

UNIVERSITÄT HAMBURG

PROJEKTARBEIT INTERACTIVE VISUAL COMPUTING

Das wunderbare Leben des kleinen Tux

Ein POV-Ray-Animationsfilm

Inga Lemme Thomas Ort Melanie Remmels



22. Januar 2016

Inhaltsverzeichnis

1 Projektidee	3
1.1 Plot des Films	4
1.2 Liste der POV-Ray-Module	4
2 Statischer Aufbau der Figuren	6
2.1 Konstruktion der Umgebung	6
2.2 Aufbau des Pinguineis	6
2.3 Aufbau der Hauptfigur	6
2.3.1 Körper	6
2.3.2 Gliedmaßen	8
2.3.3 Accessoires	8
3 Aufbau der einzelnen Animationssequenzen	9
3.1 Anfangssequenz: Erstes Anzeichen von Leben	9
3.2 Sequenz 1: Schlüpfen des Tux	9
3.3 Sequenz 2: Babytux entdeckt seine beweglichen Gliedmaßen	10
3.4 Sequenz 3: Flugzeug	10
3.5 Sequenz 4: Bildergalerie	11
3.6 Abschlusssequenz: Lebensabend	11
4 Fertigstellung des gesamten Films	12
4.1 Zusammenfügen der einzelnen Sequenzen	12
4.2 Vertonung	12
4.3 Korrekturen	12
Literatur	13

1 Projektidee

Vorlage für die Hauptfigur des hier beschriebenen Films wurde das Maskottchen des freien Kernels *Linux*. Dieses stellt einen Pinguin dar und wird kurz *Tux* genannt. *Tux* wurde im Jahr 1996 von *Larry Ewing* mit der Bildbearbeitungssoftware *GIMP* entworfen und steht seitdem zur freien Verfügung für die Gemeinde. Er darf nach Belieben verwendet und verändert werden, solange auf Nachfrage sowohl Urheber¹ als auch das verwendete Programm genannt wurden. [1]

Die Idee, dass das Logo ausgerechnet einem Pinguin nachempfunden wurde, stammt von *Linus Torvalds*, dem Gründer von Linux. Laut *Jeff Ayers*, einem Linuxentwickler, besitzt *Torvalds* eine Affinität für „flugunfähige, fette Wasservögel“, sodass letztlich der Entwurf von *Ewing* übernommen wurde. Eine weitere Anekdote, die zur Auswahl des Pinguins beigetragen hat stammt von einem Erlebnis *Torvalds* in einem Aquarium in Canberra, Australien. Dort wurde er von einem Pinguin gebissen und sei seitdem mit der Krankheit *Penguinitis* infiziert:

„Penguinitis makes you stay awake at nights just thinking about penguins and feeling great love towards them.“ [2]

Es gibt inzwischen unzählige Versionen des Maskottchens, in dieser Arbeitsgruppe haben wir uns an einer modernen und jungen Version des Pinguins orientiert wie in Abbildung 1 zu sehen.



Abbildung 1: Die Hauptfigur des Filmes wurde nach diesem Vorbild entwickelt. Das gezeigte Modell stammt vom Autor *Overlord59* und wurde unter *Creative Commons BY-NC-SA* veröffentlicht. (Quelle: <http://tux.crystalxp.net/de.id.1568-overlord59-overlord59-tux-g2.html>)

In diesem Projekt wurde das Programm *POV-Ray* verwendet, um dreidimensionale Figuren und Objekte, sowie kurze Animationssequenzen zu erstellen. Die Animationssequenzen bestehen

¹lewing@isc.tamu.edu

dabei aus mehreren Bildern, die mittels *FFmpeg* in ein Videoformat konvertiert und anschließend mit *iMovie* geschnitten und vertont wurden.

1.1 Plot des Films

Zu Beginn des Films ist zunächst ein Ei zu sehen, welches einen immer größeren Riss bekommt. Der obere Teil des Eis bricht auf und letztlich schlüpft der Tux. Der Babytux erscheint und springt aus der unteren Hälfte des Eis heraus. Anschließend entdeckt der Pinguin, dass er Füße, Flügel und seinen Schwanz bewegen kann.

Ein durch das Bild fliegendes Flugzeug soll andeuten, wie Tux die Welt und dabei allerlei Sehenswürdigkeiten entdeckt. Diese werden anschließend in einer Bildergalerie gezeigt, durch die der inzwischen herangewachsene Tux hindurch läuft.

In der Abschlusssequenz genießt Tux seinen Lebensabend auf einem Schaukelstuhl vor seinem Haus.

