

Silhouettenbasiertes 3D-Scannen

Marcel Lancelle 2002-2003, Institut für Computer Graphik, TU Braunschweig

Inhalt

1 Einleitung.....	1
2 Hardware.....	1
3 Software.....	3
4 Ergebnisse.....	10
5 Vergleich und mögliche Verbesserungen.....	13
6 Referenzen.....	14

1 Einleitung

Die Erstellung von 3D Modellen ist ein wichtiger Schritt in der Computergraphik. Mit heutigen Modellierungswerkzeugen ist dies für manche Objekte einfach zu erfüllen, doch gibt es auch viele Objekte, bei denen eine Modellierung sehr aufwendig wäre. 3D-Scanner helfen hier weiter, sofern das Objekt schon existiert und verfügbar ist. Bei jedem der vielen Verfahren des 3D Scannens gibt es Einschränkungen bezüglich der Größe, Form, Konsistenz oder anderen Materialeigenschaften des Objektes. Doch das 3D-Scannen wird häufig einfach aus finanziellen Gründen nicht eingesetzt.

Das Ziel dieser Studienarbeit war die Konstruktion einer einfachen und preisgünstigen Hardware sowie die Erstellung der Software. Das Verfahren "shape from silhouette" sollte benutzt und für automatisierte Aufnahmen ein Drehteller konstruiert werden.

Dabei wird das Objekt aus verschiedenen Blickwinkeln vor einem einfarbigen Hintergrund fotografiert und die Anteile jedes Fotos außerhalb des Objektes aus einem soliden Block ausgestanzt. Dadurch wird allerdings nur die konvexe Hülle des Objekts rekonstruiert, was der größte Nachteil dieses Ansatzes ist.

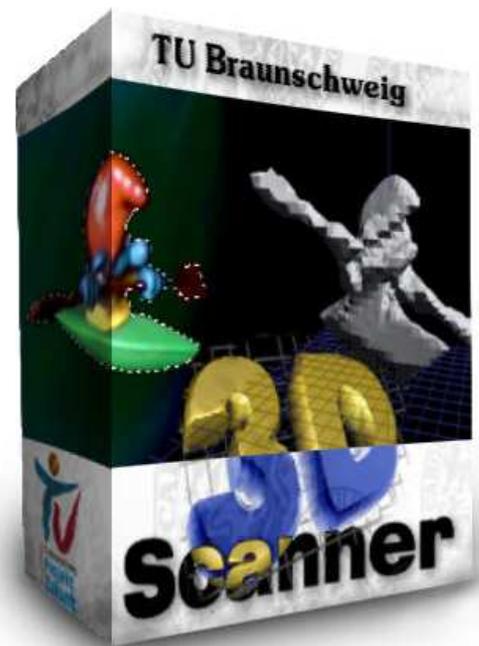
Für die automatische Segmentierung der Bilder darf das Objekt auch keine Farben ähnlich der Hintergrundfarbe enthalten bzw. diese stark reflektieren oder transparent durchscheinend sein.

Ein Vorteil ist die Möglichkeit der manuellen Korrektur der Segmentierung, was prinzipiell das Scannen von Objekten mit beliebiger Oberfläche erlaubt. Ebenfalls vorteilhaft sind die geringen Hardwarekosten. Mit einiger manueller Arbeit ist sogar eine grobe Rekonstruktion eines Objekts mit Hilfe von normalen, unkalibrierten Fotos möglich.

2 Hardware

Für die automatisierte Aufnahme kleinerer Objekte aus mehreren Blickwinkeln besteht die einfachste Möglichkeit darin, das Objekt auf einem Drehteller zu drehen und dabei mehrere Aufnahmen mit einer oder mehreren digitalen Fotokameras zu machen.

Weitere Ideen, die Kamera z.B. auf einer Schiene um das Objekt zu bewegen, wurden früh wegen zu erwarteten mechanischen Schwierigkeiten und hohem Aufwand verworfen.

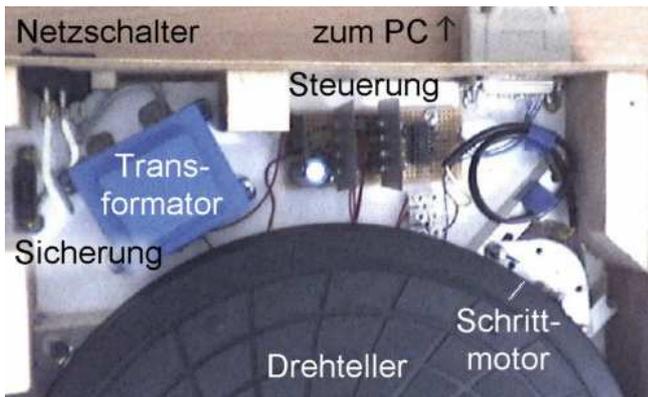


2.1 Drehteller-Mechanik

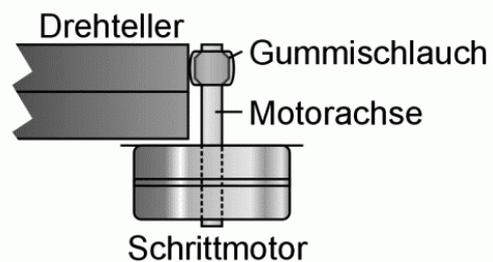
Als Drehteller wurde ein handelsüblicher drehender Untersatz (für Fernseher etc.) mit einem Durchmesser von ca. 30 cm gewählt, der einer Belastung von 100 kg standhält. Der Antrieb erfolgt über einen Schrittmotor, da die Achse dort um definierte Winkel gedreht werden kann.

Die Kraftübertragung geschieht durch ein Laufrad, das direkt auf einer Verlängerung der Motorachse befestigt ist. Dies ist wegen des Schlupfs nicht optimal, jedoch hinreichend genau (siehe Kapitel 4.1). Der Anpreßdruck kann mit einer Flügelschraube justiert werden. Die vorerst geplante Benutzung von Zahnriemen scheiterte am nicht zu sinnvollen Kosten zu beschaffenden Material.

Die Drehscheibe ist zusammen mit Elektronik und Motor in einem kleinen Gehäuse untergebracht.



