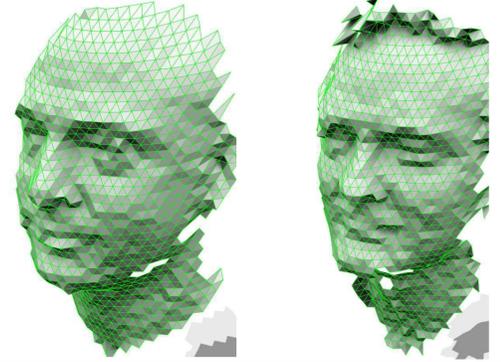


Bildbasierte 3-D-Rekonstruktion von Gesichtern mittels CUDA-gestützter Anpassung von Dreiecken

Masterarbeit, vorgelegt von Sebastian Moritz

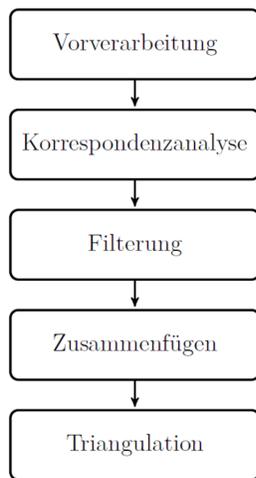


Aufgabenstellung

Die Masterarbeit ist in der Image-Processing-Abteilung des Fraunhofer Heinrich-Hertz-Institut entstanden und ist thematisch im Bereich des maschinellen Sehens und der Computergrafik angesiedelt.

Ziel der Arbeit ist die Konzeption und Umsetzung der Rekonstruktion eines dreidimensionalen Gesichtsmodells aus Bildern mit der CUDA-Technologie von NVIDIA.

Systemarchitektur



Schematische Darstellung der Fließbandverarbeitung

Die 3-D-Rekonstruktion wird nach dem Prinzip der Fließbandverarbeitung durchgeführt und besteht aus den folgenden fünf Schritten:

Vorverarbeitung

Die Vorverarbeitung dient dazu, die Eingangsbilder für die Korrespondenzanalyse vorzubereiten, mit dem Ziel, die Bestimmung der Korrespondenzen zu beschleunigen.

Korrespondenzanalyse

Die Korrespondenzanalyse dient dazu, korrespondierende Dreiecke in den Bildern eines Stereokamerasystems zu bestimmen. Für ein vordefiniertes Polygongitter, das mit dem linken Bild verbunden ist, wird ein Polygongitter von angepassten Dreiecken gesucht, das mit dem rechten Bild verbunden ist.

Filterung

Nach der Korrespondenzanalyse wird die Filterung durchgeführt, um fehlerhafte Korrespondenzen zu detektieren und zu entfernen.

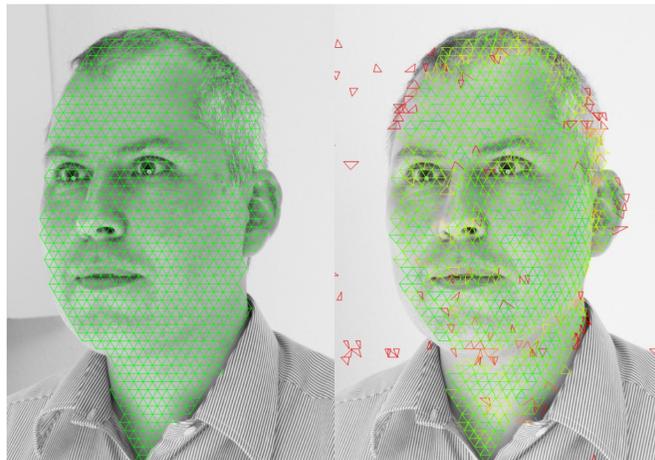
Zusammenfügen

Das Zusammenfügen betrifft nur die Dreiecke des rechten Polygongitters. Das Zusammenfügen ist notwendig, weil die Dreiecke nach der Korrespondenzanalyse noch nicht miteinander verbunden sind.

Triangulation

Aus den zweidimensionalen Polygongittern, die in den vorherigen Schritten erstellt wurden, wird in diesem Schritt ein 3-D-Modell erzeugt.

Korrespondenzanalyse



Beispiel für die Korrespondenzanalyse auf Grundlage von Dreiecken.

Novum der 3-D-Rekonstruktion bildet die Korrespondenzanalyse. Im Gegensatz zu vielen anderen Verfahren werden nicht Bildpunkte oder Bildmerkmale einander zugeordnet, sondern Dreiecke. Die Dreiecke bestimmen hierbei, welche Bildflächen als Korrespondenzen betrachtet werden. Durch Anpassen der Dreiecke an den Bildinhalt werden perspektivische Unterschiede kompensiert. Die perspektivischen Unterschiede in der Darstellung einer Szene entstehen aufgrund der räumlichen Trennung der Kameras.