1.2 Liste der POV-Ray-Module

environment.pov Aufbau der Umgebung; enthält Himmel und Boden.

egg.pov Geschlossenes Ei-Objekt.

tux.pov Körper des Tux ohne Accessoires.

bow.pov Pinke Schleife.

soother.pov Schnuller.

assempledTux.pov Zusammengebauter Tux mit Accessoires.

crack.pov Objekt zum Erstellen eines Risses im Ei-Objekt.

tuxIsBorn.pov Animation des wackelnden Eis und Animation der Risse; Tux erscheint im Ei mit animiertem Nuckeln am Schnuller.

babytuxDiscovers.pov Animation der beweglichen Gliedmaßen des Tux.

flying.pov Flugzeug, welches horizontal durch das Bild fliegt.

gallery.pov Die Bildergalerie mit dem hindurch laufenden Tux.

oldTux.pov Tux in einem Schaukelstuhl vor seinem Haus.

2 Statischer Aufbau der Figuren

Im Folgenden wird beschrieben mittels welcher POV-Ray-Funktionen und Objekte die einzelnen Figuren, Requisiten und Szenenbilder des Films erstellt wurden.

2.1 Konstruktion der Umgebung

In der Datei `environment.pov` wurden alle relevanten Objekte der Umgebung festgelegt. Der Himmel wurde mittels `sphere` der Boden mittels `plane` realisiert. Dabei wurde für den Himmel eine `color_map` eingesetzt um einen Farbverlauf herzustellen. Für eine unebene Struktur des Bodens wurde das Pattern `bumps` verwendet.

2.2 Aufbau des Pinguineis

Die Grundstruktur des Eis wurde von der Vorgängergruppe (Teil 1) übernommen und an unseren Film angepasst. Dazu wurden jeweils für den oberen Teil und den unteren Teil des Eis ein weiteres etwas kleineres Ei-Objekt erzeugt und mittels `difference` vom größeren Objekt abgezogen. Dadurch wird das Ei von innen hohl. Damit das jeweilige Objekt auch als hohl erkannt wird, wurde das innere Objekt minimal nach oben bzw. nach unten verschoben.

```
difference{
    object{ Egg_lowerpart }
    object{
        Egg_lowerpart
        translate <0, 0.1, 0>
        scale <0.9, 0.9, 0.9>
    }
}
```

2.3 Aufbau der Hauptfigur

2.3.1 Körper

Zu allererst wurden die Proportionen als `declare`-Anweisung festgelegt. Um die Proportionen unabhängig von der Größe des Tux gleich zu halten, wird nur die Höhe `tuxheight` variabel

gehalten. Die anderen Größen wie `tuxwidth` oder `radiustummy` wurden mit `tuxheight` berechnet.

Der Grundaufbau des Tux besteht aus zwei Kugeln, in POV-Ray `sphere` genannt. Einer unteren großen Kugel für den Unterleib und einer etwas kleineren für den Kopf oberhalb. Der Unterleib besteht zunächst aus *einer* Kugel. Zur Realisierung des weißen Bauches wurden zwei weitere `sphere`-Objekte erstellt, deren Schnittmenge (*intersection*) anschließend mit der großen Kugel vereinigt wurden (*union*).

```
union{
    intersection{
        sphere{ 0, radiustummy }
        sphere{ 0, radiustummy }
        scale <0.6, 1.5, 0.25>
        translate <0, 0, -radiustummy + 0.1>
    }
    pigment{ White }
    sphere{
        0, radiustummy
        pigment{ Gray10 }
    }
}
```

Die weiße Schnittmenge wurde mit den Funktionen `scale` und `translate` so verschoben und skaliert, dass der vordere Teil der Hauptkugel in den richtigen Proportionen weiß erscheint. Sowohl Kopf, als auch Unterleib wurden in einer `declare`-Anweisung als `head` und `tummy` global geltend gemacht. So können diese direkt angesprochen werden, ohne den Code immer wieder neu reproduzieren zu müssen.

Der Kopf des Pinguins wurde mit einer `sphere` generiert, die zwei Drittel der Größe des Unterleibes beträgt. In den Kopf wurden Augen mit schwarzen Pupillen eingelassen. Zunächst wurde die Pupille in einer `declare`-Anweisung festgelegt. Sie besteht wie der Bauch aus der Schnittmenge zweier Kugeln. Die beiden Augen wurden in `LeftEye` und `RightEye` deklariert. Hier wurden die Pupillen als Objekt mit einem weiteren `sphere`-Objekt vereinigt (*union*).