Mechanik - Seitenansicht

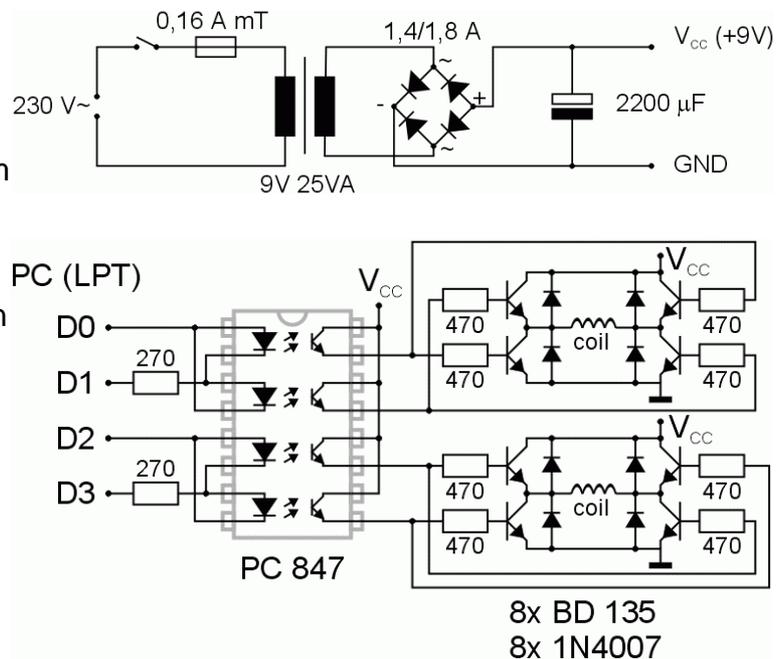


Blick in das geöffnete Gehäuse von oben; Seitenansicht der Mechanik

2.2 Drehteller-Elektronik

Der Motor ist ein bipolarer Schrittmotor. Der Parallelport wird für die Ansteuerung benutzt, um die Elektronik möglichst einfach zu halten. Auch das Timing wird direkt vom PC übernommen, so kann auf den Einsatz eines Mikrocontrollers verzichtet werden. Zum Schutz ist der PC über Optokoppler galvanisch von der restlichen Schrittmotorelektronik getrennt.

Jede der beiden Windungen wird über eine H-Brücke mit etwa $\pm 9V$ versorgt. Dioden schützen die Leistungstransistoren vor Induktionsspitzen. Die Verlustleistung der Leistungstransistoren wird durch elektrisch isolierte Kühlbleche abgeführt.



2.3 Kamera

Webcams sind heutzutage schon für 10 EUR zu erhalten. Diese sehr günstigen Kameras sind mit CMOS-Bildsensoren ausgestattet deren Bilder stark verrauscht sind und deren Farben zu wünschen übrig lassen. Manuelle Korrektur der Segmentierung ist bei deren Einsatz beim derzeitigen Segmentierungsalgorithmus wahrscheinlich nötig.

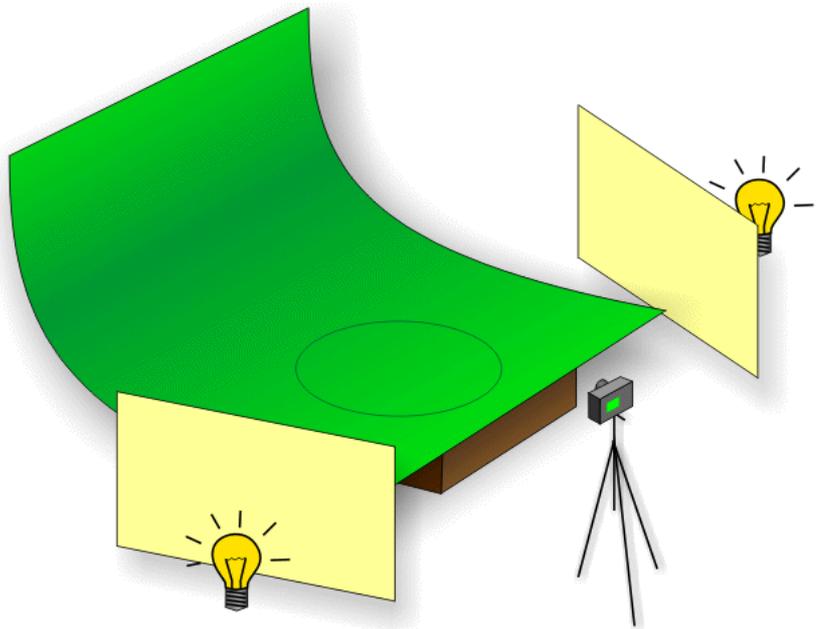
Gute Ergebnisse sind bereits mit relativ günstigen CCD-Kameras zu erreichen. Eine Auflösung von 640x480 ist dabei meistens schon ausreichend. Die Verzerrungen durch die Kunststofflinsen der billigen Webcams sind kein großes Problem, da die Software diese größtenteils korrigiert.

Prinzipiell ist auch der Einsatz einer Analogkamera mit anschließendem Scannen möglich, jedoch viel aufwendiger.

Ein Stativ ist für die einfache Ausrichtung der Kamera sehr nützlich.

2.4 Hintergrund und Beleuchtung

Für die hier benutzte automatische Segmentierung ist ein einfarbiger Hintergrund notwendig. Bunter Filz eignet sich besonders gut; bei manchen Objekten kann auch schwarzer Stoff benutzt werden. Gegenwärtig wird hellgrüner Filz benutzt. Der Hintergrund sollte möglichst gleichmäßig ausgeleuchtet werden, insbesondere sollte auch der Schattenwurf weitestgehend vermieden werden. Eine großflächige, diffuse Beleuchtung von allen Seiten wäre optimal.



2.5 Kosten

Mindestens benötigt werden ein PC und ein Fotoapparat. Drehteller, Stativ, einfarbiger Hintergrund und gute Beleuchtung sind hilfreich. Je nach der Bereitschaft zur Improvisation und Eigenarbeit sowie bereits vorhandener Hardware kann ein extrem günstiges System zusammengestellt werden.

3 Software

Zur Volumenrekonstruktion bieten sich zwei Verfahren an. Zum einen kann die Segmentierung einen Linienzug liefern, mit denen Pyramiden erstellt und mit CSG kombiniert werden. Die andere häufig benutzte Methode, die auch hier verwendet wurde, erstellt einen Voxelaum der anschließend trianguliert wird.

3.1 Drehtelleransteuerung

Die Ansteuerung des Parallelports erfolgt direkt über die Ports. Unter Windows 2000 bzw. Windows NT wird der Treiber PortTalk benutzt, um diesen Zugriff zu erlauben. Installations- und deinstallationsroutinen für den Treiber sind ebenfalls eingebaut.

Eine Kalibrierung der Schrittzahl wird vorgenommen, indem die Schritte für eine Umdrehung gezählt werden. Für eine sanfte Beschleunigung wird eine Drehzahlrampe gefahren, die den Motor linear bis zur Maximalgeschwindigkeit beschleunigt.

3.2 Bildaufnahme

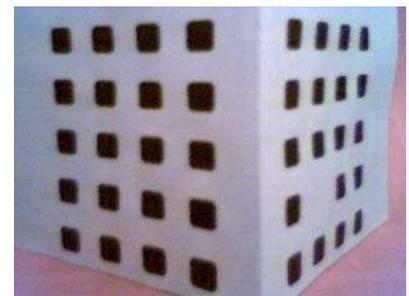
Die Bilder können über die TWAIN-Schnittstelle aufgenommen oder aus einer Datei geladen werden.

Die TWAIN Schnittstelle ist die gebräuchlichste Methode, um (Stand-)bilder aufzunehmen. Leider unterstützen nicht alle Treiber die praktische Funktion, Bilder automatisch (ohne Benutzereingriff) aufzunehmen.

Als Alternative wurde daher auch die Video for Windows (VfW) Schnittstelle in Erwägung gezogen. Die gleichzeitige Aufnahme mit mehreren Kameras ist prinzipiell möglich, wird von der Software aber noch nicht unterstützt.

