

# Prozedurale Generierung von Wirbeltierskeletten

Masterarbeit von

Nina Zimbel

An der Fakultät für Informatik  
Institut für Visualisierung und Datenanalyse,  
Lehrstuhl für Computergrafik

13. Mai 2020

Erstgutachter: Prof. Dr.-Ing. Carsten Dachsbacher  
Zweitgutachter: Prof. Dr. Hartmut Prautzsch  
Betreuernder Mitarbeiter: Dr. Johannes Schudeiske

# Inhaltsverzeichnis

<b>1. Einleitung</b>	<b>1</b>
<b>2. Biologie der Wirbeltiere</b>	<b>2</b>
2.1. Das Skelett . . . . .	3
2.2. Von Groß und Klein . . . . .	4
2.3. Mythologie . . . . .	5
<b>3. Grundlagen</b>	<b>7</b>
3.1. Prozedurale Generierung . . . . .	7
3.2. Animation, Physiksimulationen, IK . . . . .	7
3.3. Grammatiken . . . . .	8
3.4. PCA . . . . .	8
3.5. Quantil-Quantil-Diagramme . . . . .	9
3.6. Genetische Algorithmen . . . . .	9
<b>4. Bisherige Arbeiten</b>	<b>11</b>
4.1. 3D-Editoren . . . . .	11
4.1.1. Ziva . . . . .	11
4.1.2. ZSpheres in Zbrush . . . . .	11
4.1.3. 3DS MAX Biped . . . . .	11
4.2. Forensik und Archäologie . . . . .	12
4.3. Computerspiele und Filme . . . . .	12
4.3.1. No Man's Sky . . . . .	12
4.3.2. Film „Avatar“ . . . . .	12
<b>5. Erste Ansätze</b>	<b>13</b>
5.1. Bestandteile eines Skeletts . . . . .	13
5.2. Aufbau als Grammatik . . . . .	14
5.3. Pose des Skeletts . . . . .	14
<b>6. Beispielanalyse mit Hilfe von PCA</b>	<b>17</b>
6.1. Datenerhebung . . . . .	17
6.2. Analyse der Eingabedaten . . . . .	19
6.3. Analyse der Ergebnisse . . . . .	23
6.4. Bedingte Verteilungen . . . . .	24
<b>7. Erzeugung eines Skeletts</b>	<b>28</b>
7.1. Überblick über den Ablauf der Generierung . . . . .	28
7.2. Extremitäten . . . . .	29
7.2.1. Berechnung der Bodenhöhe . . . . .	30
7.2.2. Algorithmus zur Ausrichtung der Beine . . . . .	30
7.2.3. Zusätzliche Ansatzpunkte für Extremitäten . . . . .	33
7.2.4. Anordnung und Anzahl der Extremitäten . . . . .	33

7.3. Wirbel und Rippen . . . . .	34
7.4. Knochenmodelle . . . . .	35
<b>8. Zusätzliche Features</b>	<b>38</b>
8.1. Speichern und Laden von Skeletten . . . . .	38
8.2. Erzeugung von Variationen . . . . .	38
8.3. GUI . . . . .	39
<b>9. Implementierungsdetails</b>	<b>40</b>
9.1. Programmiersprache . . . . .	40
9.2. Dateiformate . . . . .	40
9.2.1. OpenSim . . . . .	40
9.2.2. OBJ . . . . .	41
9.2.3. FBX . . . . .	41
9.2.4. Alembic . . . . .	41
9.3. Aufbau der Software . . . . .	41
9.4. Zufall . . . . .	42
9.5. Transformationsmatrizen . . . . .	42
9.6. PCA . . . . .	42
9.6.1. Annotation der Bilder . . . . .	43
9.6.2. Anpassung der PCA-Ergebnisse zur Weiterverarbeitung . . . . .	43
9.7. Gelenke . . . . .	44
9.8. Probleme der Beinpositionierung bei kurzen Beinen . . . . .	44
<b>10. Erweiterungs- und Verbesserungsmöglichkeiten</b>	<b>46</b>
10.1. Muskeln . . . . .	46
10.2. Haut . . . . .	46
<b>Literatur</b>	<b>47</b>
<b>Anhang A. Vergleichende Anatomie</b>	<b>50</b>
<b>Anhang B. D'Arcy Thompson</b>	<b>53</b>
<b>Anhang C. Erhobene Daten für PCA</b>	<b>54</b>
C.1. Mit PCA erzeugte Datenpunkte . . . . .	54
C.2. Skelettbilder . . . . .	54
C.3. Gewicht der Tiere . . . . .	59



# 1. Einleitung

Idee und Ziel

- Theorie + Implementierung
- prozedurale Generierung
- Es gibt schon Pflanzen, Landschaften, Wolken etc. aber keine Tiere
- viele verschiedene (und möglicherweise doch ähnliche) Tiere generieren für Spiel/Film
- Inspiration für Künstler, 3D-Animation
  - Skelett darf nicht zu abstrakt sein, da es sonst zu wenig Informationen zum konkreten Tier liefert.
  - Ein detailliertes Skelett ist für Wesen sinnvoll, die es so noch nicht gibt bzw. wo man keine richtige Vorstellung davon hat wie es „funktioniert“, z.B. bei mehr Gliedmaßen.
- Einschränkung auf Skelett, da sonst zu umfangreich für eine Masterarbeit
- Wirbeliere, da Skelette sehr ähnliche
- möglicherweise interaktive Bedienung
- auch fiktive Tiere

## 2. Biologie der Wirbeltiere

*„Nichts anderes in der Natur hat eine herrlichere Struktur als der Körper der Wirbeltiere.“*

— „Vergleichende und funktionelle Anatomie der Wirbeltiere“ [11, S. 1]

Dieses Kapitel soll einen Überblick über die biologische Beschaffenheit der Wirbeltiere geben. Da dies keine biologische Arbeit ist, werden nur diejenigen Merkmale und Eigenschaften weiter ausgeführt, die im Folgenden zum Verständnis nötig sind.

Wirbeltiere, oder auch Vertebrata, sind Tiere mit einer skelettartigen Schädelkapsel, einem Cranium. Deshalb werden sie auch Schädeltiere oder Craniota genannt.

Auf den ersten Blick ist diese Definition etwas überraschend, da als Haupteigenschaft nicht die Wirbelsäule, sondern der Schädel genannt wird. Das liegt daran, dass zu den Wirbeltieren auch einige Tiere gezählt werden, die gar keine Wirbelsäule sondern „nur“ eine Chorda dorsalis besitzen. Die Chorda dorsalis besteht aus weichem Gewebe, nicht aus Knochen. Sie ist das Achsenskelett der Chordatiere, einem Tierstamm, zu dem auch die Wirbeltiere gehören. [11, S. 27 f.]

Hier in dieser Arbeit soll es aber nur um Wirbeltiere mit Wirbelsäule gehen.

Es gibt fünf große Gruppen von Wirbeltieren. Die historisch ersten Wirbeltiere sind die Fische. Danach entwickelten sich die Tetrapoden, Wirbeltiere mit vier Gliedmaßen, zu denen alle anderen Gruppen gehören. Zunächst entstanden die Amphibien, die noch nicht vollkommen terrestrisch leben. Daraus entwickelten sich die Reptilien, die erste Klasse der Wirbeltiere, die alle Strukturen besitzt um vollkommen an Land zu leben. (Es gibt aber auch Reptilien die trotzdem wieder im Wasser leben.) Außerdem gibt es noch die Vögel, die sich auf das Fliegen spezialisiert haben, und die Säugetiere, die ihre Jungen säugen. [11, Kapitel 4]

Auch der Mensch ist ein Säugetier und besitzt eine Wirbelsäule. Da sich Wirbeltiere, vor allem in ihrem Skelett, sehr ähneln, lassen sich viele der im folgenden Abschnitt aufgeführten Eigenschaften des Skeletts, vermutlich leicht „am eigenen Leib“ nachvollziehen.

## 2.1. Das Skelett

*„Das innere, gelenkige Skelettsystem der Vertebraten ist einzigartig im Tierreich. Es ist das wichtigste aller Organsysteme für das Studium der Wirbeltiermorphologie.“ — [11, S. 131]*

Wirbeltiere haben ein inneres Skelett und ihr Körper ist bilateralsymmetrisch [11, S. 27]. Das bedeutet, dass ihre linke Körperseite symmetrisch zur rechten ist.

### Wirbelsäule

Der Aufbau und die Form der Wirbelsäule hängt von der Lebensweise des entsprechenden Wirbeltiers ab. Ein wesentlicher Einflussfaktor sind die Kräfte, die auf sie wirken. Bei Fischen muss die Wirbelsäule dem Druck der starken Axalmuskeln, die seitlich am Körper entlang führen, entgegenwirken. Bei Tetrapoden sind diese Axalmuskeln zurückgebildet, dafür muss die Wirbelsäule aber der Schwerkraft widerstehen. Und bei Vögeln ist die Wirbelsäule speziell an den Flug angepasst und deshalb u. a. besonders leicht. [11, Abschnitt 9.2, S. 168 ff.]

Auch die Anzahl der Wirbel unterscheidet sich teilweise erheblich. Die Halswirbelsäule ist z. B. bei den Fischen noch gar nicht herausgebildet. Amphibien haben einen Wirbel, der für die Beweglichkeit des Kopfes zuständig ist. Bei Reptilien ist die Halsregion meist schon stärker abgesetzt. Säugetiere haben, bis auf wenige Ausnahmen, genau 7 Halswirbel, egal wie lang der Hals ist. Vögel haben die meisten Halswirbel, nämlich 10 bis maximal 31 beim Trauerschwan [28], meistens aber zwischen 15 und 20. [11, Abschnitt 9.2]

Ebenso unterscheidet sich die Anzahl der Wirbel in anderen Teilen der Wirbelsäule. Teilweise sind Wirbel sogar miteinander zu einem größeren Knochen verwachsen. Betrachtet man nur die Gruppe der Säugetiere, so kommt man auf etwa 12 bis 14 Brustwirbel, 5 bis 7 Lendenwirbel, 3 bis 5 Kreuzwirbel und 4 bis 22 Schwanzwirbel [7].

Auch die Form der einzelnen Wirbel unterscheidet sich sowohl zwischen den verschiedenen Tieren als auch entlang der Wirbelsäule einer einzelnen Art [11, Abschnitt 9.1 und Abbildung 9.2]. Um diese Vielfalt zu abstrahieren wird hier mit Platzhaltern gearbeitet, die für alle Arten von Wirbeln stehen. Falls später eine genauere Unterscheidung nötig sein sollte, können diese leicht spezialisiert werden.

### Extremitäten

Die Extremitäten der Tetrapoden sind nach einem „Grundbauplan“ aufgebaut (siehe Abbildung 2.1a). Dieser wird jedoch vielfach abgewandelt. Einige Beispiele sind in Abbildung 2.1b zu sehen. [9, S. 487]

Bei Fischen haben die Extremitäten keine Verbindung zur Wirbelsäule. Erst Amphibien bilden Schulter- und Beckengürtel aus, da sie zur Fortbewegung an Land nötig sind. [11, Abschnitt 9.2.3] Der Einfachheit halber wird hier aber nicht weiter beachtet ob eine Verbindung zur Wirbelsäule besteht, oder nicht.

Schulter- und Beckengürtel bestehen jeweils aus mehreren Knochen, die je nach Art mehr oder weniger ausgebildet, unterschiedlich geformt oder sogar komplett zurückgebildet sein können [11, Absatz 9.7]. Um das im Folgenden zu vereinfachen, wird jeder Extremitätengürtel durch einen Knochen repräsentiert. Der Schultergürtel wird durch ein Schulterblatt ersetzt und der Beckengürtel durch einen „Beckenknochen“.

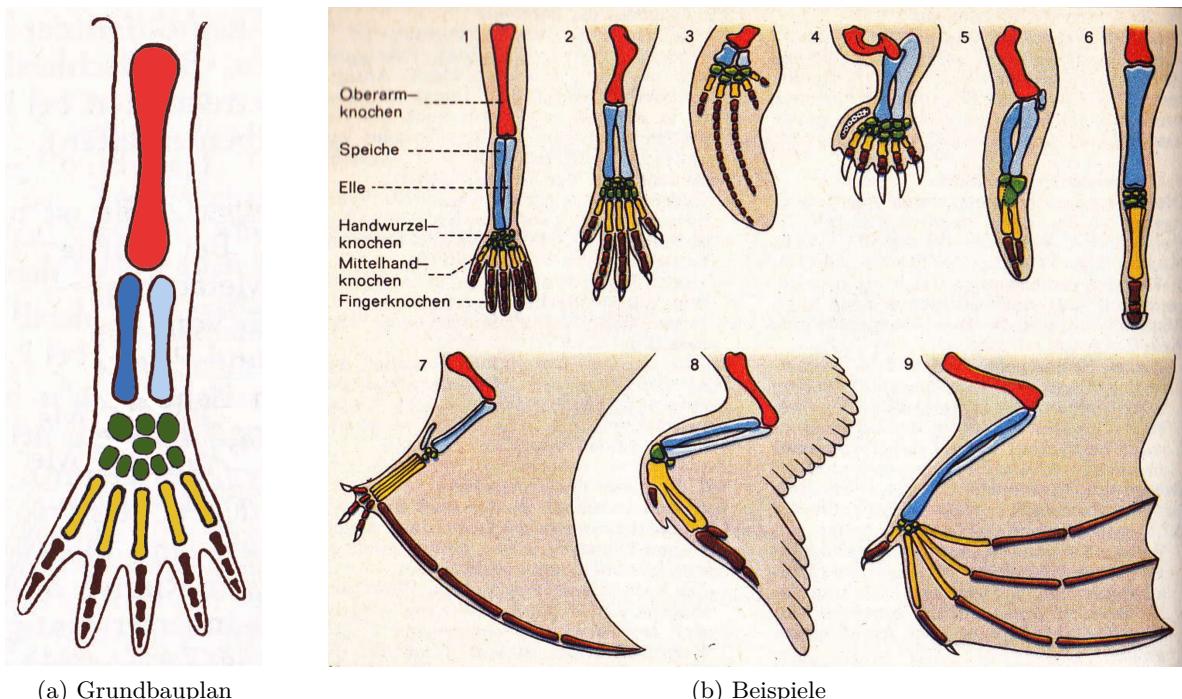


Abbildung 2.1.: Extremitäten von Tetrapoden, Oberarm rot, Speiche dunkelblau, Elle hellblau, Handwurzelknochen grün, Mittelhandknochen gelb, Finderknöchen braun  
 (a) Grundbauplan [9, S. 487, vereinfacht und eingefärbt],  
 (b) Beispiele für Vordergliedmaßen von Mensch (1), Eidechse (2), Wal (3), Maulwurf (4), Pinguin (5), Pferd (6), Flugsaurier (7), Vogel (8) und Fledermaus (9). [27, S. 474]

## Abstraktion

Das Skelett von Wirbeltieren ist im Allgemeinen also ein kompliziertes Konstrukt aus vielen Einzelteilen, die zwischen verschiedenen Arten stark variieren. Um einen Algorithmus zu entwerfen, der Skelette nachempfundene Gebilde konstruiert, ist es also notwendig ein erheblich vereinfachtes Modell zu erstellen.

