das Oberflächen-

modellierungsverfahren, welches dem realen Vorgang des Formens und Zeichnens einer Objektoberfläche nachempfunden sein soll. Wie diese Intention im Programm mittels unterschiedlicher Werkzeuge gestützt wird soll nun beschrieben werden.

2.2 Positionierung

Wir überspringen den Teil der Objekterstellung und gehen davon aus, dass das Gitternetz eines beliebigen 3D-Objekts bereits zur Verfügung steht.

Um das Objekt nun bearbeiten zu können, ist es notwendig seine gesamte Fläche betrachten zu können. Da das Objekt jedoch dreidimensional abgebildet wird, ist es unvermeidbar, dass es Stellen gibt, die verdeckt bleiben und infolgedessen für den Betrachter nicht sichtbar sind. Das Objekt muss also beweglich gemacht werden.

Für eine erste Positionierung und spätere Betrachtung des Objekts bietet Gingolds Programm einen speziellen Modus. Wechselt man in diesen Modus, wird das Objekt mit einer scheibenförmigen transparenten Platte hinterlegt. Durch Druck mit bis zu drei Fingern auf die Platte lässt sich das Objekt in der 3D-Ansicht beliebig drehen.

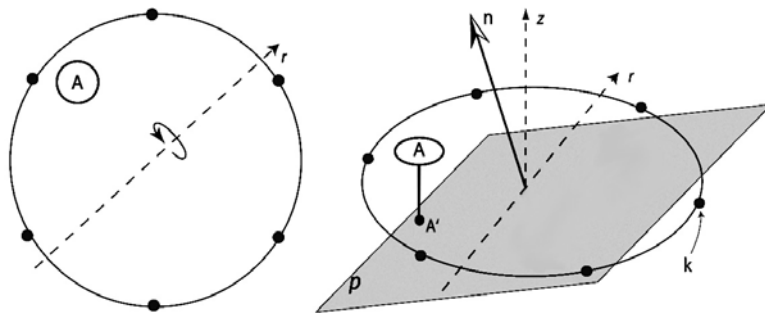


Fig. 2. Darstellung der Rotationsplatte mit Anfangspunkt A und Zielpunkt A' [1].

Der Druck auf die Platte wird als Rotation um eine Achse interpretiert, wobei die Stärke des Drucks den Höhenunterschied zwischen Anfangspunkt und Zielpunkt und somit den Rotationswinkel bestimmt (Fig.2). Die Normale auf der Platte in Figur 2 symbolisiert die Position des zu rotierenden Objekts. Durch Drücken und Ziehen eines einzelnen Punktes lässt sich das Objekt verschieben. Berührt man die Platte an zwei Punkten gleichzeitig und zieht diese auseinander oder führt sie zusammen, so wird in das Objekt hinaus- bzw. hineingezoomt.

Außerhalb dieses Modus ist die Position des Objekts fixiert. Der Blickwinkel ist durch Drücken und Ziehen eines einzelnen Punktes veränderbar. Wir stellen uns vor, das Bild auf dem Monitor gleiche der Übertragung einer Kamera, deren Position sich auf einem festen Radius um das Objekt herum verändern lässt, mit der wir jedoch nicht zoomen können.

2.3 Texturplatzierung und Skalierung

Um dem Objekt nach erfolgreicher Platzierung ein vorhandenes Bild als Textur zuzuweisen, muss eine *Parametrisierung* stattfinden, eine Umrechnung des Objekts von der Polygonform in Parameterdarstellung, sodass die Punkte des Bildes Punkten auf dem Gitternetz zugeordnet werden können.

In Gingolds System erfolgt die Parametrisierung durchgehend automatisch. Der Benutzer bleibt infolgedessen stets in der 3D-Ansicht und kann das Bild direkt auf dem 3D-Objekt positionieren.

Das Verschieben des geladenen Bildes auf dem Objekt erfolgt durch *Drag-And-Drop*, Drücken und Halten eines Punktes auf dem Bild, solange dessen Position geändert werden soll.

Da ein unabhängig von dem 3D-Objekt erstelltes Bild selten die richtige Auflösung aufweist, enthält das Programm ein Skalierungswerkzeug mit dem sich die Größe des Bildes der des Objekts anpassen lässt.