3.3 Kamerakalibrierung, Koordinatentransformation

Vom Kalibrierungsobjekt werden zwei Aufnahmen aus unterschiedlichen Blickwinkeln benutzt, um alle notwendigen Kameraparameter sowie die Position der Drehachse zu berechnen. Diese Berechnung ist jedoch wegen Ungenauigkeiten nicht sehr präzise, so daß ebenfalls die Möglichkeit besteht, die Kante des Kalibrierungsobjekts als Drehachse zu definieren. Dabei wird nur eine Aufnahme des ausgerichteten Kalibrierungsobjekts benötigt.



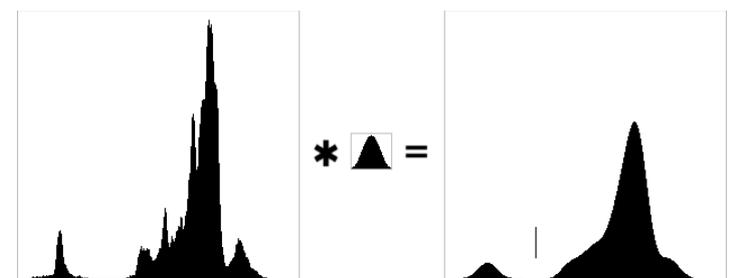
typische Aufnahme eines Kalibrierungsobjektes

3.3.1 Segmentierung

Eine getestete Methode wird im folgenden beschrieben.

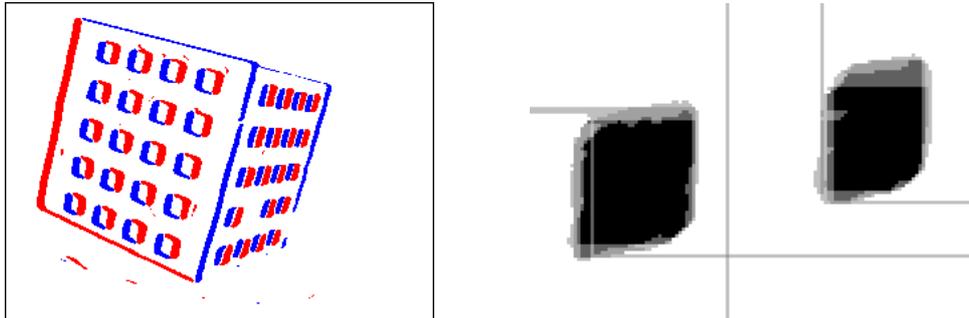
Jede Aufnahme wird erst mit einem Medianfilter bearbeitet und anschließend in Grauwerte umgewandelt. Helle und gesättigte Pixel werden zu hellen, dunkle zu dunklen Pixeln. Nun wird das Bild binarisiert, indem als Schwellenwert die Position des ersten Minimums des geglätteten Histogramms gewählt wird. Die Glättung erfolgt durch Faltung mit einem einer Gaußfunktion ähnlichen Kernel. Ein anschließendes dilatieren stellt sicher, daß ein ausgefranster Rand der Objekte keine Probleme bereitet. Diese Methode

funktionierte jedoch nicht mit allen Aufnahmen, da die Farbwerte der schwarzen Punkte teilweise schon relativ stark gesättigt und unterschiedlich hell sind. Die Grafik zeigt ein Histogramm, bei dem die bimodale Entscheidung einfach und erfolgreich getroffen wurde.



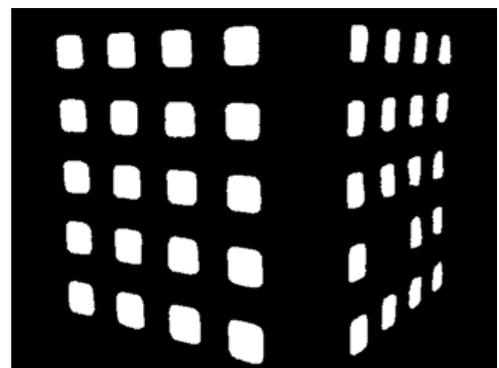
Schwellenwert (durch Linie gekennzeichnet) zum Binarisieren an erstem Minimum des geglätteten Histogramms

Als besser hat sich jedoch folgender Ansatz erwiesen, den ich für eine frühere Realtime-Anwendung entwickelte. Das Bild wird dabei jeweils von oben, unten, rechts und links Linie für Linie durchlaufen. In einem Array mit Bildgröße werden dabei die Werte an den Positionen inkrementiert, die auf eine weiß-schwarz-Kante folgen.



Links: Durchlauf von links nach rechts: blau kennzeichnet Anfang, rot Ende eines schwarzen Bereichs des Ursprungsbildes; Rechts: Ausschnittsvergrößerung nach Durchlauf aller vier Richtungen

Ein Binärbild der schwarzen Objekte im Bild entsteht durch die Auswahl der Werte mit dem Maximalwert vier. Um robust auf entstandene Fehler zu reagieren, werden die Objekte noch geschlossen, also erodiert und anschließend dilatiert.



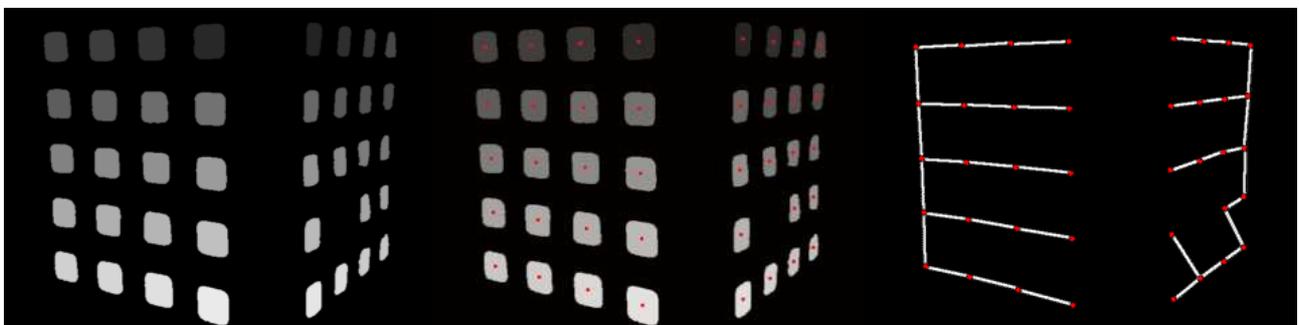
fertige Segmentierung des ersten Beispiels

3.3.2 Zuordnung

Alle Objekte im Bild werden nummeriert und anschließend deren Mittelpunkte bestimmt, indem die Mittelwerte der Koordinaten berechnet werden. So ist die Position subpixelgenau bekannt.

Von den erkannten schwarzen Punkten sind nun ihre Bildkoordinaten bekannt, sie müssen nun aber noch den dreidimensionalen Weltkoordinaten zugeordnet werden. Da der Fotoapparat je nach Objekt sinnvollerweise senkrecht oder waagrecht ausgerichtet wird, wurde ein drehungsinvariantes Verfahren benutzt was durch die spezielle Anordnung der Kalibrierungspunkte möglich wird.

Die Punkte werden zunächst in zwei Gruppen für die beiden Flächen des Kalibrierungsobjekts getrennt, indem ein minimal spannender Baum bis auf die letzte Kante konstruiert wird. Eine Gruppe enthält nun 20, die andere 19 Punkte, so lassen sie sich eindeutig zuordnen. Die Anordnung der Punkte erfolgte unter Berücksichtigung von guter Eignung sowohl für horizontale als auch für vertikale Aufnahmen, so daß im Bildraum Breite und Höhe der Fläche der Punkte etwa gleich groß sind.



