GDV OpenGL-Praktikum

Bastian Kreuzer (734877), Adrian Müller (734922)

# Kurzbeschreibung Animation Skiroboter

In einer schneebedeckten Gebirgslandschaft fahren 12 humanoide Roboter einen Berg hinunter. Der in Ausgangssicht befindliche vorderste Roboter auf der Piste lässt sich mit den Tasten J und L nach rechts bzw. links bewegen. Die übrig restlichen Roboter fahren die Piste in Kurven hinab.

Ein Roboter besteht aus Kopf, Hals, Torso, Ober- und Unterarm, Ober- und Unterschenkel, Füßen und Skiern. Die Roboter sind in ihrer farblichen Gestaltung einerseits nach Crashdummys und andererseits typischen Förderbandfabrikrobotern nachempfunden: Sie sind überwiegend in Orange, Grau & Schwarz gehalten und mit Symbolen versehen (Warnsymbole und Positionsmarker).

Beim Herabfahren sind alle Roboter leicht nach vorne (um die x-Achse) geneigt. Bei einer Rechts- oder Linkskurve ändert der Roboter seine Orientierung (y-Achse) und neigt sich zur Seite (z-Achse), entsprechend wird das jeweils innere Bein gefaltet, sodass die Skier auf der Piste bleiben. Die Arme sind leicht nach hinten geneigt und werden ebenfalls abhängig von der Seitenneigung bewegt.

Die Roboter bewegen sich nur rechts/links, um die Fahrt den Berg hinunter darzustellen wird das Terrain, das die Piste darstellt, kontinuierlich nach oben bewegt. Um das Terrain endlos erscheinen zu lassen, besteht die Piste aus 5 Teilterrains, wobei der hinterste Teil wieder von unten hoch geschoben wird, sobald er entsprechend weit von den Robotern entfernt ist. Um diesen Sprung zu kaschieren, wird Terrain nur bis zu einer gewissen Entfernung dargestellt und dann langsam ausgeblendet.

# Übersicht

* Trennung in 2 Codeebenen: Aufgabenstellung & „Treiber“, der von der OpenGL-Schnittstelle abstrahiert
* Übergeordneter Programmablauf:   
  

# Features

## Texturen

Jede Geometrie wird mit einer Textur dargestellt.

### Mipmaps

Mipmaps sind erforderlich, um zu verhindern, dass Texturen bei Animationen rauschen, wenn sie kleiner dargestellt werden, als ihre Originalauflösung vorgibt. Daher speichert man neben der eigentlichen Textur auch verkleinerte Versionen des gleichen Bildes ab – üblicherweise jeweils halb so groß, wie die vorhergegangene Stufe, bis zur Auflösung 1x1 Pixel herunter. Beim Rendern sucht sich die Grafikkarte die geeignetste Auflösung aus.