In Abbildung 2.2 ist ein verallgemeinerter und abstrahierter Bauplan für Wirbeltiere zu sehen. Er ist reduziert auf die Wirbelsäule, Rippen, Schädel, Extremitäten, Schulterblatt und Beckenknochen. Die Extremitäten bestehen jeweils aus Oberarm/-schenkel, Unterarm/-schenkel und Hand/Fuß. Hände und Füße werden hier jeweils als ein „Knochen“ betrachtet, obwohl sie natürlich aus vielen Einzelteilen bestehen. Es kann maximal zwei Extremitätenpaare geben, jeweils einer an jedem Extremitätengürtel. Rippen, Hals- und Schwanzwirbelsäule sind optional und auch die Anzahl der Wirbel und Rippen variiert je nach Tier.

## 2.2. Von Groß und Klein

Dieser Abschnitt widmet sich der Frage nach Unterschieden und Gemeinsamkeiten von großen und kleinen Wirbeltieren. Kann man Wirbeltiere und ihre Skelette einfach vergrößern und verkleinern? Oder sind sie besonders an ihre Größe angepasst? Unterscheidet sich das Skelett einer Maus wesentlich von dem eines Elefanten?

Zunächst ein kurzer Überblick über die Dimensionen, in denen man sich hier bewegt: Das größte und schwerste Wirbeltier ist der Blauwal, er kann bis zu 120 Tonnen wiegen (siehe C.3). Das aktuell kleinste bekannte Wirbeltier, mit einer Länge von 7 bis 8mm, ist

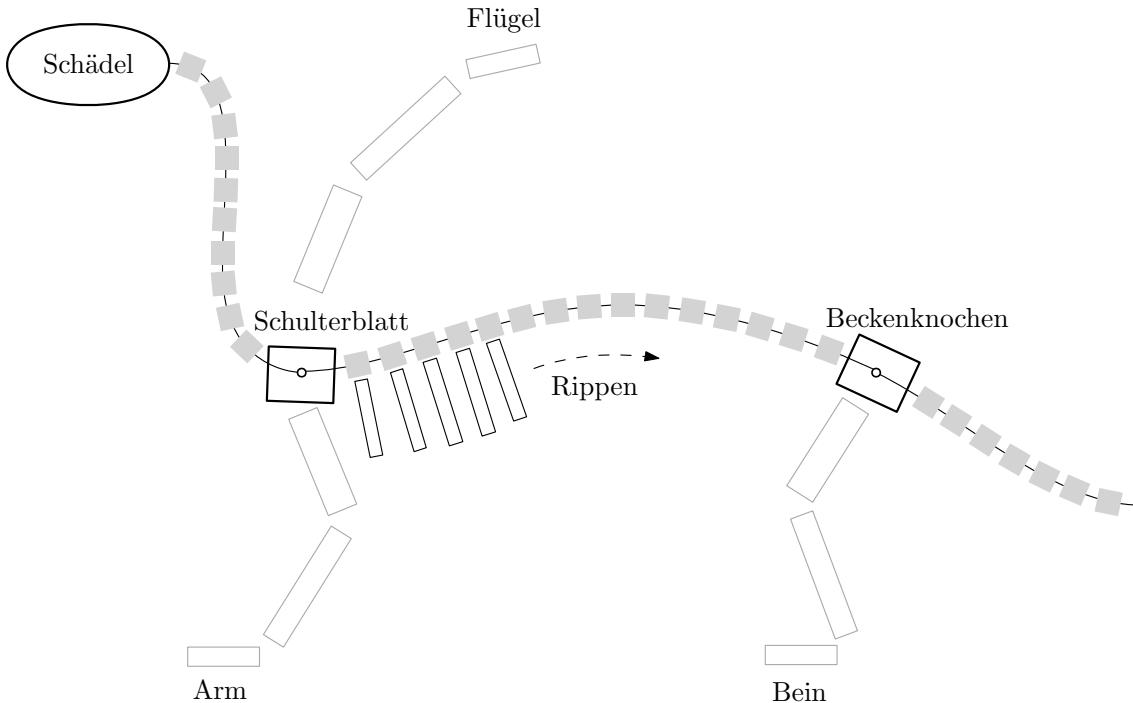


Abbildung 2.2.: Verallgemeinerter und abstrahierter Bauplan eines Wirbeltierskeletts. Es sind maximal zwei Extremitätenpaare erlaubt, jeweils eines an jedem Extremitätengürtel. Rippen und Hals- und Schwanzwirbelsäule sind optional. Die Anzahl der Wirbel und Rippen variiert je nach Tier.

der Frosch *Paedophryne amauensis* [21]. Größe und Gewicht von Wirbeltieren kann also erheblich variieren.

Wenn eine Tierart im Laufe der Evolution wächst, so ändert sich die Größe gleichmäßig. Dieses Wachstum ist aber begrenzt. Der Grund dafür ist, dass die Oberfläche nicht proportional mit dem Volumen mitwächst. Beispielsweise produziert ein Säugetier Wärme proportional zu seinem Volumen. Es kann aber nur Wärme proportional zu seiner Oberfläche abgeben. Dadurch haben große Säugetiere eher Probleme Wärme abzugeben und kleine warm zu bleiben. Ein ähnliches Problem tritt bei Knochen auf. Die Last, die ein Knochen tragen kann, ist ungefähr proportional zu seiner Querschnittsfläche. Das Skelett muss aber das komplette Gewicht des Körpers tragen, was wiederum proportional zum Volumen ist. Aus diesem Grund müssten die Knochen großer Tiere eigentlich überproportional vergrößert sein. Das ist jedoch nicht unbedingt der Fall. Das Problem an dicken Knochen ist, dass sie den Bewegungsradius, z. B. der Extremitäten, stark einschränken. Deshalb passen viele Tiere stattdessen ihre Körperhaltung und Aktivität an, um Spitzenkräfte, wie Stöße oder Oszillationen, auf ihre Knochen zu vermeiden. Deshalb wirken große Tiere auch eher „behäbiger“ als kleine. [11, Kapitel 23]

Sehr große und sehr kleine Wirbeltiere können also nicht über einfaches Skalieren ineinander überführt werden. Dennoch ähneln sich ihre Skelette sehr.

## 2.3. Mythologie

In diesem Abschnitt wird ein kurzer Blick über die Biologie hinaus, auf fantastische Tiere aus der Mythologie geworfen. Ein Algorithmus, der wirbeltierähnliche Skelette erzeugt, bewegt sich ja schon über die Grenzen der Biologie hinaus. Da ist es nur ein kleiner Schritt auch fiktive Wirbeltiere mit zusätzlichen Gliedmaßen zu erzeugen. Einen guten Überblick

über solche Tiere, nicht nur aus der Mythologie sondern auch aus aktuellerer Literatur, Filmen oder Spielen, gibt die Webseite [26].

Hier sollen drei Beispiele hervorgehoben werden, die jeweils charakteristische Merkmale besitzen. Sie stammen aus der Mythologie und bilden oft auch die Grundlage für andere fiktive Wirbeltiere.

*Zentauren* haben einen Pferdekörper, aber statt eines Halses setzt ein menschlicher Oberkörper auf dem Schultergürtel auf [17]. Sie haben also zwei Schultergürtel.

Der *Pegasus* hat ebenfalls den Körper eines Pferdes, aber auch Flügel, die zusätzlich zu den Vorderbeinen am Schultergürtel ansetzen [18]. Es ist also „möglich“ an einem Extremitätengürtel zwei Paare von Extremitäten ansetzen zu lassen.

*Asiatische Drachen* haben einen langen, schlangenartigen Körper mit oft mehr als vier Beinen. Sie können als Inspiration für Tiere dienen, die mehr als nur zwei Extremitätengürtel entlang des Rückens besitzen.

## 3. Grundlagen

### 3.1. Prozedurale Generierung

Werden Inhalte, wie Texturen oder 3D-Objekte, generiert, ohne dass diese vor der Ausführung des Algorithmus festgelegt wurden, so wird dies als *prozedurale Generierung* (PG) bezeichnet. Ursprünglich wurde PG verwendet, weil der Speicherplatz auf Computern sehr begrenzt war. Große 3D-Landschaften oder andere vorgefertigte künstlerische Werke konnten nicht gespeichert werden.

Die Demoszene entstand in den 1980er Jahren und treibt PG ins Extreme. Sie zeigt, dass beeindruckende Ergebnisse trotz stark limitiertem Speicherplatz möglich sind bzw. wie man das volle Potential von Computerhardware ausschöpft [5]. ([mehr beschreiben](#)) ToDo

Auf heutigen Rechnern ist der Speicherplatz nicht mehr so begrenzt und es kann viel Inhalt für z. B. PC-Spiele von Künstlern vorgefertigt werden. Trotzdem behält PG ihre Daseinsberechtigung, denn der Entwurf komplexer Inhalte, wie beispielsweise Landschaften mit viel Vegetation, ist aufwändig und benötigt viel Zeit. Hier können wieder Algorithmen aus der PG eingesetzt werden um Zeit und Arbeit zu sparen. Sie können sowohl verwendet werden um fertige 3D-Modelle zu generieren, als auch um als Inspiration und Hilfe für den Künstler zu dienen. *SpeedTree* [24] ist ein Beispiel für Software, die für die Generierung von Vegetation verwendet wird. Sie unterstützt und beschleunigt die Erstellung von Szenen mit viel Vegetation. [3]

Dinge, die oft prozedural generiert werden, sind Landschaften, Straßennetze, Gebäude, Menschen, Tiere oder auch Geschichten [8]. humanoide Charaktere für Computerspiele werden z. B. in der Masterarbeit [14] erzeugt. Die Zusammenstellung [3] untersucht PG von Wesen, die Videospiele bevölkern. Obwohl einige Arbeiten genannt werden, die sich mit diesem Thema beschäftigen, wird auch festgestellt, dass dieser Themenbereich noch wenig erforscht ist. Das lässt viel Spielraum für diese Arbeit.

([perlin noise \(nicht ausreichend für Skelette, mehr Einschränkungen nötig  
sonst würden nur Monster generiert\)](#) + [weitere Methodenbeispiele](#)) ToDo

### 3.2. Animation, Physiksimulationen, IK

- Knochen in Hierarchie (Baum)
- keyframes, motion capture (traditional keyframed workflow (sporeanim S.3), wo kann hier Algorithmus verwendet werden?)

- rag doll
- Skelett, Muskeln, Haut, Faszien (für lebensecht wirkende Tiere)
- Featherstone Algorithmus
- Drehmomente ausrechnen um Gleichgewicht auszurechnen (hier? oder im Kapitel zu ersten Ansätzen lassen?)
- Animation von vorher unbekannten Kreaturen (sporeanim)
- skeleton-based design tool (Yamamoto et al 2011): rudimentäres Skelett/Rig für schnelle Produktion und leichte Änderung an Charakteren <https://dl.acm.org/doi/10.1145/2073304.2073316>, <https://www.youtube.com/watch?v=yhq1aUp8QLY>

Inverse Kinematik wird unter anderem in der Robotik verwendet. Gegeben ist eine Kette von Gelenken, die über starre Verbindungsteile miteinander verbunden sind. Für ein oder mehrere Punkte auf dieser Kette können Zielpunkte im Raum angegeben werden. Das Ziel ist es dann eine Konfiguration der Gelenke zu berechnen, so dass alle Zielpunkte erreicht werden.

Es gibt viele verschiedene Methoden Probleme aus der inversen Kinematik zu lösen. Diese finden jeweils in Abhängigkeit von der konkreten Problemstellung ihre Anwendung. Ein Überblick über die verschiedenen Techniken ist in Zusammenstellung [1] zu finden.

- IK für die Animation von Beinen: Girard and Maciejewski 1985
- IK die gelernte Posen bevorzugt: style-based inverse kinematics (<https://grail.cs.washington.edu/projects/styleik/>)

### 3.3. Grammatiken

Eine *Grammatik*  $G$  [15, S. 3] ist ein Tupel  $G = (\Sigma, V, S, P)$  mit

- dem endlichen Alphabet  $\Sigma$  der Terminalsymbole
- dem endlichen Alphabet  $V$  der Nichtterminalsymbole
- dem Startsymbol  $S \in V$
- der Menge von Produktionen  $P \subseteq (V \cup \Sigma) * V(V \cup \Sigma) * \times (V \cup \Sigma) *$   
der Form  $P \subseteq (V \cup \Sigma) * V(V \cup \Sigma) * \rightarrow (V \cup \Sigma) *$
- Definition + kontextfrei und kontextsensitiv
- L-Systeme (parallele Produktion, auch kontextsensitiv/-frei) [10], Beispiel Pflanzen in selber Thesis, Städte [19]

### 3.4. PCA

#### ToDo

#### (Pseudocode?)

*Principal Component Analysis* (PCA) oder auch Hauptkomponentenanalyse [13] wird meist mit dem Ziel angewendet die Dimensionalität einer Menge von Datenpunkten zu verringern, dabei aber möglichst wenig Information zu verlieren. So wie beispielsweise in [4] bei 3D-Modellen von Gesichtern oder in [22] bei 3D-Modellen von menschlichen Körpern. Auch im Zusammenhang mit prozeduraler Generierung wird PCA oft verwendet, z. B. zur Generierung von humanoiden Charakteren für Videospiele [14] oder Texturen für Gesichter [2].

Als Ausgangspunkt für eine PCA dient eine Menge von Datenpunkten (oder Beispielen) mit jeweils  $n$  Dimensionen. Voraussetzung ist, dass die Punkte in jeder Dimension normal-/gaußverteilt sind. Dann liegen die Punkte im  $n$ -dimensionalen Raum in einem Ellipsoid.

Nun ist das Ziel herauszufinden wo die Achsen des Ellipsoids liegen. Entlang der Richtungen der Achsen sind die Beispiele wiederum normalverteilt und die Verteilungen sind jeweils unabhängig voneinander. Jetzt können die Datenpunkte vom Eingabekoordinatensystem in dasjenige Koordinatensystem transformiert werden, das von den Achsen des Ellipsoids aufgespannt wird. Interessant sind dabei die Achsen in deren Richtung die Daten die große Streuung aufweisen, die „principal components“.

Oft ist das Ziel die Dimensionalität der Eingabedaten zu reduzieren. Das wird dadurch erreicht, dass die Eingabedaten nur noch durch die „principal components“ dargestellt werden und die anderen Dimensionen weggelassen werden. Je weniger Streuung die Datenpunkte auf denjenigen Achsen aufweisen, die weggelassen werden, desto weniger Information wird verworfen.

Will man stattdessen einen zufälligen Datenpunkt erzeugen, der die gleichen Verteilungen aufweist, wie die Eingabebeispiele, so funktioniert das im transformierten Koordinatensystem ebenfalls sehr gut. Da die einzelnen Dimensionen unabhängig voneinander sind, kann man einfach für jede Dimension eine normalverteilte Zufallszahl mit der entsprechenden Varianz ziehen.

