

# SRA/ALR Prüfung

## Hochschule Mannheim

Michael Stapelberg, Felix Bruckner, Pascal Krause

Fakultät für Informatik  
Hochschule Mannheim

2012-07-19

- 1 Allgemeines
  - Unser Projekt
  - Architektur
  - Vorgehen

- 2 Hardware
  - Kinect
  - Grafikkarte
  - Handschuh

- 3 Algorithmen
  - Median Filter
  - Kalibrierung
  - Gloweffekt
  - RGB-Bild maskieren
  - Referenz-Farbe
  - Speed-Up

- 4 Ausblick

- 1 Allgemeines
  - Unser Projekt
  - Architektur
  - Vorgehen

- 2 Hardware
  - Kinect
  - Grafikkarte
  - Handschuh

- 3 Algorithmen
  - Median Filter
  - Kalibrierung
  - Gloweffekt
  - RGB-Bild maskieren
  - Referenz-Farbe
  - Speed-Up

- 4 Ausblick

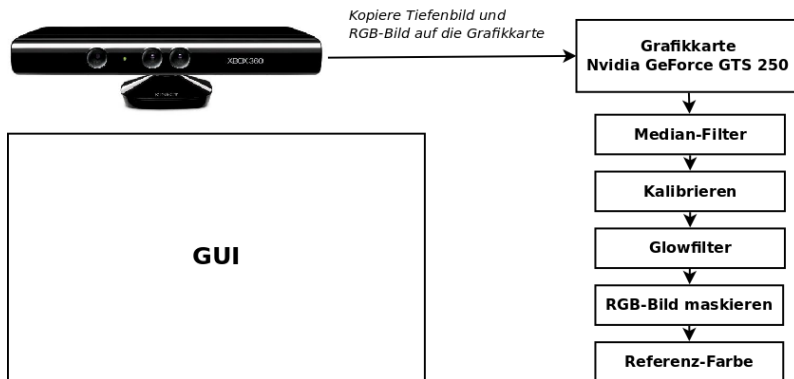
# Projekt

## Die Idee

Jede einigermaßen ebene Fläche, soll als Whiteboard dienen können.

- Die Schwierigkeiten
  - Auflösung des Tiefenbildes zu gering
  - Farberkennung auf **gesamten** Bild schlecht
- Der Lösungsansatz
  - Aufbereitung der Daten durch anwenden von Filtern

# Architektur



# Vorgehen

- Suchen eines open-source SDK
- Erforschen der Kinect
- Suchen nach Problemlösungen
- Validieren der Lösungen (CPU)
- Portieren der Lösungen auf GPU
  - Programmaufbau an CUDA anpassen
  - Algorithmen für CUDA optimieren

- 1 Allgemeines
  - Unser Projekt
  - Architektur
  - Vorgehen

- 2 Hardware
  - Kinect
  - Grafikkarte
  - Handschuh

- 3 Algorithmen
  - Median Filter
  - Kalibrierung
  - Gloweffekt
  - RGB-Bild maskieren
  - Referenz-Farbe
  - Speed-Up

- 4 Ausblick

# Kinect



- Sensors
  - 640x480 30Hz RGB Depth
  - 640x480 30Hz Depth
- Genauigkeit
  - Genauigkeit ab 50cm ca. 1,5mm
  - Genauigkeit ab 5m ca. 5cm



# Grafikkarte



- Nvidia GeForce GTS 250
- 16 Prozessoren / 8 Cores
- 1024 MB Device-Memory

# Handschuh



- 100% Baumwolle
- Hoher Tragekomfort
- Farbe: Orange

- 1 Allgemeines
  - Unser Projekt
  - Architektur
  - Vorgehen

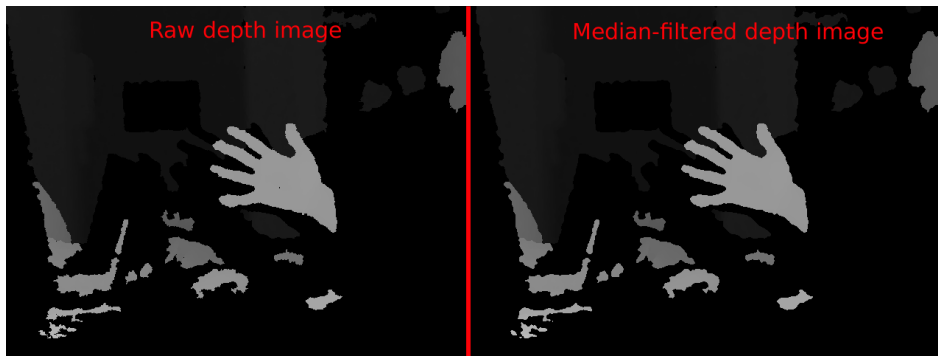
- 2 Hardware
  - Kinect
  - Grafikkarte
  - Handschuh

- 3 Algorithmen
  - Median Filter
  - Kalibrierung
  - Gloweffekt
  - RGB-Bild maskieren
  - Referenz-Farbe
  - Speed-Up