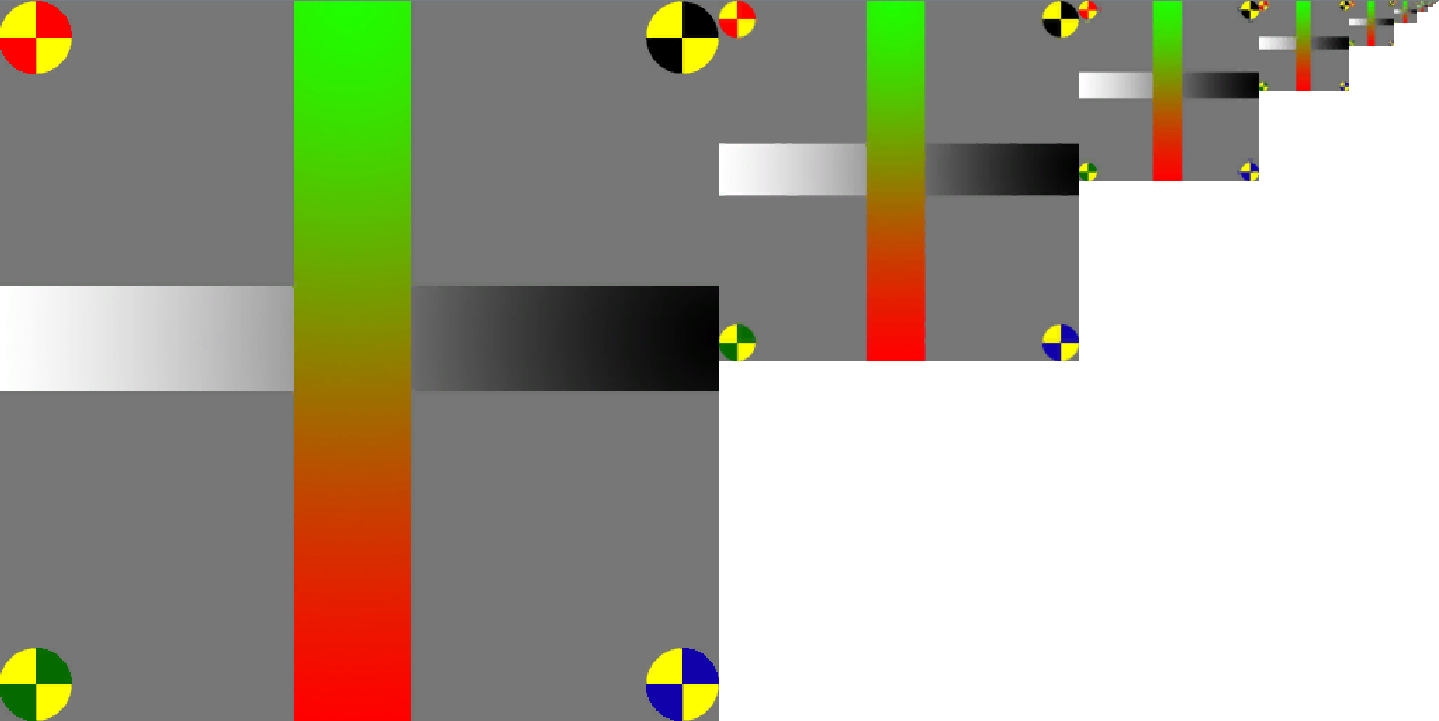


Abbildung 1: Die verwendete Testtextur und die dazu erzeugten MIPMAPs

### DirectDrawSurface (DDS)-Dateiformat, S3TC-Komprimierung

Das DDS-Dateiformat ist ein von Microsoft für seine Schnittstelle DirectX entworfenes Dateiformat zur Speicherung von Texturen, die für 3D-Rendering verwendet werden. Es ist quelloffen dokumentiert (msdn.microsoft.com, „Programming Guide for DDS“) und kann daher auch leicht mit OpenGL verwendet werden. Im Vergleich zu herkömmlichen Formaten, wie PNG oder JPEG, unterstützt es u.a. das Speichern von MIPMAPs, definiert aber kein eigenes Komprimierungsverfahren.

Zur Komprimierung verwenden wir daher die S3TC-Komprimierungen (auch bekannt als DXT1-5, wie Microsoft die Algorithmen in DDS nennt). Der Vorteil dieser Komprimierungsverfahren ist, dass Grafikkarten aller verbreiteten Hersteller das Format in Hardware ohne Zeitverlust dekomprimieren können. Neben geringeren Anforderungen an den Videospeicher und kürzeren Ladezeiten – da weniger Daten (i.d.R. via PCI-E) in den GRAM geladen werden müssen – macht es vor allem auch die eigene Implementierung des Dekomprimierens überflüssig, weil das Bild komprimiert an die Grafikkarte übertragen werden kann.