Label, Mittelpunkte, minimal spannender Baum ohne letzte Kante

Nun werden die Eckpunkte innerhalb jeder Gruppe gesucht, indem für jeden Punkt ein Diagramm erstellt wird, in dem die Winkel zu allen anderen Punkten der Gruppe eingetragen sind. Eine Lücke von mehr als 190° wird als Kriterium für einen Eckpunkt verwendet. Die Eckpunkte werden in Bezug auf deren Mittelpunkt gegen den Uhrzeigersinn sortiert. Nun wird in der Gruppe mit den 19 Punkten der Eckpunkt bestimmt, der am nächsten an der Lücke liegt. Aus der anderen Gruppe kann nun der nächste zum eben gefundenen Punkt bestimmt werden. Die restlichen Eckpunkte können nun leicht eindeutig zugeordnet werden, die anderen Punkte werden durch Zuordnung mit bilinearer Interpolation gefunden.

Dieses Matching wird in anderen Projekten häufig auf eine andere Weise ausgeführt: mit ähnlichen Kalibrierungsobjekten wird eine Kantendetektion (z.B. mit Canny) und anschließend eine Hough-Transformation ausgeführt. Die Schnittpunkte der rekonstruierten Linien werden nun verwendet. Diesen Ansatz habe ich nicht benutzt, da die billigen oder Weitwinkellinsen Geraden in gekrümmte Linien verzerren und eine zusätzliche lokale Anpassung erfordern würden, die mir recht aufwendig erscheint.

3.3.3 Berechnung der Parameter

Die Bestimmung der Kameraparameter erfolgt nach der Methode von Tsai[12] und benutzt eine modifizierte Implementierung von Reg Willson[1]. Die Eulerwinkel, die Translation sowie Bildmittelpunkt, Brennweite und der Linsenverzerrungsparameter k_1 (radiale Verzerrung erster Ordnung) werden bestimmt.

3.3.4 Transformationen

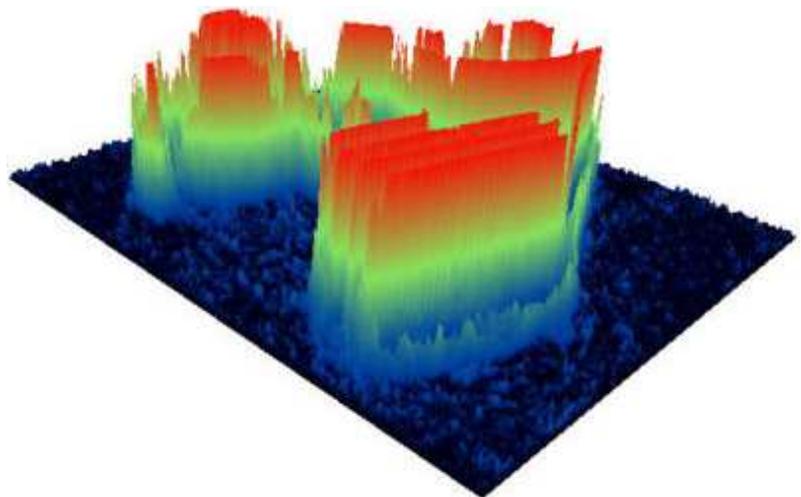
Die Bestimmung der Drehachse wird aus den zwei zueinander gedrehten Aufnahmen des Kalibrierungsobjekts berechnet. Da sich hierbei Fehler der Kalibrierung stark auswirken, steht eine zweite Methode zur Kalibrierung zur Verfügung, bei der nur eine Aufnahme erfolgt und die Kante des Kalibrierungsobjekts als Drehachse benutzt wird.

Weltkoordinaten können eindeutig einem Bildpunkt zugewiesen werden, Bildpunkte entsprechen einer Geraden im Weltkoordinatensystem.

Auch die Transformationen zwischen Kamera- und Weltkoordinaten und zwischen verzerrten und unverzerrten Bildkoordinaten können mit der Klasse angewandt werden.

3.4 Segmentierung

Die Fotos werden Median-gefiltert und in den HSL-Farbraum transformiert. Aus diesem Bild werden zwei neue erzeugt: ein Gradientenbild und ein Markierungsbild. Letzteres enthält drei Werte: Pixel, die ganz sicher zum Objekt oder zum Hintergrund zugeordnet werden können und unsichere Pixel. Diese Entscheidungen werden getroffen, indem in die Distanz zwischen den Farbwerten jedes Pixels zu den Farbwerten der Ecken ausgewertet werden.

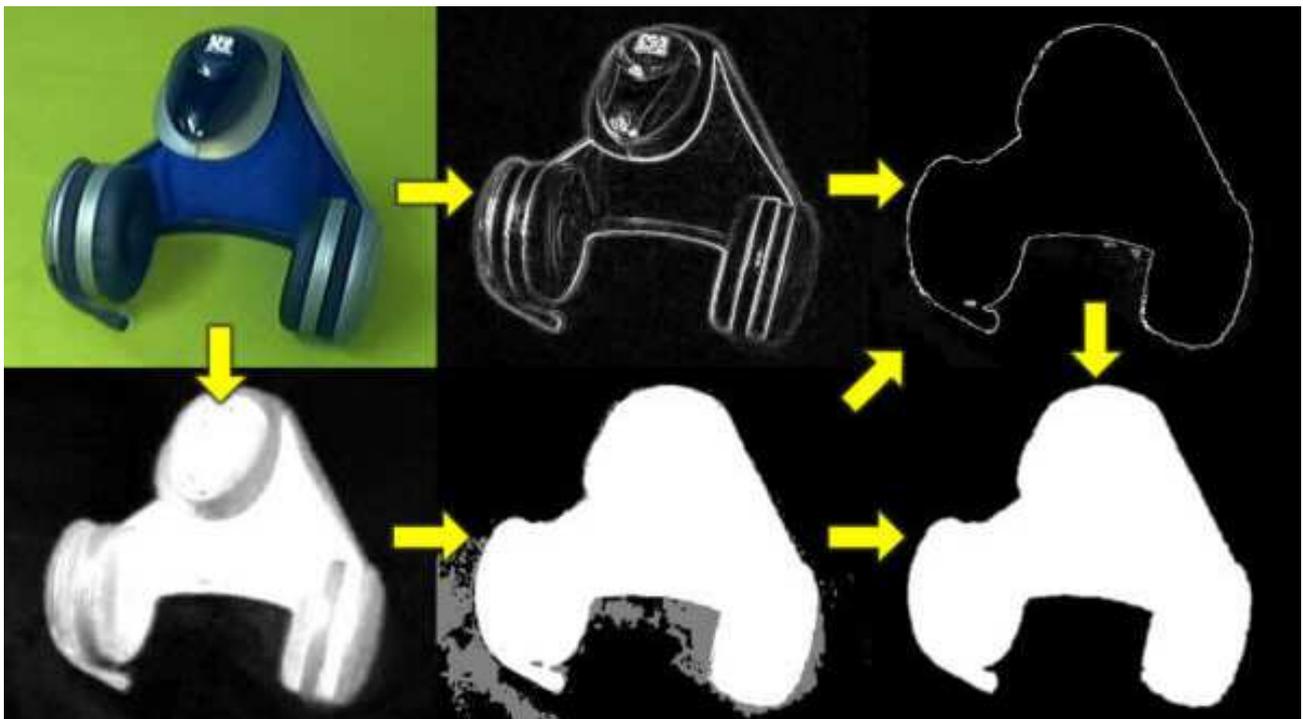


Dabei kann davon ausgegangen werden, daß in den Ecken die Hintergrundfarbe ist, jedoch wird ein kleiner Abstand zum Rand eingehalten, da einige der billigen Webcams ganz am Rand einen Streifen mit falschen Farben aufweisen.