Der Mittelpunkt des Ellipsoids ist der Mittelwert der Daten. Um die Achsen des Ellipsoids zu berechnen, wird zunächst eine Kovarianzmatrix aufgestellt. Ihre Einträge sind die Kovarianzen aller möglichen Kombinationen der Eingabedimensionen. Von dieser Matrix werden dann alle Eigenvektoren mit zugehörigen Eigenwerten berechnet. Die Eigenvektoren sind die Achsen des Ellipsoids und die Eigenwerte geben an wie groß die Varianz entlang dieser Achse ist.

Um nun die Haupteigenschaften oder „principal components“ eines Datenpunktes zu finden, stellt man ihn im Koordinatensystem des Ellipsoids dar, also als gewichtete Summe der Eigenvektoren. Dann betrachtet man die Dimensionen mit den größten Eigenwerten. Dazu zieht man zunächst den Mittelwert vom Datenpunkt ab und multipliziert ihn mit der transponierten Basiswechselmatrix. Das ist diejenige Matrix, in deren Zeilen die Eigenvektoren der Kovarianzmatrix stehen.

### 3.5. Quantil-Quantil-Diagramme

- QQ Diagramme erklären
- negative Werte sind ganz normal, da inverse CDF bei einem Mittelwert von 0 bei Werten  $< 0.5$  negativ ist
- welche Methode um Unterscheidungsanteil auszurechnen? <https://de.wikipedia.org/wiki/Quantil-Quantil-Diagramm> (verschiedene Sachen ausprobieren, zeigen dass egal?)
- noch genauer: „Zahl“ ausrechnen (paper)? (oder als Alternative vorschlagen)

### 3.6. Genetische Algorithmen

als Alternative zu Variationen mit PCA

- introduction to evolutionary design (bently)
- basics on genetic algorithms (Holland 1992) -> bei Bib bestellt

- interactive evolutionary computation (Takagi 2001) (fitness function = human evaluation)
- jon hudson thesis: creature generation using genetic algorithms [12]
  - relativ simple Regeln, nicht orientiert an spezieller Tierklasse, Einschränkung für Anzahl Beine, für Houdini, kein Skelett sondern 3D-Modell, Rig wird auch generiert
  - auch Variationen eines Tiers und Crossover zwischen mehreren Exemplaren möglich
- Grammar based genetic programming (a survey)

## 4. Bisherige Arbeiten

### 4.1. 3D-Editoren

Aufbau der erzeugten / zu erzeugenden Skelette mit Aufbau von Skeletons, die für Animation verwendet werden, vergleichen (realistischer,...)

#### 4.1.1. Ziva

- Ziva VFX Maya Plugin zur Erstellung von Charakteren und Simulation von biomechanischen Bewegungen <https://zivadynamics.com/>
- Charaktererstellung in Ziva beginnt mit der Modellierung des Skeletts. Knochen mit Animationen werden als Alembic-Datei gespeichert und dann in „Ziva-Knochen“ konvertiert. <https://discover.therookies.co/2019/06/01/vfx-in-9-steps/>

#### 4.1.2. ZSpheres in Zbrush

- <http://docs.pixologic.com/user-guide/3d-modeling/modeling-basics/creating-meshes/zspheres/>,  
Beispielvideo: <https://www.youtube.com/watch?v=Wl0XK6ggU0A>
- Möglichkeit ein „Skelett“ aus Kugeln zu erstellen. Definiert aber eher die grobe Außenhaut mit Zusatzinformationen dazu wo die Gelenke sind.

#### 4.1.3. 3DS MAX Biped

- <https://knowledge.autodesk.com/support/3ds-max/learn-explore/caas/CloudHelp/cloudhelp/2019/ENU/3DSMax-Character-Animation/files/GUID-2F6BC5D1-DD45-4C2E-AC3A-D80.html>
- Möglichkeit Skelett in einen fertig modellierten Körper einzupassen.
- Skelette sind schon vorgefertigt.
- v.a. für menschliche Skelette, aber auch (limitiert) anpassbar auf Tiere

## 4.2. Forensik und Archäologie

- forensische Gesichtsrekonstruktion ist spezialisiert auf Menschen und verwendet Zusatzinformationen wie Stockfotos von Gesichtsmerkmalen ([https://en.wikipedia.org/wiki/Forensic\\_facial\\_reconstruction](https://en.wikipedia.org/wiki/Forensic_facial_reconstruction))
- Rekonstruktion von Tieren in der Archäologie anhand des Skeletts v.a. durch Künstler (?)

## 4.3. Computerspiele und Filme

### 4.3.1. No Man's Sky

- Webseite [16]
- „For creatures, basic templates of creatures that exist on the Earth were created and then manipulated by the system, changing everything from height, weight, bone density, voice pitch, what it eats, and its behaviors, even creating variation within the species.“ (<https://nomanssky.fandom.com/wiki/Biology>)
- „Creatures were often generated by mixing and matching random parts from a library, and then adjusting the underlying skeleton so that the creature appeared realistic; a creature with a tiny body could not support a giant head, for example.“ ([https://en.wikipedia.org/wiki/Development\\_of\\_No\\_Man%27s\\_Sky](https://en.wikipedia.org/wiki/Development_of_No_Man%27s_Sky))
- Zunächst Generierung von äußerem 3D-Modell, dann Anpassung der Knochen.
- create creatures from DB (Ribeiro et al 2003) (hat das was mit no mans sky zu tun?)

### 4.3.2. Film „Avatar“

- Direhorse (Schreckenspferd) hat sechs Beine (vier vorne, zwei hinten).  
Bilder: [https://james-camerons-avatar.fandom.com/wiki/Gallery:\\_Pandoran\\_Creatures?file=Muscle.jpg](https://james-camerons-avatar.fandom.com/wiki/Gallery:_Pandoran_Creatures?file=Muscle.jpg), [https://james-camerons-avatar.fandom.com/wiki/Gallery:\\_Pandoran\\_Creatures?file=Pandora\\_ROVR\\_Direhorse.png](https://james-camerons-avatar.fandom.com/wiki/Gallery:_Pandoran_Creatures?file=Pandora_ROVR_Direhorse.png)
- Prollemuris hat Arme, mit einem Oberarm, aber zwei Unterarmen.  
Bild: [https://james-camerons-avatar.fandom.com/wiki/Gallery:\\_Pandoran\\_Creatures](https://james-camerons-avatar.fandom.com/wiki/Gallery:_Pandoran_Creatures)

## 5. Erste Ansätze

### 5.1. Bestandteile eines Skeletts

Ein Skelett besteht aus Knochen und Gelenken. Zwei Knochen sind jeweils durch ein Gelenk miteinander verbunden. Im Folgenden werden sowohl Knochen als auch Gelenke genauer beschrieben.

Jeder *Knochen* hat sein eigenes lokales Koordinatensystem und wird zunächst als Quader dargestellt. Der Ursprung des Koordinatensystems befindet sich in einer Ecke des Quaders. Informationen, die für die Darstellung eines Knochens erforderlich sind, sind also seine Ausdehnung in alle drei Raumrichtungen (die Kantenlängen des Quaders) und die Position und Orientierung des Ursprungs im globalen Koordinatensystem.

Die Knochen sollten eine Hierarchie (einen Baum) bilden, da das von Algorithmen für Animationen so erwartet wird. ([Verweis auf previous work, wenn ausgearbeitet](#))  
ToDo  
Da der Wirbeltierbauplan aus Abbildung 2.2 verwendet wird, ist das auch möglich. Im Allgemeinen ist ein Skelett aber weder zusammenhängend noch ohne Kreise.

Für eine Hierarchie ist ein erstes Element nötig, die Wurzel des Baums. Dafür bietet sich ein Knochen in der Nähe des Schwerpunkts an. Oft wird hierfür die Hüfte verwendet.

(**Beispiel/Quelle**) Da aber nicht jedes Wirbeltier eine Hüfte besitzt und es die Generierung vereinfacht, wird als Wurzelknochen ein Knochen ohne Ausdehnung in der Mitte der Rückenwirbelsäule verwendet. Dieser Knochen wird im Folgenden als *Wurzelknochen* bezeichnet. Liegt Knochen *B* in der Hierarchie direkt unter Knochen *A*, so ist *A* der *Elternknochen* von *B* und *B* ein *Kindknochen* von *A*.

Es ist sinnvoll für jeden Knochen nicht die Position im globalen Koordinatensystem anzugeben, sondern seine Position im Koordinatensystem des Elternknochens. Verfolgt man den Pfad von einem Knochen zurück zum Wurzelknochen, so kann die Position im globalen Koordinatensystem trotzdem ausgerechnet werden. ([Verweis auf IK](#))  
ToDo

Für die Darstellung werden Transformationsmatrizen mit homogenen Koordinaten verwendet. Genauere Informationen zu diesen Transformationsmatrizen und wie sie in verschiedenen Situationen berechnet werden können sind in Absatz 9.5 zu finden.

Ein *Gelenk* ist, wie auch in der Natur, ein Verbindungsstück zwischen zwei Knochen. Es legt fest wie die beiden Knochen sich relativ zueinander bewegen können. Im Gegensatz zu echten Gelenken haben Gelenke hier aber keine Ausdehnung. Sie werden am Ende im 3D-Modell nicht dargestellt.

Ein Gelenk wird im Koordinatensystem des Elternknochens dargestellt. Es wird beschrieben durch seinen Abstand zum Ursprung des Elternkoordinatensystems und Bewegungseinschränkungen für den Kindknochen. Ein Gelenk kann null bis zwei Freiheitsgrade haben. Bei einem Winkel von  $0^\circ$  hat das Kindelement die gleiche Ausrichtung wie das Elternelement. In Abbildung 7.1 sind alle Gelenke mit mindestens einem Freiheitsgrad, die im Algorithmus verwendet werden, dargestellt.

Ist ein Elternknochen mit einem Gelenk in einer bestimmten Ausrichtung gegeben, so lässt sich daraus die Transformationsmatrix des Kindelements berechnen.

## 5.2. Aufbau als Grammatik

- Aufbau als Grammatik (am Ende aber für jedes Nichtterminal genau eine Regel)
- Aufbau der (Nicht)terminale als Baum, nur Terminalen können Kinder haben, Reihenfolge der Anwendung der Regeln sollte beliebig sein
- Wachstum unter Berücksichtigung verschiedener Randbedingungen wie Bodenposition, Anzahl Extremitäten etc. Bounding Box für Nichtterminale ist dafür aber nicht nötig.
- Symmetrie der Skelette (spiegelt Elemente zum Schluss)
- Repräsentation des Zustands als Hierarchie von einzelnen Komponenten (terminale sowie nichtterminale)
- Iterative Erzeugung eines Skeletts durch eine probabilistische kontextfreie (?) Grammatik, die so erweitert ist, dass sie nicht ein einfaches Wort erzeugt, sondern einen Baum von Zeichen (nötig für Extremitäten). Verwendung von parametrischen L-Systemen [10] könnte sinnvoll sein.
- Regeln sind nicht wirklich eine Grammatik, da fast jedes nichtterminale Literal nur einmal vorkommt, wenn es für jedes Körperteil andere Regeln gibt. Oder ist es möglich so zu abstrahieren, dass z.B. Arme und Beine den gleichen/ähnlichen Regeln unterliegen? Ist das sinnvoll? (**Abstraktionsgrad, Art der Regeln**) Außerdem ist das Skelett nicht unbedingt zusammenhängend (siehe Biologie). → Darauf achten, dass das nicht von Algorithmus verlangt wird
- Brustbein sorgt dafür, dass Skelett nicht mehr baumartig → erstmal weglassen, ist wahrscheinlich auch nicht unglaublich relevant
- Kindelement erst erzeugen, wenn Elternelement terminal ist und Gelenk hat.

**To Do**

## 5.3. Pose des Skeletts

Ein Skelett ist nicht nur eine Hierarchie von Knochen, sondern wirkt auch wesentlich durch seine Pose. Das Ziel sollte also sein das Skelett in einer natürlich wirkenden Pose darzustellen, eine Pose, die das dargestellte Tier auch einnehmen würde.

Ein erster Ansatz könnte sein in jedem Schritt auszurechnen, ob das Skelett ausbalanciert ist. Dazu benötigt man den Schwerpunkt des Körpers und Position und Gewicht der Knochen. Damit kann man dann die Drehmomente der einzelnen Knochen um den Schwerpunkt berechnen. Addieren sie sich alle zu null auf, ist das Skelett im Gleichgewicht.

Hierbei tauchen gleich mehrere Fragen auf:

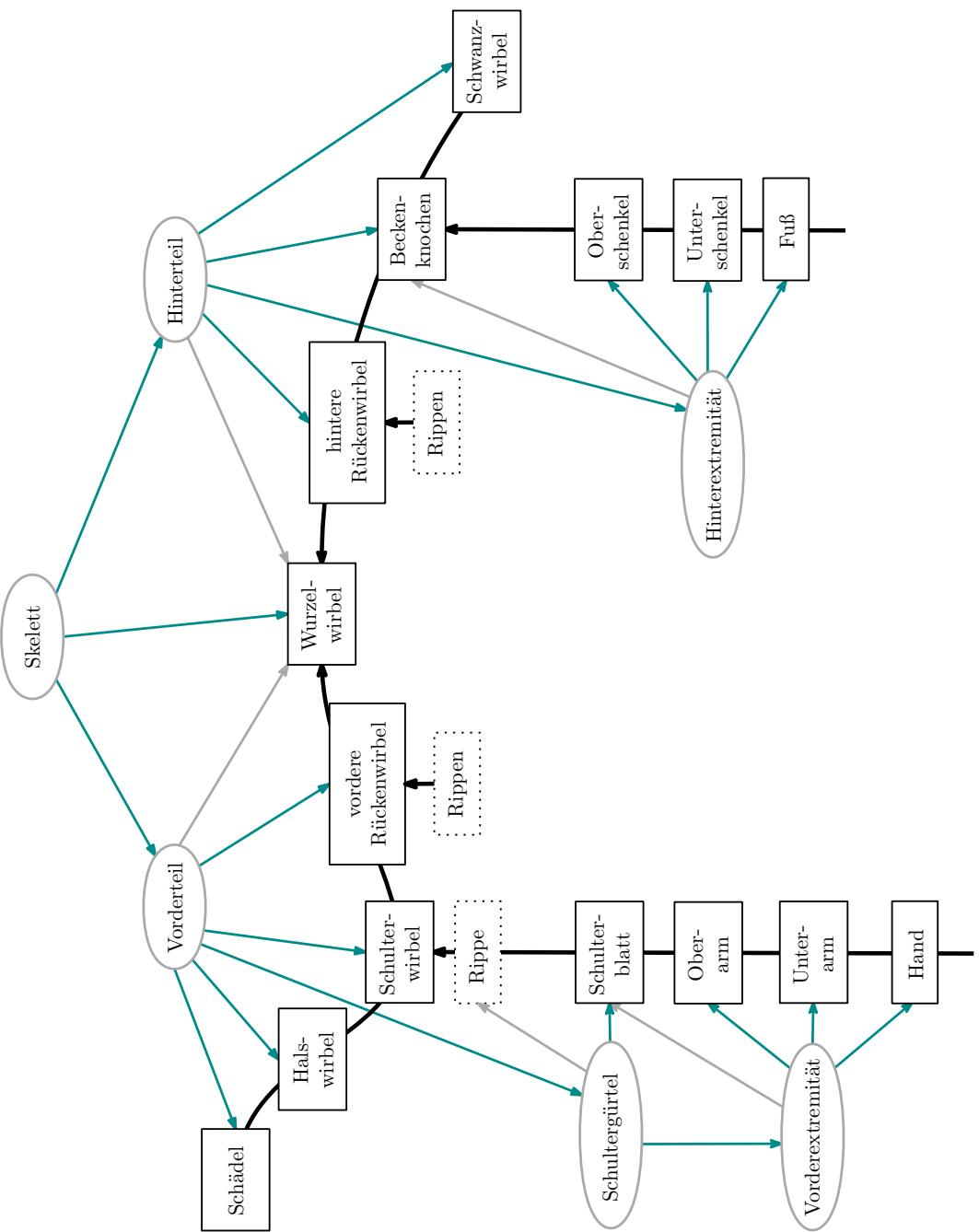


Abbildung 5.1.: Aufbau der Grammatik

**Gewicht** Wie wird das Gewicht der Knochen bestimmt? Es wird das Gewicht aller Elemente, oder zumindest das aller terminalen Elemente, benötigt. Wie wird es festgelegt? Hängt es von der Größe der Knochen ab? Außerdem müsste das ganze Gewicht des Tieres, nicht nur das der Knochen berücksichtigt werden. Wie kann bestimmt werden wieviel anderes Gewebe an einem Knochen hängt?