## Heightmap

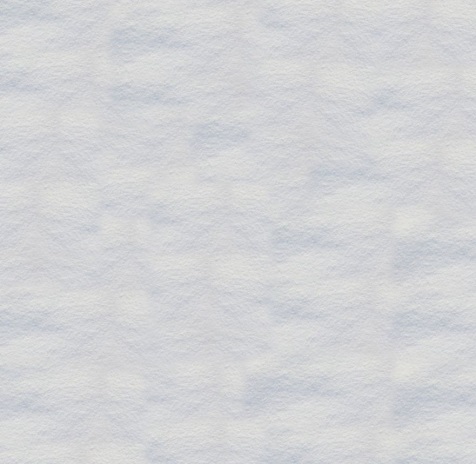
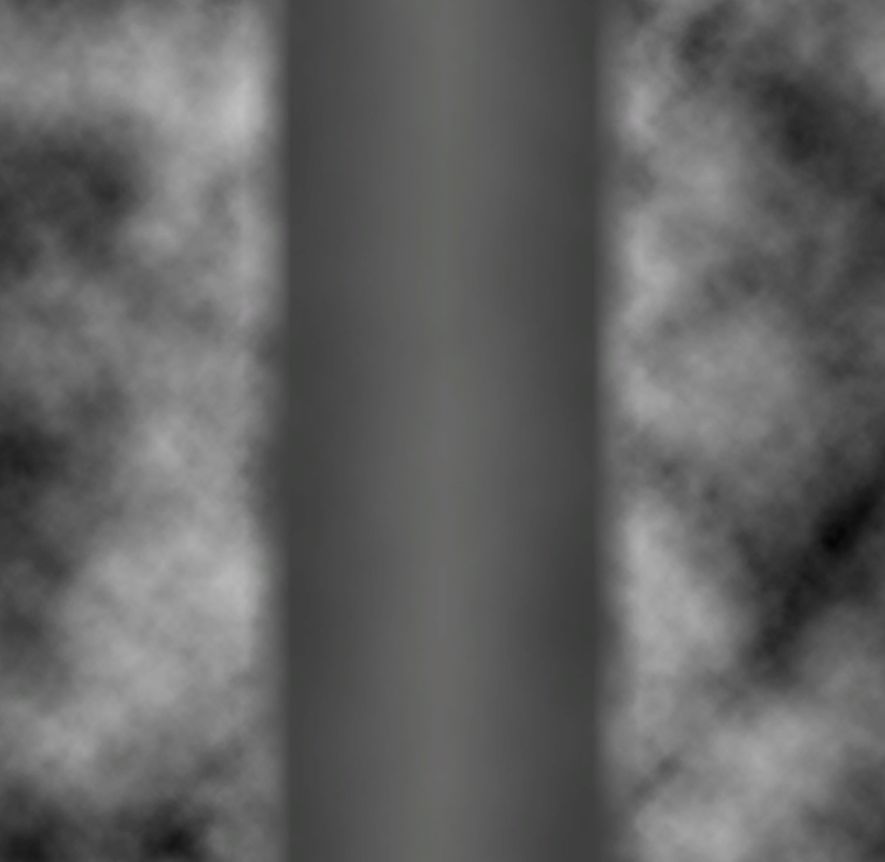


Abbildung 2: Heightmap Schneepiste Abbildung 3: Textur Schnee

Die Heightmap ist ein Bild, das ausschließlich Grauwerte enthält. Diese Grauwerte werden von einem entspechenden Shader als Höhe (y-Koordinate) eines Terrains interpretiert. Je heller der Grauwert, desto höher wird an dieser Stelle das Terrain. Die Schneepiste ist nahezu eben bzw. leicht rund und mittig in der Heightmap zu erkennen.

# Tastensteuerung

W; S Kamera vorwärts, rückwärts bewegen

A; D Kamera nach rechts, nach links bewegen

Shift; Strg Kamera auf/ab bewegen

Pfeiltasten Kamera drehen

Bild hoch Heranzoomen

Bild runter Herauszoomen

F2 Rastersicht

1 (halten) Transformationen der Roboter nicht aktualisieren

J; L Vordersten Roboter nach rechts bzw. nach links neigen