Nun wird eine markerkontrollierte Segmentierung mit einer Wasserscheidentransformation mittels einer Überflutungssimulation ausgeführt. Als Markierungsfunktion werden die eindeutig zugeordneten Pixel benutzt.

Für die schnellere Transformation wird eine Minimaauferlegungstechnik (siehe [10]) benutzt, um lokale Minima auszuschließen. Dies geschieht mit Hilfe einer morphologischen Rekonstruktion (bzw. algebraischer Schließung) mittels wiederholten geodätischen Erosionen bis zur Identität mit einem 3x3 Quadrat als strukturierendem Element.

Es hat sich gezeigt, daß eine schlechte Markierungsfunktion insbesondere bei unscharfen Kanten zu unbefriedigenden Ergebnissen führt.



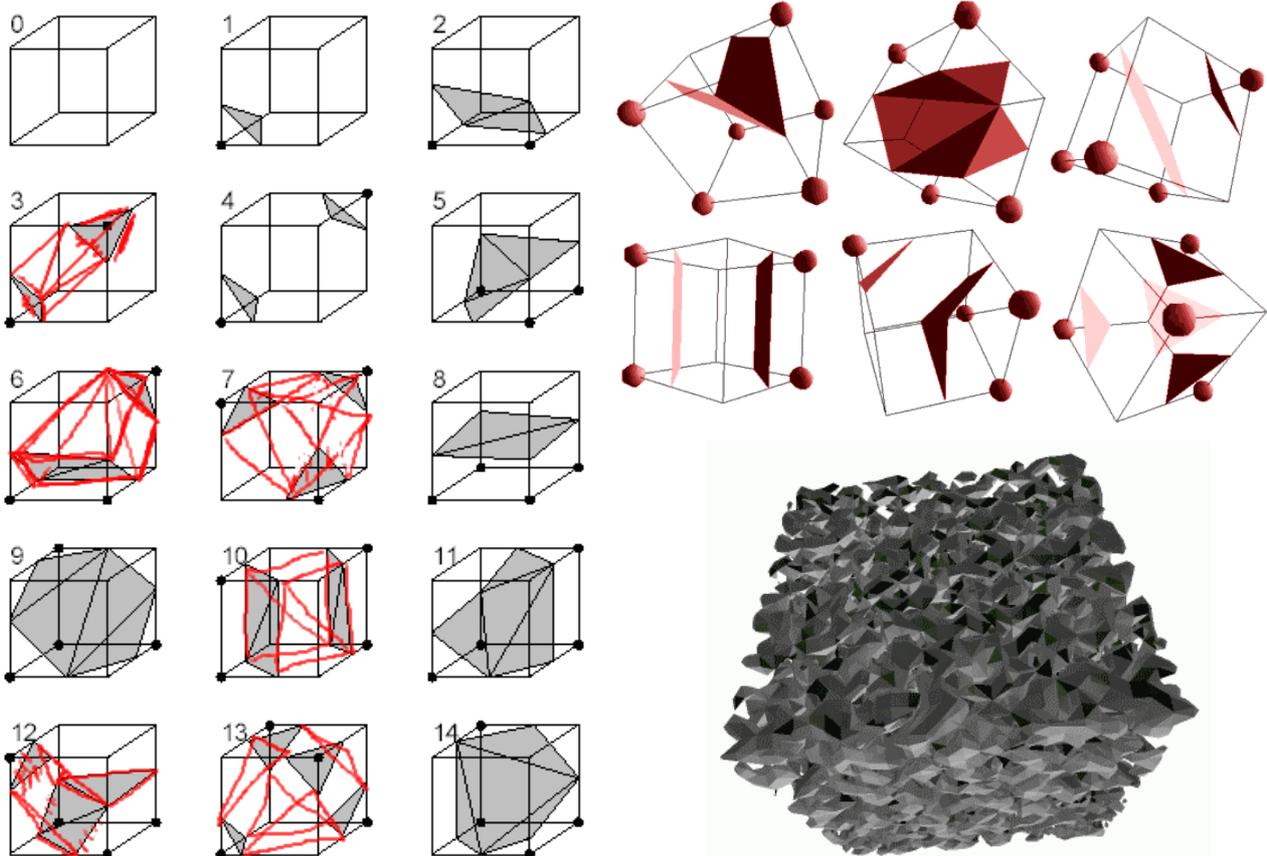
Transformationsschritte vom Original links oben zum Ergebnis rechts unten

Möglicherweise hilft der Einsatz von Snakes beim zuletzt genannten Problem.

3.5 Triangulierung der Volumendaten (Marching Cubes)

Um ein mannigfaltiges Objekt zu erzeugen, wurde der ursprüngliche Algorithmus [6] modifiziert, so daß bei nicht eindeutigen Kombinationen immer die negativen Ecken verbunden werden. Auf eine Betrachtung der Nachbarwürfel oder das Einfügen weiterer Punkte kann so verzichtet werden. Die dadurch in Einzelfällen entstehende falsche Topologie kann vernachlässigt werden.

Bei der Verarbeitung werden jeweils die Würfel zwischen zwei Schichten erzeugt, damit sich nicht die gesamten Volumendaten gleichzeitig im Speicher befinden müssen. Dreiecke bestehen aus drei Indizes für die Koordinaten. Somit kann der gleiche Punkt von allen entsprechenden Dreiecken benutzt werden.

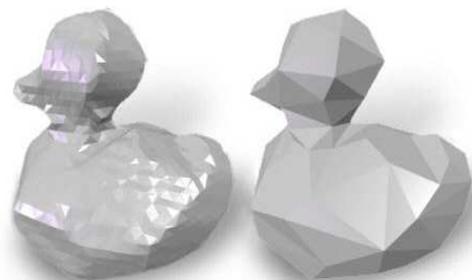


Links: Modifikationen für die Würfel mit invertierter Konfiguration; Rechts oben: Rendering dieser Konfigurationen zur Verifikation; Rechts unten: Verifikation der Mannigfaltigkeit mit Zufallswerten.

Der Algorithmus wird innerhalb eines Quaders auf einem regelmäßigen Gitter ausgeführt. Der Wert eines Gitterpunktes wird berechnet, in dem die Projektion für jedes Maskierungsbild berechnet wird. Der minimale Grauwert aller projizierten Punkte wird gewählt. Um Aliasing-Artefakte zu vermeiden, wird dabei für jede Projektion auch ein Einflußradius berechnet und mit einer zuvor berechneten Auflösungspyramide der Wert entsprechend trilinear interpoliert. Dennoch ist die Benutzung stark dezimierter Bilder problematisch, da feine Strukturen im Original zu großen grauen Bereichen nahe der Entscheidungsschwelle von Objekt und Hintergrund liegen. Weitere Filterungen wie ein Opening of Closing auf dem Maskierungsbild vor der Dezimation oder auf den 3D-Gitterdaten sollten angewandt werden, um diese Artefakte zu minimieren.