**Schwerpunkt** Wo liegt der Schwerpunkt? Wird am Anfang eine große Bounding Box für das Skelett festgelegt, in deren Mitte der Schwerpunkt liegt? Verändert der Schwerpunkt seine Position je nach dem was generiert wird?

**Gleichgewicht** Wann wird überprüft ob sich das Skelett im Gleichgewicht befindet? Ist das Gleichgewicht eine Invariante, die während der Generierung aufrecht erhalten werden soll? Oder wird es erst am Ende geprüft? Was passiert, wenn sich das Skelett nicht im Gleichgewicht befindet?

Das Hauptproblem ist, dass nicht klar ist wie das Gewicht der einzelnen Körperteile bestimmt werden soll ohne zusätzlich zu den Knochen auch noch anderes Gewebe wie Muskeln oder Eingeweide zu betrachten. Deshalb wurde ein anderer Ansatz erarbeitet.

Die Wirbelsäule ist ein zentraler Teil des Wirbeltierkörpers und bestimmt wesentlich das Aussehen des Skeletts. Ihre Form variiert von relativ gerade, z. B. bei Fischen oder Schlangen, bis hin zu stark geschwungenen Hälsen, v. a. bei Vögeln, und langen Schwänzen, z. B. bei Mäusen. Außerdem zeigt sie wie aufrecht ein Tier sich hält. Große Unterschiede sind hier zwischen Fischen und auch Vierbeinern gegenüber Vögeln zu beobachten, da Vögel nur auf ihren Hinterbeinen stehen und deshalb ihr Schwerpunkt nach hinten verschoben ist. Es gibt aber auch aufrechtere Exemplare unter den „Vierbeinern“ wie beispielsweise das Känguru oder der Tyrannosaurus Rex. (**Beispielbilder von Wirbelsäulen**)

### ToDo

Der Mensch ist natürlich auch ein Beispiel für ein sehr aufrechtes Wirbeltier. Seine Haltung unterscheidet sich aber so stark von der der anderen Wirbeltiere, dass er hier außen vor gelassen werden soll.

Viele Knochen setzen direkt an der Wirbelsäule an, wie z. B. der Kopf, die Rippen oder die Hüfte. Durch die Wirbelsäule wird also schon sehr viel vorgegeben. Deshalb eignet sie sich sehr gut als Startpunkt für die Generierung eines Skeletts. Ausgehend von ihr kann dann der Rest des Skeletts „wachsen“.

Wie soll aber nun die Lage der Wirbelsäule bestimmt werden?

Hierfür schien es sinnvoll viele Beispiele zu betrachten und Zusammenhänge zwischen verschiedenen Eigenschaften der Tiere und dem Verlauf der Wirbelsäule zu suchen. Ein geeignetes Werkzeug hierfür ist die *Principal Component Analysis* (siehe Abschnitt 3.4).

# 6. Beispielanalyse mit Hilfe von PCA

PCA kann hier verwendet werden um Position und Krümmung der Wirbelsäule bei Wirbeltieren zu untersuchen. Die Datenpunkte, die durch die PCA untersucht werden sollen, sind also Skelette von Wirbeltieren.

## 6.1. Datenerhebung

Die konkret erhobenen Beispiele sind vor allem der Datenlage bzw. der zugänglichen Quellen geschuldet. Trotzdem wurde darauf geachtet möglichst viele unterschiedliche Tierarten mit viel Variation in den erhobenen Merkmalen zu finden.

Viele Beispiele entstammen Zoologiebüchern, in denen sie als Beispiele für bestimmte Erklärungen angegeben waren (Bildquellen siehe Anhang C.2). Dem ist auch geschuldet, dass recht viele Dinosaurierskelette dabei sind. Denn von anderen Tieren gibt es als alternative Darstellungsmöglichkeit eine Außenansicht des lebenden Tieres. Das geht bei ausgestorbenen Tieren im Allgemeinen nicht.

Die Merkmale, die zur Datenerhebung ausgesucht wurden, sind charakteristisch für ein Skelett, tragen also viel zum Gesamteindruck bei. Das sind vor allem der Verlauf der Wirbelsäule und der Aufbau der Extremitäten.

Eingeschränkt wurde die Erhebung natürlich auch durch die begrenzte Datenlage. Am einfachsten zu bekommen sind 2D-Bilder mit Seitenansichten von Skeletten. Das schließt Merkmale aus, die Tiefeninformationen benötigen, z. B. den Abstand der Füße. Auch Informationen zu sehr kleinen Knochen, wie Handwurzelknochen oder die unterschiedlichen Fingerknochen, sind schwierig zu bekommen, da sie teilweise schwer zu erkennen und zu markieren sind. Deshalb ist die Erhebung auf folgende Daten eingeschränkt:

- Ein Bild mit der Seitenansicht des Skeletts. Darin wurde die Lage der *Wirbelsäule* und die Länge der Knochen der *Vorder- und Hintergliedmaßen* markiert, falls vorhanden. Hier hätte man zusätzlich auch die Positionen der Knochen und damit auch die Winkel an den Gelenken erheben können. Da die Position der Extremitäten auf den Abbildungen aber wenig vergleichbar und relativ beliebig ist, ist das nicht sinnvoll.
- Die *Tierklasse*, also ob das Tier ein Fisch, ein Amphibie, ein Reptil oder ein Säugetier ist. Dieses Merkmal lässt sich nicht auf einer kontinuierlichen Skala abbilden und ist deshalb nicht als Eingabedimension für die PCA geeignet. Es wurde trotzdem erhoben, da es für eine anderweitige Auswertung hilfreich sein könnte.

Auch andere Merkmale wie etwa der Lebensraum könnten für weitere Analysen interessant sein. Man könnte sie, je nach Ziel, in zukünftigen Analysen mitaufnehmen.

- Ob *Flügel* vorhanden sind oder nicht.
- Die Anzahl der Paare von *Beinen mit Bodenkontakt*. Hier werden sowohl Vorder- als auch Hinterbeine berücksichtigt.
- Das ungefähre *Gewicht* eines ausgewachsenen Exemplars des Tieres, zu dem das Skelett gehört, in Kilogramm. Dieses Merkmal wurde mitaufgenommen, da die tatsächliche Größe der Tiere nicht durch die Abbildungen dargestellt wird.  
Hier wurden oft Angaben zum maximalen Gewicht der Tiere verwendet, da keine Angaben zum Durchschnittsgewicht zu finden waren. Teilweise gibt es auch verschiedene (Unter-)Arten, die unterschiedlich schwer werden können, deren Skelette aber, bei der Auflösung der hier erhobenen Daten, gleich aussehen. In diesem Fall wurde ein beliebiger Wert gewählt, der zwischen dem Gewicht der leichtesten und dem der schwersten (Unter-)Art liegt. (Die Quellen zu den Gewichten sind im Anhang C.3 zu finden.)

Wie genau die Datenerhebung auf den Bildern der Skelette durchgeführt wurde ist im Kapitel „Implementierungsdetails“ im Abschnitt 9.6 nachzulesen. Das Ergebnis der Erhebung sind annotierte Bilddateien und eine Textdatei, in der Tierklasse, Flügel, Beine und Gewicht erfasst werden.

Die Lage der Wirbelsäule wird durch drei kubische Bézierkurven erfasst. Jeweils eine für Hals, Rücken und Schwanz. Hals und Rücken gehen an der Schulter ineinander über und Rücken und Schwanz an der Hüfte. Diese Bézierkurven benötigen 20 Eingabedimensionen der PCA (10 zweidimensionale Punkte). Bei manchen Tieren ist kein Hals oder kein Schwanz vorhanden. In diesen Fällen werden die 3 fehlenden Punkte jeweils mit dem ersten bzw. letzten Punkt des Rückens ersetzt.

Es wurden Bézierkurven verwendet, weil sie einfach in der Handhabung sind. Sie werden von Programmen unterstützt, mit denen Vektorgrafiken erstellt werden können, sind leicht in die Beispieldateien einzutragen und leicht zu interpretieren. Es könnten aber natürlich auch andere Repräsentationen, wie B-Splines, verwendet werden. B-Splines und Bézierkurven können sogar ineinander umgewandelt werden [20].

Bei der Annotation der Bilder sind folgende Schwierigkeiten aufgetreten:

- Bei Fischen ist nicht klar wo Rücken in Schwanz übergeht, da der Beckengürtel sich teilweise beim Kopf befindet oder auch gar nicht vorhanden ist (siehe auch Abschnitt 2.1). Bei der Datenerhebung wurde der Übergang ungefähr bei der Rücken- oder der Afterflosse festgelegt. Alternativ hätte man auch die komplette Wirbelsäule als Rücken klassifizieren können.
- Manche Hälse und Schwänze sind mit einer kubischen Bézierkurve nicht darstellbar. Das ist unter den verwendeten Beispielen der Hals von Ichthyornis und Schwan und der Schwanz von Ichthyosaurus und Koboldmaki. In diesen Fällen wurde versucht die Form möglichst gut anzunähern oder Fortsätze (wie am Schwanz vom Ichthyosaurus) einfach wegzulassen.
- Die Schwanzposition bei Tieren mit sehr langen Schwänzen ist auf den Bildern relativ beliebig. Hier wurde versucht den Schwanz möglichst gerade nach hinten fortzusetzen, auch wenn er auf dem Bild eine andere Position hat.

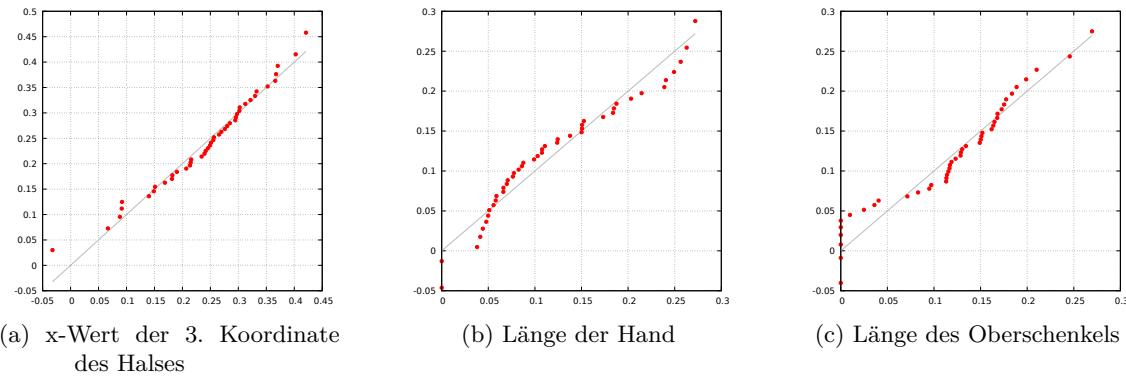


Abbildung 6.1.: Beispielhaft ausgewählte Quantil-Quantil-Diagramme von drei Eingabedimensionen. (a) weicht nicht stark von der Normalverteilung ab, (b) und (c) hingegen schon mehr, sind aber trotzdem noch akzeptabel verteilt.

## 6.2. Analyse der Eingabedaten

Insgesamt wurden 44 Datenpunkte erhoben. Das entspricht bei 29 Dimensionen  $44 \cdot 29 = 1276$  Zahlen als Eingabe. Das Ergebnis der PCA ist im wesentlichen die  $29 \times 29$  große Basiswechselmatrix bzw. die Eigenvektoren und 29 Eigenwerte. Das sind  $841 + 29 = 870$  Zahlen als Ausgabe. Immerhin gilt  $1276 > 870$  und  $44 > 29$ , dennoch wäre es besser mehr Datenpunkte zu haben.

Eine Voraussetzung dafür, dass die PCA korrekt funktioniert, ist, dass die Eingabedaten in jeder Dimension normalverteilt sind. Das soll im Folgenden genauer untersucht werden. Visualisierungen der Eingabedaten in den einzelnen Dimensionen sind in den Abbildungen C.1 und C.2 im Anhang zu finden.

Es gibt die beiden Merkmale *Flügel* und *Beine mit Bodenkontakt*, die offensichtlich nicht normalverteilt sind, da sie diskret sind und nur zwei bzw. drei Werte annehmen können. Sie können jedoch hilfreiche Informationen zur Weiterverarbeitung durch den Algorithmus liefern, weshalb sie nicht ganz außen vor gelassen werden sollten. Da das Hauptaugenmerk der PCA aber auf der Position der Wirbelsäule und dem Aufbau der Extremitäten liegt, sollen diese beiden Merkmale keinen großen Einfluss haben. Deshalb wurde der Einfluss dieser Merkmale verringert. Dazu weiter unten mehr.

Die anderen Dimensionen wurden mithilfe eines Quantil-Quantil-Diagramms (siehe Abschnitt 3.5) mit der Normalverteilung verglichen. Die Diagramme zeigen für alle Merkmale außer dem Gewicht, dass sie mehr oder weniger gut normalverteilt sind (siehe Beispiele in Abbildung 6.1 und eine vollständige Aufzählung im Anhang in den Abbildungen C.3 und C.4).

Das Merkmal *Gewicht* ist überhaupt nicht normalverteilt (siehe Abbildung 6.2 a und b). Verwendet man es als Eingabe für die PCA und generiert dann zufällig normalverteilte Punkte im Ergebnisraum, treten schnell Gewichte kleiner null auf.