- 4 Ausblick

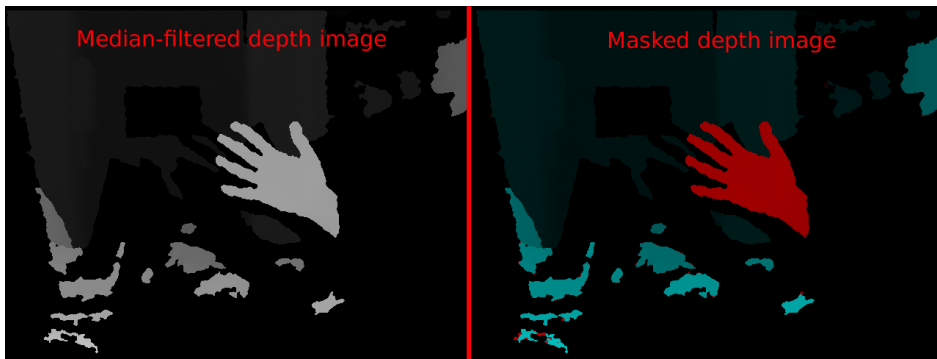
# Median Filter

- Arbeitet auf dem Tiefenbild
- Filtert das Rauschen aus dem Tiefenbild heraus



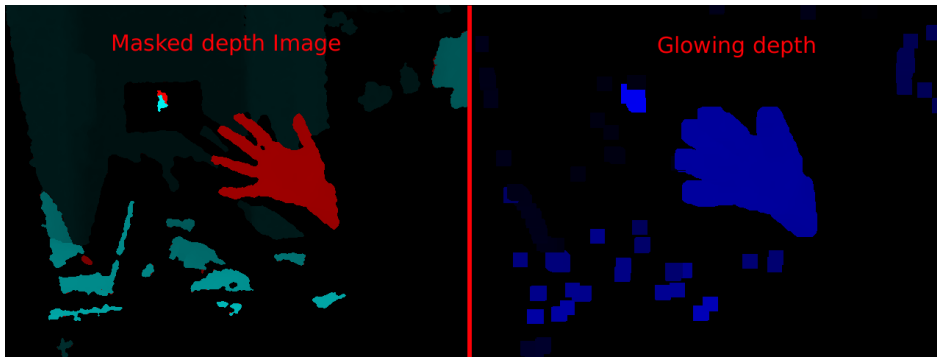
# Kalibrierung

- Es kann auf neuen Hintergrund kalibriert werden
- Arbeitet auf dem Tiefenbild
- Filtert alle sich nicht bewegenden Punkte aus dem Tiefenbild.



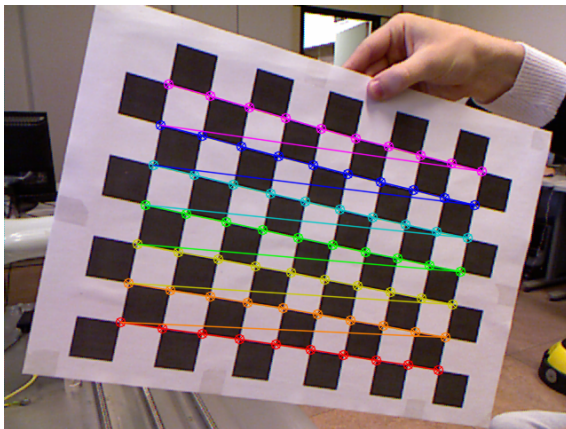
# Gloweffekt

- Arbeitet auf dem Tiefenbild
- Expandiert alle anzuzeigenden Punkte im Tiefenbild, anhand eines einstellbaren Radius



# RGB-Bild maskieren

- Arbeitet auf dem RGB-Bild
- Rechnet das Tiefenbild auf das RGB-Bild um, und maskiert alle relevanten Pixel



# Referenz-Farbe

- Arbeitet auf dem RGB-Bild
- Zeigt nur noch die Pixel auf dem RGB-Bild an, welche die referenzierte Farbe haben





# Speed-Up

$$S(p) = \frac{\text{Ausführungszeit SingleCore}}{\text{Ausführungszeit MultiCore}}$$

## Rechenzeit Tabelle

Recheneinheit	Median Filter	RGB-Bild maskieren
CPU	3 Millisekunden	3 Millisekunden
GPU	2 Mikrosekunden	1 Mikrosekunden

Speed-Up Median Filter: 5%

Speed-Up RGB-Bild-Maskierung: over9000%

- 1 Allgemeines
  - Unser Projekt
  - Architektur
  - Vorgehen
- 2 Hardware
  - Kinect
  - Grafikkarte
  - Handschuh
- 3 Algorithmen
  - Median Filter
  - Kalibrierung
  - Gloweffekt
  - RGB-Bild maskieren
  - Referenz-Farbe
  - Speed-Up
- 4 Ausblick

# Ausblick

- Bewegung interpolieren
- Extremitäten statt Pixel
- Visuell bedienbare GUI
  - Buttons
  - Gesten
- Ausführlichere Dokumentation
- Plattformabhängigkeit minimieren

Vielen Dank  
für Ihre Aufmerksamkeit