3.6 Reduktion der Polygonanzahl (Progressive Meshes)

Die Reduktion ist z.Z. noch nicht in das Programm eingebunden, funktioniert jedoch schon, indem das 3D-Modell in eine Datei exportiert und mit einem entsprechenden Programm bearbeitet wird.



Reduktion der Dreiecksanzahl auf etwa 2%

3.7 Texturierung

Die Textur wird als Farbwert für jeden Punkt des Modells berechnet. Auch hier wird der Punkt in jedes Foto projiziert und der Farbwert für jedes Foto mit einer Auflösungspyramide bestimmt. Der resultierende Farbwert wird durch eine nichtlineare Gewichtung des Skalarprodukts zwischen Normale und der jewei-

ligen Kamerablickrichtung bestimmt, damit die Textur jeweils von den Fotos am stärksten bestimmt wird, die sie am besten abbildet.

Die Dezimation erfolgt z.Z. inklusive der Hintergrundfarbe, so daß es bei sehr niedriger Auflösung zu Fehlern kommt. Da vor Allem aber Randpixel betroffen sind, die meist nur schwach gewichtet sind, kann dieser Effekt oft vernachlässigt werden.

Es kommt zu Texturfehlern bei z.B. Rillen oder Löchern, da kein Sichtbarkeitstest bei der Gewichtung durchgeführt wird. Korrekterweise dürfte nur bei direkter Sichtverbindung die Farbe mit eingerechnet werden. Das Bild zeigt diesen Effekt.

Ebenfalls problematisch ist die Reduktion der Anzahl der Dreiecke, denn viele Objekte benötigen eine deutlich höhere Auflösung für die Textur als für die Geometrie. Projizierte Texturen können hier weiter helfen, doch die direkte Benutzung der Originalbilder ist keine gute Lösung, da dies viele sein können.



Die Kerze „färbt ab“: die Figur ist fälschlicherweise teilweise rot, die Kerze teilweise weiß

3.8 Zusammenführen von Projekten

Zwei Projekte können kombiniert werden. So kann auch die Textur von allen Seiten erstellt werden, in dem das Objekt bei einem Scandurchgang z.B. auf die Seite gelegt bzw. auf den Kopf gestellt wird. Translation und Rotation müssen dabei aber manuell angepaßt werden. Dafür werden die 3D-Modelle beider Ausgangsprojekte erzeugt. Das neue Objekt kann nun mit der Maus verschoben und gedreht werden.

Es ist sogar möglich, vollkommen unkalibrierte Fotos eines Objekts so manuell in die richtigen Winkel zu drehen. Allerdings ist dies sehr schwierig, da Kameraparameter geschätzt werden müssen, die Objekte normalerweise nicht aus der gleichen Entfernung aufgenommen werden und es auch das manuelle Zusammenführen schwierig abzuschätzen ist.

3.9 Geschwindigkeit und Speicherbedarf

Zeitkritisch sind einige Filter bei der Segmentierung sowie das Erzeugen der Volumendaten. Letztere könnten durch einen hierarchischen Ansatz z.B. mit Octrees am stärksten profitieren, da z.Z. noch die zeitlich aufwändige Projektion für jeden Gitterpunkt mindestens einmal berechnet werden muß.

Der Aufwand für die Segmentierung steigt etwa linear mit der Bildgröße (Pixelanzahl).

Der Aufwand für das Erzeugen der Volumendaten wächst etwa linear mit der Anzahl der Gitterpunkte und sublinear mit der Anzahl der benutzten Fotos.

Der Speicherbedarf ist bei der Erzeugung der Volumendaten und bei der Texturierung besonders hoch, da hier jeweils die Bilder inklusive Auflösungspyramiden sowie das Modell selbst im Speicher liegen. Die Volumendaten werden scheibenweise verarbeitet, so daß nur immer zwei Scheiben im Speicher liegen und somit keinen großen Speicherbedarf haben.

Die ausführbare Datei ist z.Z. kleiner als 400 KByte (gepackt < 200 KByte) und benötigt keine Installationsroutine. Somit können Besucher der Website das Programm problemlos testen.

Um den Speicherbedarf für die Fotos zu reduzieren, ist die Unterstützung des JPEG-Formats sinnvoll, jedoch noch nicht implementiert.

3.10 GUI / Frontend

Da praktisch für alle Webcams bzw. digitalen Fotoapparate Treiber für Windows existieren, wurde dieses OS gewählt und auch dafür ein GUI entwickelt.

3.11 Hilfedateien

Eine HTML-Hilfedatei erklärt die Benutzung und Funktionsweise des Programms.

4 Ergebnisse

4.1 Genauigkeit und Auflösung

Die Genauigkeit hängt von vielen Faktoren ab: die Auflösung der benutzten Bilder, die Entfernung von der Drehachse, die Qualität der Segmentierung, die Genauigkeit des Winkels des Drehtellers und das Zusammenpassen der Größe von Kalibrierungsobjekt und zu scannendem Objekt spielen eine Rolle.

Die limitierenden Faktoren sind die Kalibrierung und die Segmentierung. Bei einem Scan wurden beispielsweise Brennweite und Kameraentfernung durch die Kalibrierung sehr schlecht geschätzt, so daß das Objekt zwar die richtige Form aber fast die doppelte Größe des Originals besitzt. Die Angabe von Genauigkeiten ist also schwierig.

4.1.1 Drehtellermechanik

Eine kurze Meßreihe der Anzahl der Schritte für eine Umdrehung verdeutlicht die Genauigkeit der Mechanik: 1526, 1523, 1522, 1523, 1522, 1523. Die maximale Abweichung < 1° ist meist zu vernachlässigen, wird jedoch mit zunehmendem Abstand zur Drehachse und abnehmender Objektgröße immer kritischer.

4.1.2 Kamerakalibrierung

Die auf Subpixel genaue Bestimmung der Bildkoordinaten der Kalibrierungspunkte sowie deren Anzahl scheinen eine gute Genauigkeit sicherzustellen. Die relativ großen Fehler bei der Berechnung der Drehachse zeigen u.a. aber, daß auch hier noch Verbesserungen nötig sind.

Auch neue Kalibrierungsobjekte sollten benutzt werden, bei denen die Punkte in Kamerarichtung weiter entfernt sind (gut für Schätzung Perspektive) oder auch Aufnahmen von oben möglich machen (wie z.B. eine Pyramide).

Hier ein Beispiel für Unterschiede von geschätzten Kameraparametern bei drei Kalibrierungen aus jeweils einem anderen Blickwinkel aber gleicher Kamera und gleichem Kalibrierungsobjekt (die Werte müßten also gleich sein):

	Aufnahme 1	Aufnahme 2	Aufnahme 3
Brennweite	8.51 mm	8.69 mm	8.45 mm
k1 (Linsenkrümmung)	0.00104 1/mm ²	0.00200 1/mm ²	0.00177 1/mm ²

4.1.3 Segmentierung

Die Segmentierung hat sicherlich den größten Einfluß auf die Artefaktbildung. Bei z.B. einem kleinen Fehler in den Kameraparametern verändert sich evtl. die Größe des Objekts, doch die Form bleibt erhalten. Ein Segmentierungsfehler an einem schwachen Gradienten jedoch führt sofort zu deutlich sichtbaren Rillen am Objekt.