Betrachtet man das Gewicht jedoch auf einer logarithmischen Skala, so ist es normalverteilt (siehe Abbildung 6.2 c). Deshalb ist es sinnvoll das Gewicht zunächst zu logarithmieren, bevor es in die PCA eingeht. Welcher Logarithmus hier zur Umrechnung der Daten verwendet wird schlägt sich nur als linearer Faktor nieder und ist eine Frage der Skalierung. Es wurde der Zehnerlogarithmus verwendet. Da das Gewicht, genauso wie die Merkmale *Flügel* und *Beine mit Bodenkontakt*, kein Hauptmerkmal für die Untersuchung sein sollte, wurde es ebenfalls kleiner skaliert.

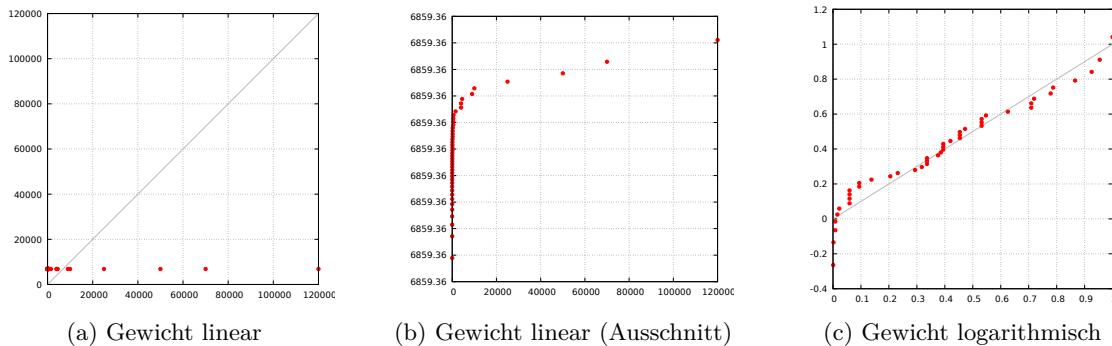


Abbildung 6.2.: Quantil-Quantil-Diagramme des Gewichts, einmal linear (a,b) und einmal mit logarithmischer Skala (b)

Generell bewirkt die Skalierung einer Dimension eine Gewichtung, denn durch eine Skalierung ändert sich die (Ko-)Varianz und somit auch die Kovarianzmatrix. Seien beispielsweise  $s, t \in \mathbb{R}$ , dann bewirkt eine Skalierung mit  $s$  in Dimension  $x$  und eine Skalierung mit  $t$  in Dimension  $y$  eine Skalierung von  $s \cdot t$  der Kovarianz  $\text{Cov}(x, y)$  von  $x$  mit  $y$ , da  $\text{Cov}(sx, ty) = (sx - s\mu_x)(ty - t\mu_y) = st \cdot \text{Cov}(x, y)$ , mit Erwartungswert  $\mu_i$  in Dimension  $i$ .

Wie genau wurden nun die einzelnen Merkmale nur skaliert?

Zunächst wurden alle Merkmale auf das Intervall  $[0, 1]$  skaliert, damit alle den gleichen Einfluss haben. Bei Koordinaten oder Längen im Bild bedeutet das, dass sie durch 1000 geteilt werden, da sie in Pixeln dargestellt werden und das Bild eine Größe von  $1000 \times 1000$  Pixeln hat. Bei Längen wären dabei theoretisch auch Werte größer 1000px möglich. Solche Längen wären aber unrealistisch und werden deshalb ignoriert.

Koordinaten und Langen im Bild sind diejenigen Merkmale, die hier am interessantesten sind. Deshalb sollten sie den groten Einfluss auf das Ergebnis der PCA haben. Alle anderen Merkmale werden deshalb kleiner skaliert.

Man könnte statt einer Skalierung mit 0.001 auch für jedes einzelne Merkmal den maximal und minimal angenommenen Wert ermitteln und sie dann so skalieren, dass sie Intervalle gleicher Länge abdecken. Das würde ausgleichen, dass z. B. kleine Längen eine kleinere Varianz und damit auch einen kleineren Einfluss haben. Es wurde aber die oben beschriebene Variante verwendet, da es natürlich wirkt, dass kleine Merkmale im Bild auch weniger wichtig sind. Falls es in Zukunft Gründe für eine andere Gewichtung gäbe, ließe sich das aber leicht anpassen.

Die diskreten Merkmale *Flügel* und *Beine mit Bodenkontakt* und das logarithmische *Gewicht* wurden zunächst ebenfalls auf das Intervall  $[0, 1]$  skaliert. Das bedeutet für das angepasste Gewicht  $\bar{w} = \frac{\log(w+1)}{\log(\max+1)}$ . Das schwerste Wirbeltier ist der Blauwal mit 120 Tonnen (siehe Abschnitt 2.2). Deshalb ist hier  $\max = 120.000\text{kg}$ .

Danach wurden die Werte noch einmal durch 100 geteilt, um ihren Einfluss zu verringern. Das Ziel war, dass diese Merkmale nicht als große Einträge in den größten Eigenvektoren (den „principal components“) auftauchen. Ohne diese Skalierung sind diese Merkmale recht dominant. Mit der Skalierung hingegen sind sie in den größten Eigenvektoren unter den kleinsten Werten zu finden.

Interessant ist die Projektion der Eingabedaten auf die größten beiden Eigenvektoren. In Abbildung 6.3 ist gut zu vergleichen was die Effekte der Skalierung der Eingabedaten sind. Ganz links sind die Ergebnisse zusehen, die entstehen, wenn alle Merkmale nur auf das Intervall  $[0, 1]$  skaliert werden. In der Mitte geht das Gewicht nicht mehr linear, sondern

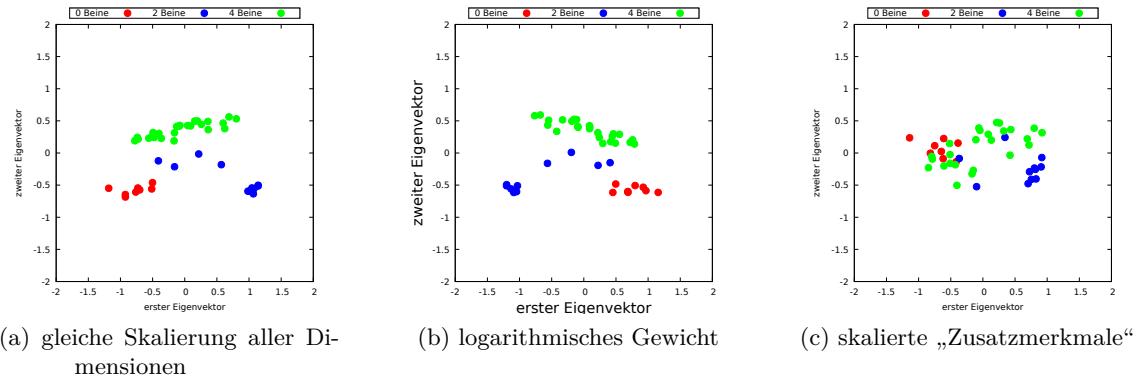


Abbildung 6.3.: Dargestellt sind hier jeweils die Projektionen der Eingabedaten auf die ersten beiden Eigenvektoren. Für jede Version wurden die Eingabedaten unterschiedlich vorverarbeitet. (a) Skalierung aller erhobenen Daten auf das Intervall  $[0, 1]$ , (b) zusätzlich Verwendung von logarithmischem Gewicht, statt linearem, (c) zusätzliche Skalierung der Merkmale *Flügel*, *Anzahl Beine* und *Gewicht* mit 0,01. In Version (b) wurde der erste Eigenvektor durch die PCA umgedreht, weshalb das Cluster der Zweibeiner auf der linken, statt der rechten Seite zu sehen ist.

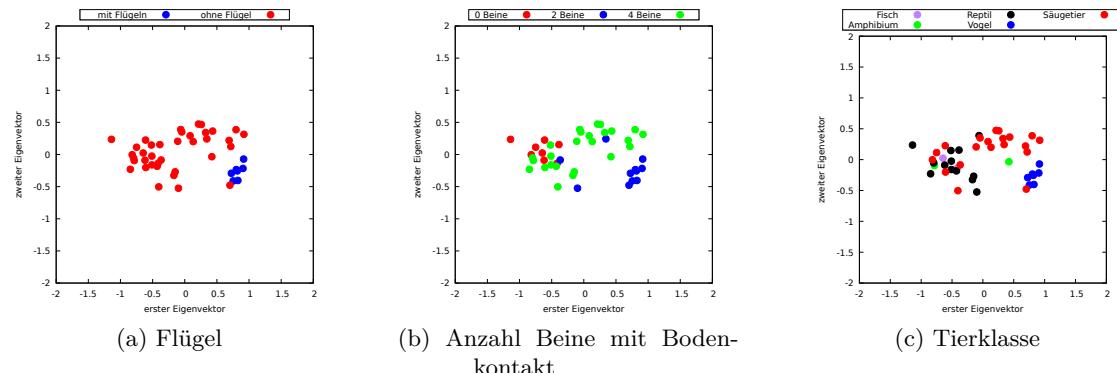


Abbildung 6.4.: Projektion der, wie in Abbildung 6.3c skalierten, Eingabedaten auf die Ebene, die durch den ersten und zweiten Eigenvektor aufgespannt wird. Markiert sind jeweils ob Flügel vorhanden sind (a), die Anzahl der Beine (b) und die Tierklasse (c).

logarithmisch ein und ganz rechts sind *Flügel*, *Anzahl Beine* und *Gewicht* zusätzlich klein skaliert. Gut zu sehen ist, wie sich die Clusterbildung durch die Anpassungen verringert.

In Abbildung 6.4 ist noch einmal jeweils die Projektion der skalierten Daten auf die ersten beiden Eigenvektoren zu sehen. Diesmal sind die Daten anhand der verschiedenen diskret erhobenen Merkmale markiert. Es ist z. B. schön zu sehen, dass alle Tiere mit Flügeln auch Vögel sind und dass fast alle Tiere, die zwei Beine haben, Vögel sind. Vier Tiere haben zwei Beine, sind aber keine Vögel: die Ohrenrobbe, der Seehund, der Tyrannosaurus Rex und das Känguru.

Auch im Koordinatensystem der Eigenvektoren sollten die Daten wieder normalverteilt sein. Dies wurde ebenfalls mit Quantil-Quantil-Diagrammen untersucht. In Abbildung 6.5 sind die Ergebnisse für die ersten drei Eigenvektoren zu sehen. Es gibt, wie bei den Eingabedaten, keine allzu großen Abweichungen.

Der Mittelwert der skalierten Eingabedaten ist in Abbildung 6.6 visualisiert. Wie auch bei der Datenerhebung ist hier die Position der Wirbelsäule in einem  $1000 \times 1000$  Pixel Bild

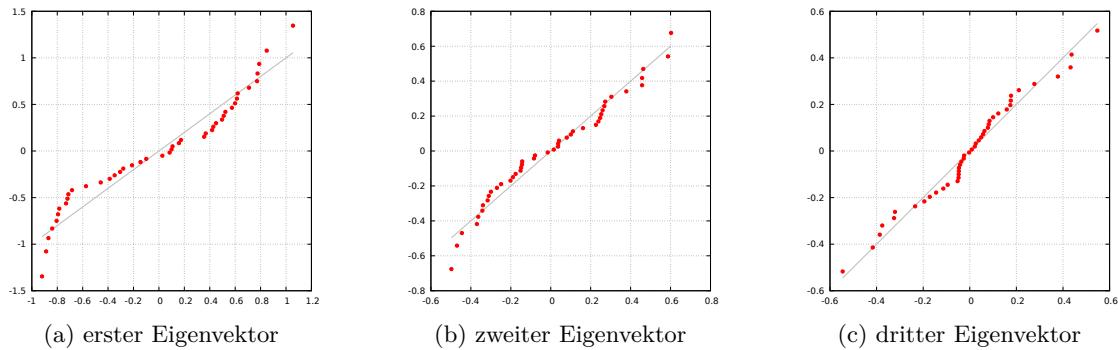


Abbildung 6.5.: Quantil-Quantil-Diagramme der Eingabedaten projiziert auf die größten drei Eigenvektoren.



Abbildung 6.6.: Visualisierung des Mittelwerts der skalierten Eingabedaten. Die Werte, die nicht grafisch visualisiert sind, sind folgende: *Flügel* 0, 159, *Beine mit Bodenkontakt* 1, 39, *Gewicht* 93kg.

gezeigt. Da von den Knochen der Vorder- und Hinterextremitäten nur die Längen erhoben wurden, sind ihre Positionen nicht realistisch. Die restlichen Daten sind nicht visualisiert, sondern nur in Textform angegeben.

Den minimalen Abstand zum Mittelwert hat der Klippschliefer (siehe Abbildung 6.7a). Zusätzlich zum Bild wurde für den Klippschliefer folgende Daten erhoben: *Tierklasse* Säugetier, *Flügel* nein, *Anzahl Beine mit Bodenkontakt* 4, *Gewicht* 4kg.

Den maximalen Abstand hat die Schlange. Die erhobenen Daten sind hier: *Tierklasse Reptil, Flügel nein, Anzahl Beine mit Bodenkontakt 0, Gewicht 50kg.*

Die Schlange ist allerdings das einzige Tier zu dem es kein „echtes“ Bild des Skeletts gibt. Das liegt daran, dass es keine seitlichen Abbildungen von ausgestreckten Schlangen gibt. Sie werden eigentlich immer gekrümmt dargestellt, da sonst das Bild sehr lang und schmal werden würde. Deshalb wurde für die Schlange nur eine horizontale Linie knapp über dem unteren Bildrand eingezeichnet, die den Rücken darstellen soll. Extremitäten und ersichtliche Punkte an denen der Rücken in Hals oder Schwanz übergeht gibt es bei Schlangen nicht.

Der Punkt mit dem zweitgrößten Abstand zum Mittelwert ist das Känguru (siehe Abbildung 1).