Der Schnabel des Tux wurde mittels eines `cone`-Objektes realisiert. Hierbei wurden Zentrum und Radius der beiden Enden, sowie die Skalierung des gesamten Objektes so gewählt, dass ein flach gedrückter Kegel entsteht.

2.3.2 Gliedmaßen

Der Tux besteht weiterhin aus zwei Füßen und zwei Flügeln. Die Flügel bestehen aus jeweils einem Objekt `Wing`, welches aus einer Differenz aus `cone` und `sphere` gebildet wurde. Die Füße bestehen aus dem Objekt `Foot`, der Schnittmenge aus `sphere` und `box`. Dadurch wurde die Sphäre halbiert und es ist nur die obere Hälfte sichtbar.

In der Rückansicht ist ein Schwanz zu sehen. Dieser wurde aus einer einfachen `cone` in passender Größe generiert und lässt sich als `Tail`-Objekt ansprechen.

2.3.3 Accessoires

Der Tux ist in diesem Film ein weibliches Jungtier, daher wurde eine Schleife (`Bow`) und ein Schnuller (`Soothie`) konstruiert. Die Schleife wurde zunächst als ein `PartBow`-Objekt deklariert, welches eine `cone` generiert. In einer `union` wurden anschließend zwei dieser Objekte zusammengefügt, wobei eines um 180° gedreht wurde. Der Knoten der Schleife wurde innerhalb der `union` mittels `sphere` umgesetzt.

Für den Schnuller wurde eine `sphere` und ein `torus` vereinigt. Die beiden Accessoires wurden in jeweils eine Datei ausgelagert, dadurch sind die Objekte vom eigentlichen Körper unabhängig und können bei Bedarf auch weggelassen werden.

3 Aufbau der einzelnen Animationssequenzen

Der überwiegende Teil der einzelnen Sequenzen wurde jeweils innerhalb einer Datei mittels der Funktion `clock` und mehreren `if/elsif/else`-Statements realisiert. Dazu wurde zu Beginn die Variable `MyClock` deklariert mit den jeweiligen Zeiteinheiten verglichen, um anschließend die gewünschte Bewegung festzulegen.

```
#declare My_Clock = Start + (End - Start) * clock;  
  
#if (My_Clock <= 1)  
/* tue etwas */  
  
#elseif (My_Clock <=2)  
/* tue etwas anderes */  
...
```

Mittels dieser Deklarationen war es möglich die folgenden kurzen Sequenzen zu erstellen.

3.1 Anfangssequenz: Erstes Anzeichen von Leben

Das Ei beginnt sich mehrere Male hin und her zu bewegen. Dazwischen gibt es immer wieder Pausen. Für das Wackeln des Eis wurde die sinus-Funktion verwendet. Diese wurde mit `MyClock` verknüpft und in eine `rotate`-Funktion integriert. Dadurch bewegt sich das Ei in Abhängigkeit der Zeit hin und her.

3.2 Sequenz 1: Schlüpfen des Tux

Zunächst bekommt das Ei einen Riss, welcher immer größer wird und sich um das ganze Ei ausdehnt. Sobald die obere Hälfte von der Unteren des Eis komplett getrennt ist, hebt die obere Hälfte ab und fliegt nach hinten weg.

Zur Realisierung dieser Szene wurde ein Objekt `Crack` erstellt. `Crack` besteht aus mehreren quadratischen `box`-Objekten die ineinander verdreht wurden.

```

union{
    box{
        <-0.5, 0, -0.5>
        <0.5, 0.1, 0.5>
    }
    box{
        <-0.5, 0, -0.5>
        <0.5, 0.1, 0.5>
        rotate <5, 20, 10>
    }
    ...
}

```

Dieses Objekt wurde so klein erstellt, dass es in das Ei-Objekt passt. Anschließend wurde das crack-Objekt abhängig von der Zeit in der x- und z-Richtung größer skaliert und eine Differenz zum Ei-Objekt gebildet. Die Animation zeigt im Ergebnis einen Riss, der sich immer mehr vergrößert und am Ende zwei Ei-Hälften erscheinen.

3.3 Sequenz 2: Babytux entdeckt seine beweglichen Gliedmaßen

Wie eingangs erwähnt wurden hier je nach Zeiteinheit die einzelnen Gliedmaßen bewegt. Dazu wurden die einzelnen Gliedmaßen aus der Datei `tux.pov` angesprochen. Diese wurden anschließend mit den Funktionen `rotate` oder `translate` bewegt, wobei diese mit der Funktion `clock` verknüpft wurden.