Die automatische Segmentierung bietet noch Möglichkeiten zur Erhöhung der Genauigkeit, da im Moment noch keine gute Markierungsfunktion benutzt und nur eine binäre Segmentierung durchgeführt wird.

4.2 Probleme, Artefakte und manuelle Nachbearbeitung

Prinzipbedingte Artefakte sind nicht zum Objekt gehörende Pyramiden oder Füllungen von Objekten mit Einbuchtungen. Wird beispielsweise eine Tasse von schräg oben gescannt, so wird nicht nur der Innenraum vollständig gefüllt, sondern es entsteht oben auf noch eine Pyramide. Diese Pyramiden können meist durch den richtigen Blickwinkel bzw. die Kombination mehrerer Blickwinkel vermieden werden.

Problematisch Artefakte sind die bereits erwähnten Rillen, die schon durch eine fehlerhafte Segmentierung eine manuelle Nachbearbeitung notwendig machen können.

Bei runden Objekten und wenigen Aufnahmen werden Kanten (an den Schnittstellen der Silhouettenpyramiden) besonders sichtbar.

Die Möglichkeit der manuellen Nachbearbeitung bietet weitere Möglichkeiten, vor allem, um vor dem Erstellen der Volumendaten störende Objekte zu entfernen. Dies können Fixierungen sein, die das Objekt während des Scannens stützen. Die manuelle Segmentierung in Verbindung mit dem Zusammenführen mehrerer Projekte und der Schätzung von Kameraparametern ermöglicht auch die grobe Rekonstruktion von Objekten mit Hilfe von normalen Fotos.

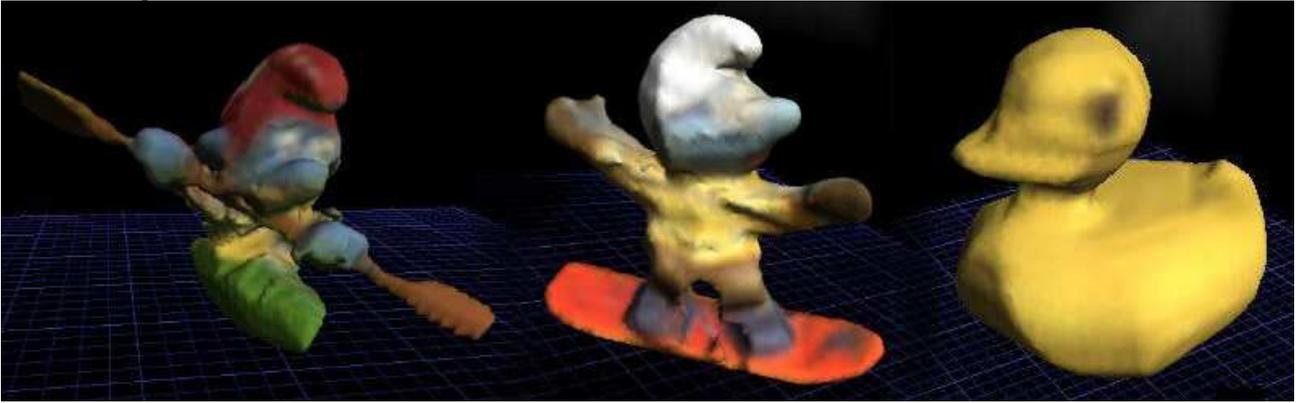
4.3 Arbeitsaufwand

Anstatt der Parallelportansteuerung wäre es wohl doch einfacher gewesen, einen Mikrocontroller einzusetzen und die serielle Schnittstelle zu benutzen.

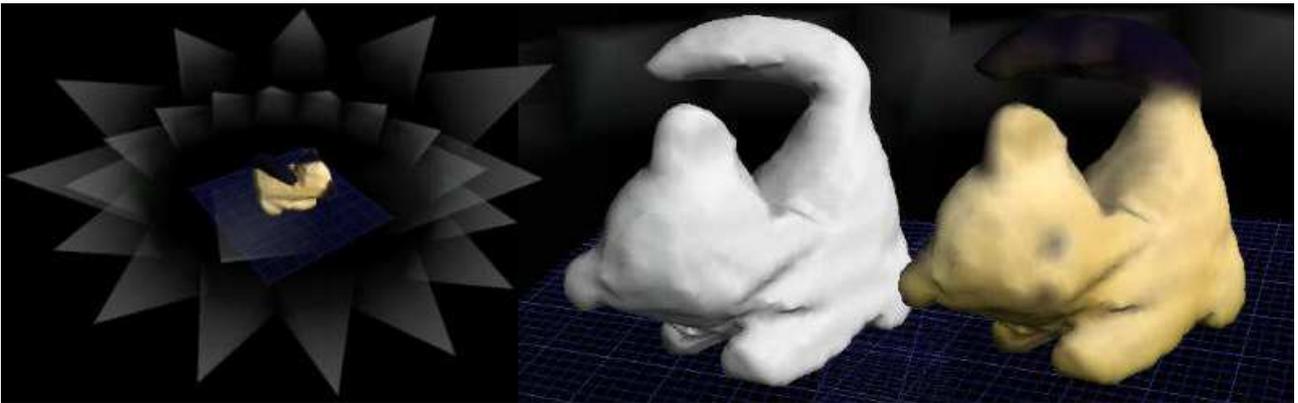
Die Entwicklung einer eigenen Klasse für den TWAIN-Zugriff war aufwendig. Zwei bereits existierende Klassen konnten nicht benutzt werden, da eine nicht funktionierte und die andere vom Leistungsumfang nicht ausreichte.

Arbeit	Stunden
Hardware	100
Kamerazugriff	60
Kalibrierung	75
Segmentierung	75
Marching Cubes (zusätzlich)	25
Bericht	55
diverses	250
gesamt	640