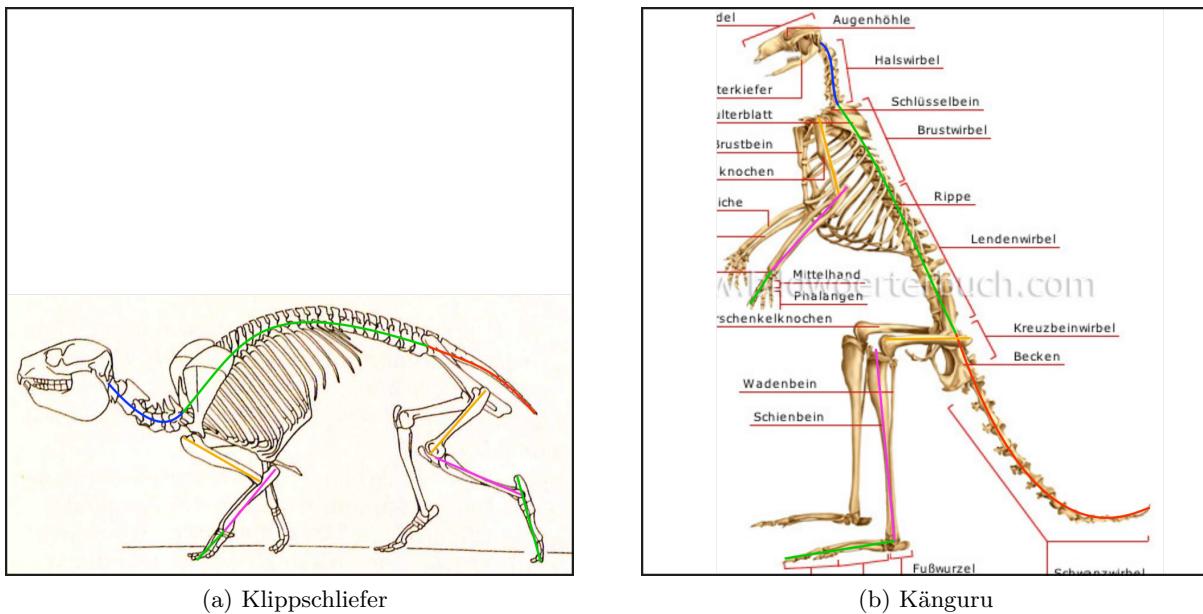


Abbildung 6.7.: Annotierte Bild des Skeletts eines Klippschließers (a) und eines Kängurus (b). Die Teile der Wirbelsäule und die Knochen der Extremitäten sind hier jeweils mit der gleichen Farbe markiert wie in der Visualisierung des Mittelwerts (Abbildung 6.6)

dung 6.7b). Zusätzlich zum Bild gibt es hier folgende Daten: *Tierklasse* Säugetier, *Flügel* nein, *Anzahl Beine mit Bodenkontakt* 2, *Gewicht* 50kg

### 6.3. Analyse der Ergebnisse

Zu 29 Eingabedimensionen gibt es auch 29 Eigenvektoren mit Eigenwerten größer 0. Der kleinste hat einen Eigenwert von 0,000001. Von den Eigenwerten sind 6 größer als 0,01. Leider reichen diese 6 Dimensionen noch nicht aus, um die Eingabedaten hinreichend anzunähern. Bei manchen Tieren funktioniert das zwar ganz gut (siehe Archaeopteryx, Abbildung 6.8), bei anderen aber eher schlechter (siehe Frosch, Abbildung 6.9). Die Daten konnten somit von der PCA nicht sehr gut komprimiert werden. Trotzdem sind die berechneten Eigenvektoren hilfreich um zufällige Punkte mit der Verteilung der Eingabebeispiele zu generieren (wie auch beschrieben in Abschnitt 3.4).

Um zu verstehen in welchem Bereich sich die zufällig generierten Punkte bewegen, ist es sinnvoll sich anzuschauen was die *principal components* „bedeuten“. In Abbildung 6.10 sind zwei Datenpunkte zu sehen, bei denen jeweils alle Dimensionen auf null festgelegt sind, nur der Wert für die Dimension des größten Eigenvektors ist die positive bzw. negative Standardabweichung. Die Varianz entlang eines Eigenvektors ist gegeben durch den Eigenwert. Die Standardabweichung ist also die Wurzel des entsprechenden Eigenwerts. Es ist zu sehen wie Punkte entlang des ersten Eigenvektors verändert werden bzw. welchen Einfluss er ausübt. Abbildungen für die sechs größten Eigenvektoren sind im Anhang in Abbildung C.5 zu finden.

Bei allen Eingabedimensionen, außer der Position der Wirbelsäule, kann man die Frage stellen, ob sie nötig sind, oder ob sie die Ergebnisse der PCA eher verschlechtern. Deshalb wurde ausprobiert verschiedene (Kombinationen von) Merkmalen wegzulassen. Die Ergebnisse unterscheiden sich aber nicht extrem von der PCA mit allen Daten.

Leider gibt es keine gute Möglichkeit die Qualität der Ergebnisse der PCA zu messen.

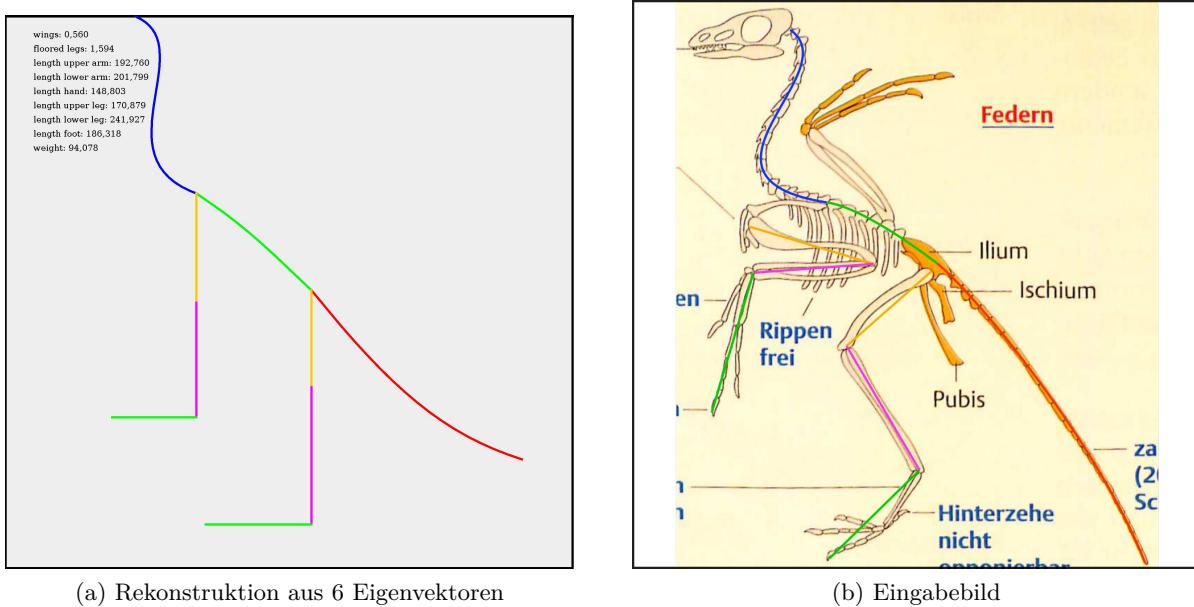


Abbildung 6.8.: Archaeopteryx (a) Rekonstruktion aus den größten 6 Eigenvektoren, nicht visualisierte Daten: *Flügel* 0, 56, *Beine mit Bodenkontakt* 1, 594, *Gewicht* 94, 1kg  
 (b) Eingabebild

Man könnte den Unterschied zwischen den Eingabedaten und den Rekonstruktionen aus den Linearkombinationen der Eigenvektoren mit den größten Eigenwerten messen. Aber das liefert, durch das Fehlen von verschiedenen Dimensionen kein einheitliches Maß. Da jede Dimension dem Algorithmus, der später Skelette generieren soll, helfen könnte, wurde kein Merkmal verworfen.

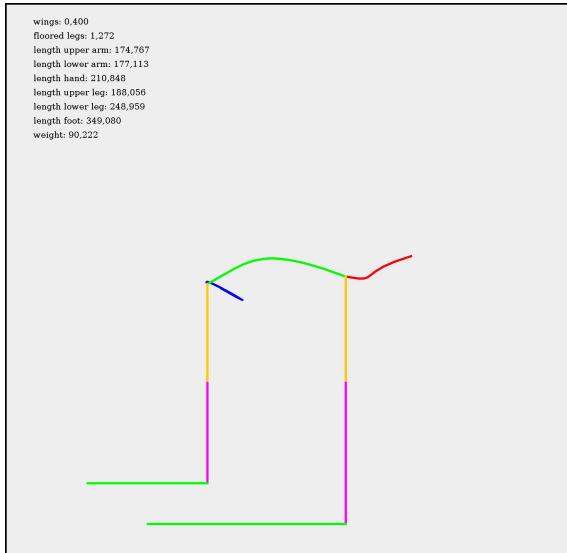
Außerdem gibt es die Möglichkeit die Eingabedaten in mehrere Mengen aufzuteilen und diese einzeln zu analysieren. Hierbei gibt es zunächst das Problem, dass sich dann die Anzahl der Datenpunkte noch weiter reduziert, was die Ergebnisse nicht mehr repräsentativ macht. Merkmale, die sich zur Unterteilung in Mengen anbieten würden, sind die diskreten, also *Flügel* und *Beine mit Bodenkontakt*. Sie sind auch klar als Cluster im Koordinatensystem der PCA ohne angepasste Skalierung zu erkennen (Abbildung 6.3 a und b). Getestet wurde die Aufteilung anhand der Werte für die Flügel, da sich dadurch nur zwei Gruppen ergeben. Tatsächlich liefert sie bessere Rekonstruktionen aus den größten Eigenvektoren. Das liegt aber natürlich in erster Linie daran, dass die zu untersuchende Datenmenge, jeweils verkleinert wurde.

Ein weiteres Problem daran die Daten in mehrere Mengen aufzuteilen ist, dass dann keine Skelette mehr erzeugt werden können, die zwischen den beiden Gruppen liegen. Tatsächlich scheinen aber die Datenpunkte, die „zwischen“ den Gruppen erzeugt werden, relativ sinnvoll auszusehen. Auch das ist ein Argument dafür keine Aufteilung vorzunehmen.

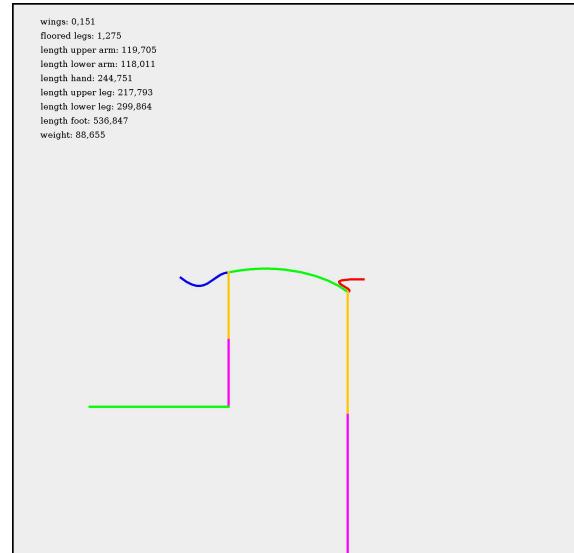
## 6.4. Bedingte Verteilungen

Aus den Ergebnissen der PCA lassen sich sehr gut zufällige Skelette erzeugen. Es ist aber schwer gezielt Eigenschaften festzulegen. Der erste Schritt dies zu erreichen ist Eigenschaften, die so schon in den erhobenen Daten vorkommen, auf gewählte Werte festzulegen. Das können beispielsweise die Anzahl der Beine sein oder ob das Skelett Flügel haben soll.

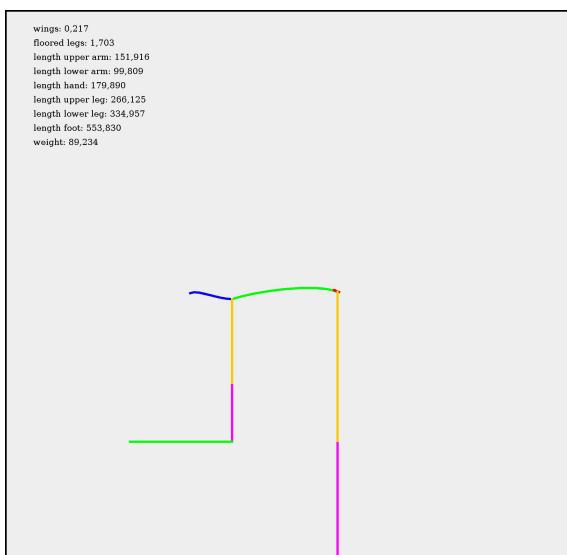
Dazu kann man, statt die ursprünglichen Daten und deren Verteilung zu verwenden, die entsprechenden bedingten Verteilungen bilden. Dazu muss der bedingte Mittelwert und die bedingte Kovarianzmatrix, wie in [6] (S. 116 f.) beschrieben, bestimmt werden.



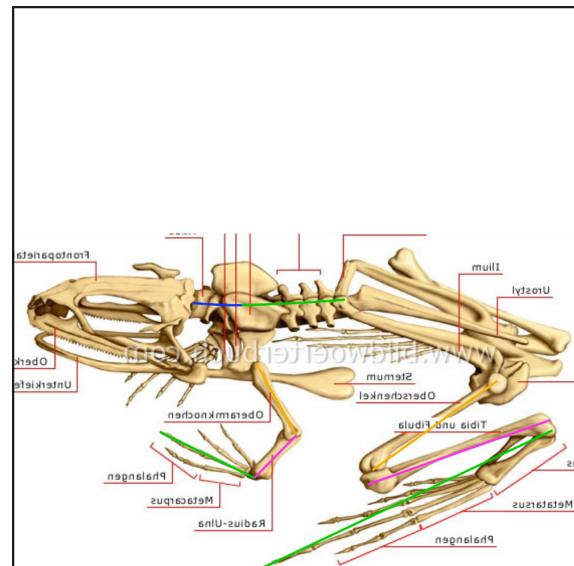
(a) 6 Eigenvektoren, Flügel 0, 4, Beine mit Bodenkontakt 1, 27, Gewicht 90, 2kg



(b) 10 Eigenvektoren, Flügel 0, 151, Beine mit Bodenkontakt 1, 28, Gewicht 88,7kg



(c) 20 Eigenvektoren, Flügel 0, 217, Beine mit Bodenkontakt 1, 7, Gewicht 89, 2kg



(d) Eingabebild

Abbildung 6.9.: Frosch, Rekonstruktionen aus den größten 6, 10 und 20 Eigenvektoren und das Eingabebild (d). Nicht visualisierte Dimensionen sind jeweils unter dem entsprechenden Bild angegeben. Zusätzlich zum Eingabebild erhobene Daten sind: *Tierklasse Amphibie, Flügel 0, Beine mit Bodenkontakt 2, Gewicht 0,01kg.*