2.4 Dehnung und Stauchung

Wurde die Textur platziert, lässt sie sich im nächsten Schritt mit Hilfe von Fixpunkten deformieren. Die Fixpunkte kann man sich hierbei vorstellen wie Stecknadeln auf einem dehnbaren Stoff. Im englischen werden sie *Constraints* genannt. Es lassen sich beliebig viele Constraints zum Bearbeiten der Textur einfügen, gleichzeitig zur Deformation oder zur gegenseitigen Beeinflussung nutzen.

Ist solch ein Constraint an einem beliebigen Punkt eingefügt, kann man die Textur durch Verschieben dieses Punktes deformieren. Das Bild verhält sich dabei wie ein elastisches Material. Entfernt man eines der Constraints, so reagiert die Textur wie bei einem „Undo“ in herkömmlichen Programmen: sie springt zurück zu ihrer ursprünglichen Position. Der Einfluss weiterer im Bild enthaltener Constraints wird hierbei berücksichtigt. Auf diese Weise lässt sich relativ einfach mit der Textur hantieren, in etwa so, wie man es in der realen Welt mit einem Modell und einem zu fixierenden Überzug handhaben würde.

Die Constraints lassen sich nach erfolgreicher Bearbeitung schließlich einfrieren, das heißt die aktuelle Form und Position des Bildes wird als neue Grundform übernommen und man kann den Arbeitsvorgang fortsetzen, die Textur erneut deformieren oder in andere Bearbeitungsmodi übergehen.

2.5 Mehrere Bildquellen für eine Textur

Ein möglicher nächster Schritt der Bearbeitung ist das Hinzufügen eines weiteren Bildes in Form einer zweiten Bildebene (engl. *Layer*, ähnlich verwendet wie in manchen Bildbearbeitungsprogrammen). Diese kann getrennt von der ersten Bildebene modifiziert oder mit der anderen Bildebene verschmolzen werden.

Hat man seinem Objekt zwei oder mehr Bilder hinzugefügt, lässt sich nun das *Blendwerkzeug* benutzen, um eine Bildfläche in die andere zu projizieren. Dies geschieht indem die Transparenz, der sogenannte *Alpha-Kanal-Wert*, der angewählten

Fläche erhöht wird, sodass die sich darunter befindliche Fläche zum Vorschein kommt.

2.6 Regulierung der Einwirkungsfläche

Für noch detailliertere Vorgehensweise bietet Gingolds Verfahren ein Klebewerkzeug (engl. *glue*). Dieses kann auf der Oberfläche wie ein Klebestift verwendet werden und dient als virtuelle Trennwand zwischen den zu bearbeitenden Flächen. Auf diese Weise kann sichergestellt werden, dass schon fertig modifizierte Teilflächen unberührt bleiben.

Eine ähnliche Herangehensweise bietet das Radiuswerkzeug, mit dessen Hilfe man einen kreisförmigen Bereich beliebiger Größe auswählen kann, der das zu bearbeitende Gebiet eingrenzt. Dank des Multi-Touch-Interfaces lassen sich Justierung des Kreisradius und Positionierung des Werkzeuges auf dem Bild zeitgleich durchführen.

3. Fazit

Gingolds Intention war es ein User-Interface zur 3D-Texturmodellierung zu entwickeln, welches durch realitätsnahe Bildanzeige und Textureditierung unmittelbar auf dem Modell intuitiv nutzbar ist. Hierzu wurde das Programm auf einem Drafting-Table implementiert und der Anzeigenwechsel zwischen 2D und 3D komplett vermieden.

Ob unerfahrene Nutzer jedoch Zugang zu diesem Programm finden und es tatsächlich als intuitiv besser nutzbar im Vergleich zu herkömmlichen Programmen bewerten, ist unklar, denn eine Nutzerevaluation hat nicht stattgefunden. Desweiteren ist es fraglich inwiefern weitere Programmeigenschaften in dieses System integriert werden können, denn aktuelle 3D-Modellierungsprogramme weisen weit mehr Features auf als Gingold in seinem Programm integriert hat.

Insgesamt lässt sich sagen, dass Gingolds Entwicklung einen guten Ansatz für den Gebrauch von Multi-Touch-Medien, insbesondere im Bereich der 3D-Computergrafik, bietet.

4. Quellenangaben

1. Y. I. Gingold, P. L. Davidson, J. Y. Han, D. Zorin: A Direct Texture Placement and Editing Interface. <http://www.mrl.nyu.edu/~gingold/texslide> (05.05.2007).
2. A. Watt: 3D Computer Graphics. Addison-Wesley (2000).
3. R. J. Wolfe: 3D Graphics A Visual Approach. Oxford University Press (2000).