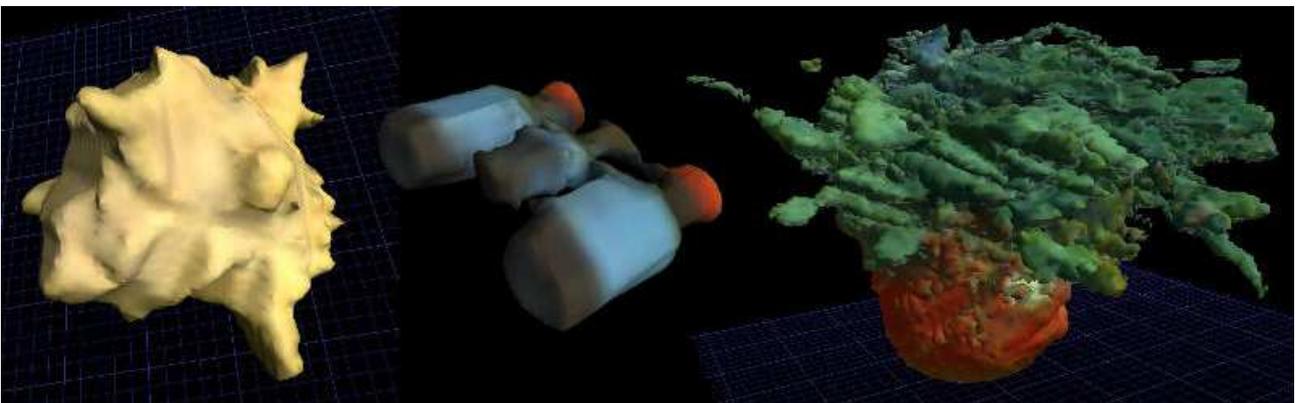
4.4 Beispiele



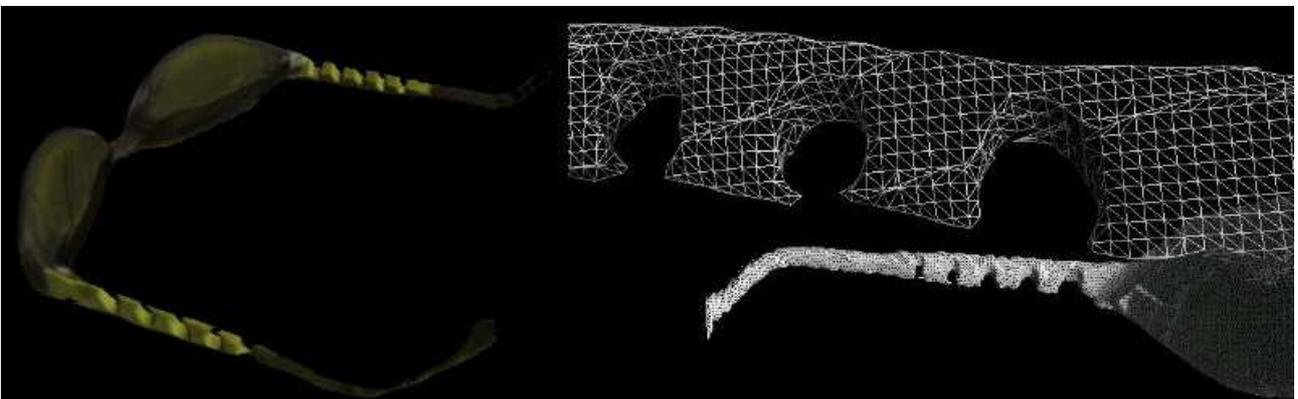
Kajak-Schlumpf, Snowboard-Schlumpf, Badeente



Stoffkatze: Kamerablickwinkel, ohne und mit Textur



Muschel, Spielzeugfernglas, Pflanze



Sonnenbrille; rechts: Fehler in der Struktur durch ungenaue Kalibrierung sind deutlich sichtbar



manuelle Rekonstruktionen mit jeweils 5 bzw. 6 Fotos



5 Vergleich und mögliche Verbesserungen

5.1 Wichtigste mögliche Verbesserungen

- mehrere unterschiedlich farbige, auswechselbare Hintergrundstoffe
- eine bessere Beleuchtung
- png- und jpg-Unterstützung
- bessere Segmentierung
- Kalibrierungsobjekt z.B. in Pyramidenform, um Aufnahmen von oben zu ermöglichen
- Filterung der 3D-Daten vor der Triangulierung
- automatisches Merging
- Einbau der Polygonreduktion (progressive meshes) in die Anwendung
- Statuszeile mit Infos über gerade verfügbare Tasten- / Mausfunktionen
- schnellere Triangulierung durch hierarchische Verarbeitung: nur Bereiche mit Vorzeichenwechsel müssen bearbeitet werden

5.2 Kommerzielle silhouettenbasierte 3D-Scanner

 <p>3D ScanBook und 3D ScanStation von Dimension 3D-Systems</p>	 <p>ShadowBox von CamSys</p>	 <p>TDV Series Color 3D Scanner von Shenzhen</p>
<p>Preis: ab ca. 4500 EUR Genauigkeit: bis 0.5% der Objektgröße Besonderheiten: • Merging und Polygonreduktion • Export in sehr kleine Dateien (f. Internet) • weitere Zusatzsoftware wie Texturmalprogramm</p>	<p>Preis: ab ca. 5000 EUR Genauigkeit: 0.03% Besonderheiten: • Merging und Polygonreduktion</p>	<p>Preis: ? Genauigkeit: 0.03 – 0.05 (%?)</p>

5.3 Vor- und Nachteile des Verfahrens

Vorteile	Nachteile
<ul style="list-style-type: none">• kein extra Scan für Texturen (schon inklusive)• schneller als Laser-Scanner• mit vielen Kameras können auch bewegte Objekte aufgenommen werden• benötigte Hardware sehr günstig• beliebiges Objektmaterial	<ul style="list-style-type: none">• langsamer als manch andere Verfahren• Einschränkungen bzgl. Objekten: keine Rekonstruktion von Einbuchtungen• der Einsatz eines Drehteller anstatt einer bewegte Kamera schränkt die möglichen Objekte zur automatisierten Aufnahme stark ein

6 Referenzen

6.1 Externer Source Code

- [1] "Camera Calibration using Tsai's Method": Reg Willson
- [2] "Port Talk I/O Port Driver": Beyond Logic, <http://www.beyondlogic.org>
- [3] NeHe OpenGL Base Code, <http://nehe.gamedev.net>
- [4] Progressive Meshes aus der Meshlib: Sven Havemann, TU-Braunschweig

6.2 Bibliographie

- [5] "Klappe auf (3D-Volumenschnitt-Scanner von Dimension 3D-Systems)": Dr. Jürgen Rink, c't 3/2000
- [6] "Marching Cubes: A High Resolution 3D Surface Construction Algorithm": William E. Lorensen, Harvey E. Cline, Computer Graphics, Volume 21, Number 4, July 1987
- [7] "An Evaluation of Implicit Surface Tilers": Paul Ning and Jules Bloomenthal, IEEE Computer Graphics and Applications, Vol. 13, Number 6, November 1993
- [8] "Marching Cubes 33: Construction of Topologically Correct Isosurfaces": Evgeni V. Chernyaev, Institute for High Energy Physics
- [9] "TWIN Specification, Version 1.9": TWIN Working roup Committee, <http://www.twain.org>
- [10] "Morphologische Bildverarbeitung": Pierre Soille, Springer Verlag
- [11] "The Virtual Reality Modeling Language": The VRML Consortium Incorporated, 1997
- [12] "A versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-Shelf TV Cameras and Lenses": Roger Y. Tsai, IEEE Journal of Robotics and Automation, Vol. RA-3, No. 4, August 1987, pages 323-344
- [13] "A Flexible New Technique for Camera Calibration": Zhengyou Zhang, Microsoft Research
- [14] "Volumetric Scene Reconstruction From Multiple Views": Charles R. Dyer, University of Wisconsin, Foundations of Image Understanding (2001)
- [15] "Low-Cost Model Reconstruction from Image Sequences": Caleb Lyness, Otto-Carl Marte, Bryan Wong and Patrick Marais, Collaborative Visual Computing Laboratory, Department of Computer Science, University of Cape Town
- [16] "Introductory techniques for 3-D Computer Vision", Trucco E. and Verri A., Prentice-Hall, Inc. 1998