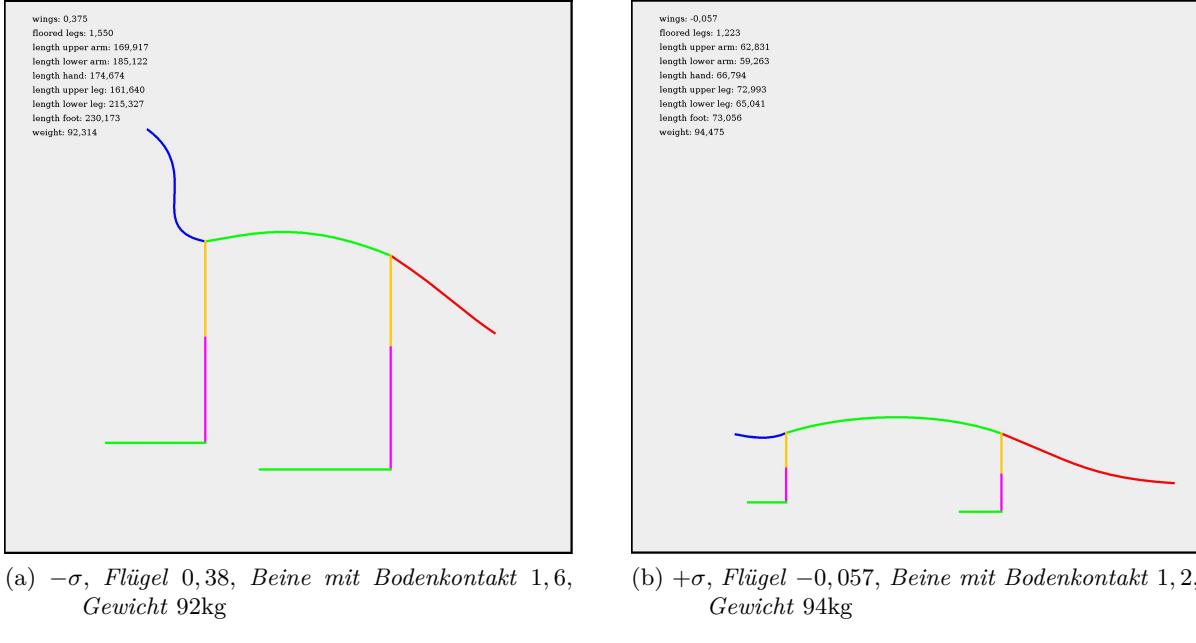


Abbildung 6.10.: Datenpunkte im PCA-Koordinatensystem. Die Koordinate für den größten Eigenvektor nimmt den Wert der positiven bzw. negativen Standardabweichung  $\sigma$  an, alle anderen sind null. Für Abbildungen zu den größten sechs Eigenvektoren siehe Abbildung C.5 im Anhang

Seien

$$x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}, \mu = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} \text{ und } \Sigma = \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix}$$

Der Zufallsvektor  $x$  enthält  $n$  normalverteilte Zufallsvariablen,  $q$  Variablen im Vektor  $x_1$  und  $n - q$  in  $x_2$ . Die Einträge in  $\mu$  sind die jeweils zugehörigen Mittelwerte und  $\Sigma$  ist die Kovarianzmatrix. Die Verteilung für  $x_1$  unter der Bedingung, dass  $x_2 = b$  hat den Mittelwert  $\bar{\mu}$  und Kovarianzmatrix  $\bar{\Sigma}$  mit

$$\bar{\mu} = \mu_1 + \Sigma_{12}\Sigma_{22}^{-1}(b - \mu_2), \bar{\Sigma} = \Sigma_{11} - \Sigma_{12}\Sigma_{12}^{-1}\Sigma_{21}$$

Verwendet man nun  $\bar{\Sigma}$  als Eingabe für die PCA, so erhält man nur noch Daten für Skelette unter den vorher festgelegten Bedingungen  $b$ . Zu beachten ist hier, dass die Eigenvektoren sich im Vergleich zur PCA ohne Bedingungen verändern. Falls man diese also zur Bestimmung eines Datenpunktes verwendet hat (z. B. in einer Benutzeroberfläche), so müssen die Werte für die neuen Eigenvektoren neu ausgerechnet werden.

Erzeugt man nun in diesem bedingten Raum zufällige Beispiele, zeigt sich sehr deutlich, dass das Festlegen von nur einer Dimension auch die anderen Dimensionen stark einschränken kann. Legt man z. B. die Anzahl der Beine fest, so haben die Skelette, die dann bedingt zufällig generiert werden, eine recht geringe Varianz. Das ist daran zu erkennen, dass der Verlauf der Wirbelsäule sehr ähnlich ist (siehe Abbildung 6.11a).

Um das etwas zu umgehen, wird die Eingabe als Intervall aufgefasst, aus dem zufällig ein Wert gezogen wird. Legt der Benutzer z. B. einen ganzzahligen Wert fest, so wird ein kleiner zufälliger Wert aufaddiert oder abgezogen. So wird nicht immer genau der gleiche Wert verwendet. Zwei Beispiele sind in den Abbildungen 6.11 b und c gezeigt.

Schaut man sich nun die erzeugten Wirbelsäulen mit einem Paar tragender Beine an (Abbildung 6.11c), so liegen sie alle recht nah an der mittleren Wirbelsäule (Abbildung

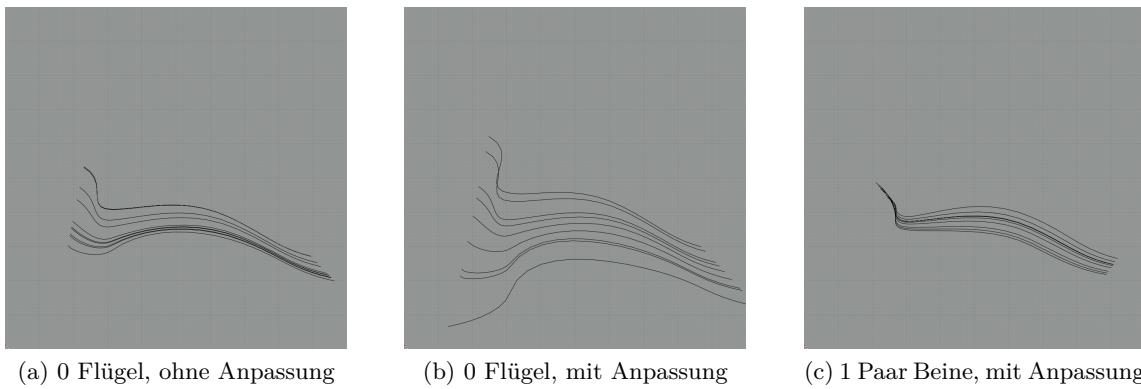


Abbildung 6.11.: Jeweils 10 bedingt zufällig generierte Wirbelsäulen. (a) und (b) ohne Flügel, (c) mit einem Paar Beine. In (b) und (c) wird ein zufälliger Wert aus  $[-0.5, 0.5]$  auf die Bedingung (Flügel = 0 bzw. Beine = 1) aufaddiert.

6.6). Das wirkt zunächst überraschend, da die meisten Tiere in den erhobenen Beispielen, die zwei Beine haben, Vögel sind. Sie haben eine Wirbelsäule, die hoch über dem Boden liegt und relativ aufrecht ist. Dann gibt es noch Känguru und Tyrannosaurus Rex. Bei ihnen liegt die Wirbelsäule ähnlich. Es muss aber auch einen Übergang von bodennahen Vierbeinern wie Krokodilen oder Fröschen über Zweibeiner zu Fischen ohne Beine geben. Nur gibt es dafür in der Natur eher wenige Beispiele. In den erhobenen Daten sind das nur der Seehund und die Ohrenrobbe.

Nun möchte man vielleicht Bedingungen an das zu generierende Skelett stellen, die nicht schon genau so in den erhobenen Daten repräsentiert sind. Bedingungen an Dimensionen, die gar nicht erhoben wurden, können entweder, unabhängig von der PCA, in den Ersetzungsregeln erzwungen werden, wie z. B. die Anzahl der Flossen, oder sie müssen in die Erhebung eingefügt werden.

Bedingungen, die schon implizit in den erhobenen Daten enthalten sind, sind beispielsweise die Schwanz- oder Halslänge. Um an sie Bedingungen stellen zu können, könnte man ebenfalls eine zusätzliche Dimension einfügen. Sie kann einfach aus den schon bestehenden Dimensionen errechnet werden. Dabei wäre aber das Problem, dass nur die Dimensionalität der Daten erhöht wird, nicht aber die eingegebenen Informationen. Es wären also mehr Eingabebeispiele nötig, nur dafür, dass die PCA eine offensichtliche Korrelation der Dimensionen erkennt.

Die bessere Alternative ist die Eingabedimensionen der PCA umzuparametrisieren. Die Länge des Schwanzes in x-Richtung, ist z. B. implizit in der Differenz der x-Koordinaten des ersten und letzten Kontrollpunktes der Bezièrkurve des Schwanzes enthalten. Ersetzt man nun den absoluten x-Wert des letzten Kontrollpunktes durch den Abstand in x-Richtung zum ersten Kontrollpunkt, so lässt sich die Schwanzlänge in x-Richtung ganz einfach als Bedingung an die PCA stellen.

Betrachtet man nun konkret den Wunsch die Schwanzlänge festzulegen, ist das nicht ganz so einfach. Die Längen in x- und y-Richtung lassen sich zwar leicht festlegen, diese Längen sagen aber im Allgemeinen noch nicht viel über die tatsächliche Länge der Bezièrkurve aus. Verlangt man z. B. eine Länge von 0px für den Schwanz in x-Richtung, so haben die generierten Datenpunkte einen Schwanz der zwar auf gleicher Höhe beginnt und endet, aber trotzdem vorhanden ist und einen kleinen Bogen beschreibt. Um die wirkliche Länge des Schwanzes zu messen, müsste man also noch mehr Aufwand in die Umparametrisierung stecken oder doch eine zusätzliche Dimension für die PCA in Kauf nehmen.

# 7. Erzeugung eines Skeletts

## 7.1. Überblick über den Ablauf der Generierung

**ToDo** (Koordinatensystem beschreiben, Achsen etc auch für später (zB Beinalgo) festlegen)  
**ToDo** („Kommunikation“ über SkeletonGenerator)

Generierung eines Skeletts

1. **PCA:** Lese die Beispieldaten ein und berechne den Basiswechsel (siehe Kapitel 6).
2. **Punkt im PCA Raum:** Bestimme einen (bedingt) zufälligen Punkt im PCA-Raum oder verwende einen vorgegebenen.
3. **Weitere Merkmale:** Bestimme aus den „Wahrscheinlichkeiten“ für *Flügel* und *Beine mit Bodenkontakt* Anzahl, Art und Ansatzpunkte der Extremitäten (Abschnitt 7.2) und lege fest an welchen Wirbeln Rippen ansetzen (Abschnitt 7.3).
4. **Ersetzungsregeln:** Starte mit dem Nichtterminal *Skelett* und wende so lange Ersetzungsregeln an bis nur noch Terminaler übrig sind (Abschnitt 5.2).
5. **Spiegelung:** Spiegle alle Elemente, die nicht auf der Wirbelsäule liegen. Das sind Extremitäten, Rippen und Schulterblatt. (Technische Details zur Spiegelung der Transformationsmatrizen sind in Abschnitt 9.5 zu finden.)
6. **3D-Modell:** Generiere aus den Abmessungen der terminalen Elemente 3D-Objekte und füge sie zu einem kompletten 3D-Modell zusammen. Dafür werden bestehende Modelle von echten Knochen oder einfache Quader eingesetzt (siehe Abschnitt 7.4).

- Bei echten Wirbeltieren gibt es sehr viele verschiedene Arten von Gelenken [11, Absatz 21.5.2]. Der Einfachheit halber werden hier nur Gelenke betrachtet, deren Winkel schon durch den Grundbauplan festgelegt sind (z.B. zwischen Wirbeln), und

Gelenke in Extremitäten mit einem Freiheitsgrad (z.B. an Ellenbogen oder Knie) (näheres zu Gelenken in Extremitäten weiter unten)

- Einheiten der PCA für Koordinaten [0, 1000], deshalb sind die Wirbelsäulen der generierten Skelette auch in diesem Rahmen. Blender interpretiert eine Einheit als 1m. Deshalb wirken sie sehr groß.
- Die Abmessungen der Knochen in die verschiedenen Richtungen ist bei den meisten Knochen relativ beliebig gewählt und oft auch immer gleich (außer bei Längen, die von PCA vorgegeben sind). Dafür gibt es keine biologische oder anatomische Grundlage. Man könnte hier sicherlich noch mehr machen (mehr Zufall, mehr anatomisch korrekt etc.) (**in Implementierungsdetails aufzählen was an Abmessungen alles beliebig (oder auch weniger beliebig) festgelegt wurde.**) ToDo

## 7.2. Extremitäten

Ein Punkt im PCA-Raum gibt schon viele Eigenschaften des zu generierenden Skeletts vor. Zu den Dingen, die noch festgelegt werden müssen, zählen z. B. die Anzahl und Anordnung der Wirbel und Rippen und vor allem die Ausrichtung der Extremitäten.

Die Bestandteile einer Extremität sind durch den Grundbauplan (Abbildung 2.2) vorgegeben und die Länge der Bestandteile durch die entsprechenden Merkmale des PCA-Datenpunkts. Die Schwierigkeit besteht nun darin in den entsprechenden Ersetzungsregeln (siehe Abbildung 5.1) auch die Positionierung vorzunehmen.

Das Skelett soll in einer Art Ruheposition dargestellt werden. Im Allgemeinen ist aber nicht klar wie die Ruheposition einer Extremität aussieht. Das ist schon allein daran zu erkennen, dass auf Darstellungen von Wirbeltierskeletten Flügel manchmal ausgestreckt und manchmal eingefaltet sind. Auch Beine sind meist so angeordnet, dass es aussieht als würde das entsprechende Tier gerade laufen. Dies ist auf Abbildung 6.7a am Beispiel des Klippschliefers sehr gut zu sehen.

Wie in Kapitel 6 zur PCA schon erwähnt, ist es deshalb auch schwer möglich die Ausrichtung der Extremitäten bzw. die Winkel an den Gelenken als zusätzliche Dimension in den PCA-Raum mitaufzunehmen.

Die Positionierung der Extremitäten bleibt also ein Problem mit unklaren Anforderungen und vielen Freiheitsgraden.

Ein erster Schritt an das Problem heranzugehen, ist es in kleinere Unterprobleme zu zerlegen. Extremitäten können anhand ihrer Funktion in vier Kategorien eingeteilt werden: Flügel, Flossen, Extremitäten mit Bodenkontakt (im Folgenden als Beine bezeichnet), und Extremitäten ohne Bodenkontakt (im Folgenden als Arme bezeichnet).

Für Flügel, Flossen und Arme gibt es keine besonderen Anforderungen außer, dass sie als solche zu erkennen sein sollten. Deshalb werden sie nach folgenden simplen Anweisungen orientiert: (**Beispielbilder?**) ToDo

- Flossen: Ausrichtung gerade nach hinten (orientiert an Welt-x-Achse)
- Arme: Der Oberarm zeigt senkrecht nach unten (orientiert an Welt-y-Achse), im Ellenbogengelenk ist ein 90° Winkel und die Hand verlängert Unterarm nach vorne.
- Flügel: Jedes beteiligte Gelenk hat ein Intervall mit festen Grenzen speziell für Flügel, aus dem zufällig ein Winkel gewählt wird.

Nun bleibt nur doch die Ausrichtung der Beine, für die die zusätzliche Anforderung gilt, dass sie den Boden berühren sollen.