Untersuchungsergebnis

Für die Korrespondenzanalyse wird ein Ähnlichkeitsmaß benötigt, mit dem ein numerischer Wert für die Ähnlichkeit von zwei Bildausschnitten berechnet werden kann. Hierfür wurden mehrere Ähnlichkeitsmaße untersucht und bewertet. Hierbei stellte sich heraus, dass das ZNCC (engl. Zero Mean Normalized Cross Correlation) Ähnlichkeitsmaß der beste Kompromiss zwischen Laufzeit und Zuverlässigkeit ist. Mit dem ZNCC können sowohl additive als auch multiplikative Helligkeitsunterschiede kompensiert werden.

Für die Filterung wurden ein nachbarschaftsbasierter und ein schwellwertbasierter Algorithmus konzipiert und untersucht. Die besten Ergebnisse wurden mit der Filterung durch benachbarte Dreiecke erzielt. Hierbei wird die Beziehung der Dreiecke zueinander berücksichtigt. Für jedes Dreieck eines Polygongitters wird der Abstand zu seinen unmittelbar benachbarten Dreiecken bestimmt. Berechnet wird der euklidische Abstand der Eckpunkte der Dreiecke. Eine Korrespondenz wird als fehlerhaft betrachtet, wenn ein Dreieck zu weit von seinen benachbarten Dreiecken entfernt ist, also eine bestimmte Schwelle überschreitet.

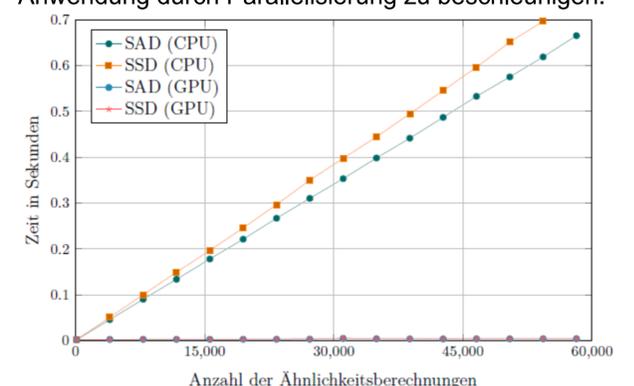
Ziel des Zusammenfügens ist es, die einzelnen Dreiecke zu einem Polygongitter zusammenzufügen. Hierfür wurden ein bildbasierter und ein medianbasierter Ansatz analysiert. Die Untersuchungen haben gezeigt, dass das Zusammenfügen mittels Median am schnellsten funktioniert. Beim Zusammenfügen der Eckpunkte durch den Median werden korrespondierende Eckpunkte von benachbarten Dreiecken bestimmt und der Median für die x und y Koordinaten ermittelt. Der Median wird verwendet, um zu verhindern, dass Ausreißer das Ergebnis zu stark beeinflussen.

Implementierung

Für das Testen der Algorithmen wurde eine Software entwickelt, die über eine Konfigurationsdatei bzw. Konsolenparametern initialisiert werden kann. Bei der Entwicklung stand die einfache Testbarkeit der Algorithmen im Vordergrund.

Alle Teile der Verarbeitungskette mit CUDA zu parallelisieren, ist aus mehreren Gründen nicht sinnvoll. Die Wartbarkeit verringert sich beispielsweise, je mehr Programmteile parallelisiert werden. Darüber hinaus ist der Entwurf von paralleler Software aufwendiger, als der von sequenzieller Software. Alle Teile einer Software zu parallelisieren, wäre also sehr ineffizient. Aus diesen Gründen ist es zielführender, nur die Teile zu parallelisieren, die eine Steigerung der Leistung bewirken.

Die Aufgabe ist also, zu bestimmen, welche Programmteile das höchste Potenzial haben, die Leistung des Programms zu steigern. Hierfür wurden die Laufzeiten eines sequenziell ablaufenden Prototyps untersucht. Hierbei wurde deutlich, dass die Korrespondenzanalyse das größte Potenzial hat, die Anwendung durch Parallelisierung zu beschleunigen.



Beschleunigung der Ähnlichkeitsberechnung durch CUDA

Zusammenfassung

Ziel der Materarbeit war es, ein Verfahren zur bildbasierten 3-D-Rekonstruktion von Gesichtern zu erstellen. Darüber hinaus befasst sich die Arbeit mit der Parallelisierung der 3-D-Rekonstruktion mittels CUDA. Neben den theoretischen Aspekten, die bei der Parallelisierung beachtet werden müssen, wurde die konkrete Umsetzung mit CUDA dargestellt.

Fazit

Die Ergebnisse der Masterarbeit zeigen, dass eine bildbasierte 3-D-Rekonstruktion von Gesichtern mit dem vorgestellten Verfahren durchgeführt werden kann.

Folgende Ziele wurden erreicht:

- Korrespondenzanalyse mittels Dreiecken, die robust, korrekt und zuverlässig ist,
- Beschleunigung der Anwendung durch Portierung auf die Grafikkarte.

Ausblick

Die Bestimmung von Korrespondenzen dauert selbst mit CUDA sehr lange. In der vorliegenden Arbeit wurde die Korrespondenzanalyse auf einer Grafikkarte ausgeführt. Es ist jedoch möglich, die Korrespondenzanalyse so aufzuteilen, dass sie auf mehreren Grafikkarten ausgeführt werden kann. Dadurch könnte eine weitere Beschleunigung des Verfahrens herbeigeführt werden.