Dies ist auch die erste Sequenz, in welcher eine Kamerafahrt realisiert wurde. Dazu wurde die Funktion `camera` wie ein Objekt behandelt und zu den jeweiligen Zeiteinheiten entsprechend rotiert und translatiert.

3.4 Sequenz 3: Flugzeug

Das Flugzeug wurde aus den Beispieldaten von Friedrich A. Lohmüller [3] verwendet. Dazu wurde die Datei `Plane_00.inc` inkludiert und mit den Funktionen `rotate`, `scale` und `translate` an die Bedürfnisse angepasst. Zur Animation wurde das Objekt mittels `clock` translatiert.

3.5 Sequenz 4: Bildergalerie

Zur Realisierung der Bildergalerie wurden verschiedene Bilddateien an das Makro `PictureFrame` übergeben, welches in der Datei `pictureFrame.pov` implementiert wurde. Dieses Makro wurde anschließend für jede Bilddatei im Modul `gallery.pov` aufgerufen und entsprechend platziert. Dieses Modul wurde wiederum neben `walkingTux.pov` in die eigentliche Animationssequenz `bigTux.pov` inkludiert.

3.6 Abschlussequenz: Lebensabend

In der Abschlussequenz wurde der blaue Himmel mittels dem Modul `stars.inc` in einen Sternenhimmel geändert. Als Textur der `sphere` wurde `Starfield1` gewählt und skaliert. Bis auf Tux wurden alle Objekte von Friedrich A. Lohmüller [3] verwendet und angepasst. So wurden beispielsweise die Wände des Haus-Objektes weiß gefärbt. Dazu sind die entsprechenden Objekte wieder als Makros implementiert.

Dem Objekt Schaukelstuhl musste ein Winkel übergeben werden. Um ein Schaukeln zu simulieren, wurde dieser mit der Sinusfunktion und `clock` verknüpft. Tux wurde mit dem Schaukelstuhl in einer `union` vereinigt.

4 Fertigstellung des gesamten Films

Alle Module, die Animationssequenzen enthielten wurden mittels *FFmpeg* in Videoclips konvertiert. Das Kommandozeilenprogramm *FFmpeg* besitzt zahlreiche Parameter, welche Bildqualität und Format beeinflussen. In diesem Projekt wurden aus den einzelnen Bildern mp4-Dateien mit folgenden Parametern erstellt:

```
ffmpeg -framerate 24 -i DATEIEN.png -c:v libx264 -r 30 -pix_fmt yuv420p  
MOVIE.mp4
```

4.1 Zusammenfügen der einzelnen Sequenzen

Das Zusammenfügen der Animationssequenzen war mittels *iMovie* intuitiv möglich. Dazu wurden die Filme einfach per Drag and Drop in die Zeitleiste gezogen. Da die Geschwindigkeiten alle von der Funktion *clock* abhingen, mussten diese noch entsprechend beschleunigt bzw. verlangsamt werden.

Um den Übergang zwischen den einzelnen Clips zu optimieren, wurden die *iMovie-Übergangsstile* verwendet. Zum Anzeigen von kurzen Texten wurde die Titel-Funktion verwendet.

4.2 Vertonung

Der komplette Film ist mit Musik unterlegt. Dabei wechselt die Musik zu den jeweiligen Abschnitten. Für einen fließenden Übergang wurde die Fade-Funktion verwendet. Zu bestimmten Zeitpunkten sind Soundeffekte eingefügt, die alle aus der Bibliothek aus *iMovie* stammen.

4.3 Korrekturen

Übergänge, die nicht korrekt geschnitten werden konnten wurden durch Einzelbilder ersetzt, die in das Programm *iMovie* zum jeweiligen Zeitpunkt eingefügt wurden.

Literatur

- [1] Larry Ewing. Linux 2.0 penguins. <http://isc.tamu.edu/~lewing/linux/>. [Online; besucht 24. November 2015].
- [2] Michelle Delio. The story behind tux the penguin. <http://www.wired.com/2001/03/the-story-behind-tux-the-penguin/>, 2001. [Online; besucht 24. November 2015].
- [3] Friedrich A. Lohmüller. Beschreibungen und beispiele zu pov-ray. http://www.f-lohmueller.de/pov_tut/pov__ger.htm. [Online; besucht 22. Januar 2016].



Abbildung 2: Poster zum POV-Ray-Animationsfilm „Das wunderbare Leben des kleinen Tux“.