### 7.2.1. Berechnung der Bodenhöhe

Zunächst könnte man davon ausgehen, dass der Boden einfach auf Höhe null sein sollte. Das Problem hierbei ist aber, dass die von der PCA berechneten Längen für die Extremitäten meist so kurz sind, dass die Beine dann den Boden nicht mehr erreichen würden. Das liegt daran, dass auf den Bildern, die als Eingabebeispiele für die PCA verwendet wurden, der Boden meistens nicht ganz am unteren Rand ist. Hier würde rigoroses Abschneiden der Bilder auf Fußhöhe wahrscheinlich helfen, es würde in vielen Fällen aber auch ein Großteil der Füße verloren gehen.

Die Höhe des Bodens wird also anhand der von der PCA generierten Längen der Extremitäten festgelegt. Theoretisch würde es reichen einfach das kürzeste Bein im komplett ausgestreckten Zustand zu betrachten und den Boden auf die Höhe dessen Endpunkts festzulegen. Das führt aber zu unnatürlich aussehenden Beinen, da das kürzeste Bein dann genau senkrecht nach unten führen muss um den Boden zu erreichen.

Deshalb wird nur ein bestimmter Anteil der Länge der Beine betrachtet. So wird erzwungen, dass die Beine, wenn sie auf dem neu definierten Boden stehen, auch etwas gekrümmmt sind. Dieser Anteil ist in der Implementierung mit 0.8 festgelegt, lässt sich aber natürlich auch variieren.

Wie in Kapitel 2 beschrieben gibt es verschiedene Arten von Füßen. Die oben beschriebene Berechnung geht davon aus, dass der Boden mit der Fußspitze berührt wird. Aber natürlich gibt es auch Wirbeltiere, die mit dem flachen Fuß auf der Erde stehen. Deshalb wird zusätzlich zur Bodenhöhe noch eine Wahrscheinlichkeit berechnet, dass Verse bzw. Handgelenk den Boden berühren.

Wird die Bodenhöhe nach oben verschoben, weil die Beine insgesamt zu kurz sind, ist diese Wahrscheinlichkeit null. Sind die Beine so lang, dass schon ohne die Bodenhöhe anzupassen die Verse bzw. das Handgelenk auf den Boden reicht, so ist sie eins. Ansonsten ist die Wahrscheinlichkeit

$$\frac{\text{Beinlänge} - \text{Höhe des Extremitätengürtels über 0}}{\text{Länge des Fußes}}$$

Wenn es Vorder- und Hinterbeine gibt, so wird für beide diese Wahrscheinlichkeit berechnet und dann der Mittelwert genommen. Denn es ist sinnvoll, dass alle Beine den gleichen Punkt auf den Boden bringen. Deshalb legt auch das erste Bein, das generiert wird, basierend auf der oben berechneten Wahrscheinlichkeit, fest, welcher Punkt gewählt wird.

**(Wann wird welcher Fuß verwendet?)**

**To Do**

### 7.2.2. Algorithmus zur Ausrichtung der Beine

Eine Herangehensweise dieses Problem zu lösen, wäre inverse Kinematik zu verwenden. Da das vorliegende Problem aber recht viele Randbedingungen hat, die ausgenutzt werden können, ist es gar nicht unbedingt nötig einen schwergewichtigen Algorithmus zu implementieren, der ein allgemeineres Problem löst.

Hier ist z. B. in den allermeisten Fällen klar in welche Richtung ein Gelenk gedreht werden muss um den Fuß dem Boden zu nähern oder ihn vom Boden zu entfernen. Außerdem ist gar kein spezieller Punkt auf dem Boden vorgegeben, der erreicht werden soll.

Deshalb wurde hier ein eigener, recht simpler Algorithmus entwickelt, der auf dieses spezielle Problem angepasst ist.

Die Gelenke in den Extremitäten werden hier so vereinfacht, dass genau einen Freiheitsgrad haben. Eltern- und Kindknochen sind also auf einer gemeinsamen Ebene festgelegt. Die

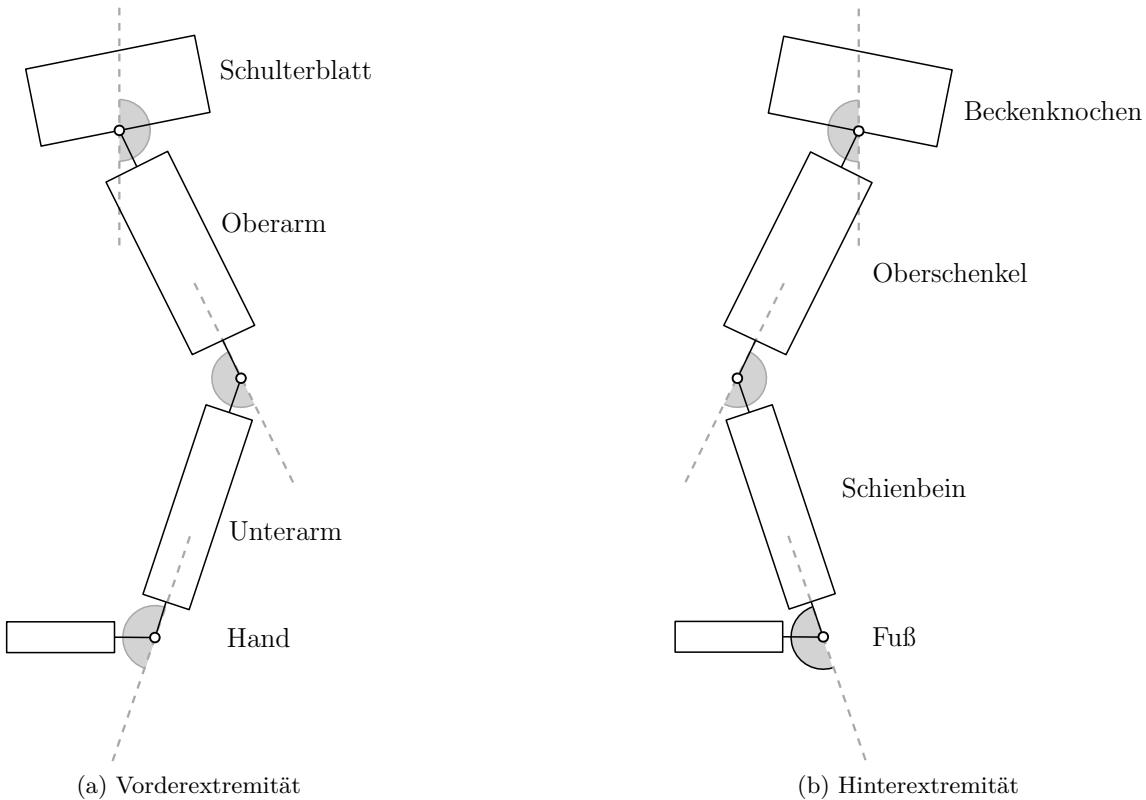


Abbildung 7.1.: Visualisierung der Bewegungsradien für die Gelenke in Extremitäten mit Bodenkontakt. Darstellung in Seitenansicht (wie in Abbildung 2.2). Die grauen Halbkreise zeigen die möglichen Winkel für eine Ruheposition des Skeletts. Unterarm, Unterschenkel und Fuß können jeweils maximal die Verlängerung des darüberliegenden Knochens bilden und minimal komplett an ihm anliegen. Oberarm und Oberschenkel können nicht über die Senkrechte hinaus gedreht werden.

verwendeten Gelenke mit ihren Einschränkungen sind in Abbildung 7.1 zu sehen. Der minimale und maximale Winkel für die Gelenke an Schulterblatt und Beckenknochen orientieren sich an der Welt-y-Achse, da außerhalb dieses Bereichs keine sinnvollen Ruhepositionen entstehen würden.

Theoretisch haben das Hüft- und das Schultergelenk nicht nur einen, sondern zwei Freiheitsgrade. Der Oberschenkel lässt sich nicht nur nach vorne und hinten bewegen, sondern auch seitlich abspreizen. Das lässt sich auch leicht als zweite Art von Gelenk im Code abbilden. Allerdings liefert der Algorithmus dann oft seltsam anmutende breitbeinige Tiere. Deshalb wurde der zweite Freiheitsgrad hier außen vor gelassen. Obwohl es natürlich in der Natur auch viele Tiere mit nach außen gestellten Beinen gibt, wie z. B. Echsen.

### (Pseudocode?)

**To Do**

Der Algorithmus geht iterativ vor. In jedem Schritt wird für jedes Gelenk berechnet ob sein Winkel vergrößert oder verkleinert werden muss um den dazugehörigen Knochen näher zum Boden zu bewegen. Mit dem „dazugehörigen“ Knochen ist hier derjenige der beiden an das Gelenk anschließenden Knochen gemeint, der das Kindelement des anderen ist. Die Drehrichtung lässt sich relativ leicht herausfinden indem die Ausrichtung des Knochens mit der Welt-y-Achse verglichen wird. Je senkrechter der Knochen ausgerichtet ist, desto ausgestreckter ist das Bein. Es gibt also globale Randbedingungen und lokale Einschränkungen je nach Gelenk.

Die Startposition der Extremität ist maximal angewinkelt. Die Gelenke beginnen also

**ToDo**

mit ihren kleinst- bzw. größtmöglichen Winkeln. In den folgenden Iterationen wird dann derjenige Endpunkt der Extremität dem Boden genähert, der zum Schluss Bodenkontakt haben soll. (**würde wahrscheinlich auch mit ausgestrecktem Bein als Startpos gehen; was wäre dabei zu beachten**) Ohne weitere Einschränkungen kann es nun passieren, dass unnatürliche Positionen auftreten, in denen z. B. der Fußspann näher am Boden ist als die Fußsohle. Oder es kann passieren, dass ein Knochen über die positive Welt-y-Achse hinaus gedreht wird. Das Problem dabei ist, dass die Einschränkungen an den Gelenken nicht zulassen, dass der Knochen sich unbegrenzt in diese Richtung weiterdreht und der Knochen dann „feststeckt“. Deshalb wird nach jeder Drehung festgestellt ob so eine Situation eingetreten ist und wenn ja, wird die Drehung rückgängig gemacht. Die Drehung wird ebenfalls rückgängig gemacht, falls sie bewirkt, dass die Knochen unterhalb der Bodenhöhe landen. (**Beispielbilder**) So ist zu jeder Zeit garantiert, dass die Knochen auf der „richtigen“ Seite der y-Achse liegen und nicht unterhalb der Bodenhöhe sind.

**ToDo**

In jeder Iteration werden die Winkel, um den die Gelenke gedreht werden, um einen bestimmten Anteil verkleinert. Zu Beginn soll, mit großen Veränderungen, eine grobe Ausrichtung der Gelenke vorgenommen werden, die dann immer weiter verfeinert wird. Der Startwinkel darf nicht zu klein sein, weil die Gelenke sonst ihre Zielpositionen nicht erreichen können. Ist der Startwinkel allerdings zu groß, bewirkt es in vielen Fällen nur, dass die Drehung nicht durchgeführt werden kann, weil die oben genannten Randbedingungen verletzt werden.

Die Verkleinerung des Winkels darf nicht zu schnell geschehen, weil dann wiederum die Endposition nicht erreicht werden kann. Wenn sie aber zu langsam geschieht passiert in vielen Schritten wiederum nichts wegen verletzter Randbedingungen.

Durch Ausprobieren wurden folgende Zahlen als sinnvoll erachtet: Startwinkel  $40^\circ$ , später dann jeweils  $\frac{6}{7}$  davon.

Falls sich der Abstand zum Boden kaum verändert, liegt also die Vermutung nahe, dass die Gradzahl zu groß ist und deshalb alle möglichen Winkeländerungen invalide sind. Deshalb wird in diesem Fall die Gradzahl für die nächste Iteration stärker verkleinert (halbiert).

**ToDo**

### (Wkten für Gelenke, Vorteil?)

Treten sehr kurzen Beinen auf, hat der Algorithmus außerdem einige kleine Probleme. Diese werden genauer in Abschnitt 9.8 beschrieben. Da die Beine aber in diesen Fällen, wie gesagt, sehr kurz sind, ist es für den Gesamteindruck gar nicht besonders wichtig wie genau sie angeordnet sind.

**ToDo**

Wie in Kapitel 8 beschrieben, lassen sich auch die Eingabebeispiele der PCA laden. Bei ihnen sind dann alle Attribute, die die PCA liefert, schon festgelegt. Alle anderen müssen jedoch noch generiert werden. Dazu gehören auch die Beine. Vergleicht man nun die Beinstellung, die der oben beschriebene Algorithmus generiert, mit der Beinstellung auf dem Eingabebild, lassen sich teilweise sehr große Unterschiede feststellen. (**gutes und schlechtes Beispiel (schlecht: Brachiosaurus, Elefant)**)

Um in allen Fällen eine realistisch wirkende Positionierung der Beine zu bekommen, müsste noch sehr viel mehr Arbeit in den Algorithmus gesteckt werden. Außerdem bräuchte der Algorithmus mehr Informationen zum Tier. Solche Zusatzinformationen könnten beispielsweise die Art des Fußes oder die Fortbewegungsart sein. Auch könnte es helfen, wenn es eine sinnvolle Möglichkeit gäbe die Winkel an den Gelenken als Dimension für die PCA mitaufzunehmen. Dafür müsste man sich aber, wie zu Beginn des Kapitels schon erwähnt, auf eine kanonische Ruheposition einigen und dann auch noch Beispiele in genau dieser Position finden.

Ein fertig generiertes Skelett wird höchstwahrscheinlich auch noch „weiterverarbeitet“. Soll z. B. ein animiertes Tier daraus werden, so müssen Bewegungszyklen geschaffen werden.

Dafür muss jedes Gelenk vielfach bewegt werden. Soll ein Tier mit Haut und Muskeln daraus werden, so müssen Muskeln an den Knochen ansetzen, die dann einen nicht unerheblichen Anteil an der Positionierung der Beine haben.

Die von dem hier beschriebenen Algorithmus generierte Position kann also gut als erster Eindruck dienen muss aber in den meisten Fällen noch angepasst werden. Ausgehend von der gegebenen Datenlagen und von den zu erwartenden Anwendungen ist es aber nicht sinnvoll den Algorithmus weiter zu verfeinern.

### 7.2.3. Zusätzliche Ansatzpunkte für Extremitäten

Ansatzpunkte für Extremitäten sind natürlich zunächst der Hüftgürtel und der Schultergürtel. Um auch die Generierung fantastischer Tiere (siehe Abschnitt 2.3) zu ermöglichen, ist es aber möglich dies zu erweitern.

Eine einfache Möglichkeit ist hier zunächst die Anzahl der möglichen Extremitätenpaare von zwei auf vier zu erhöhen, indem einfach an der Hüfte und der Schulter jeweils zwei Paare ansetzen dürfen. Dafür wurden an der Hüfte bzw. der Schulter mehrere Gelenke direkt hintereinander angelegt, an denen Extremitäten ansetzen können. Das ermöglicht beispielsweise Tiere wie den Pegasus.

Flügel und Arme dürfen hierbei weiterhin nur an der Schulter ansetzen, Beine und Flossen an beiden Stellen. Der Grund dafür ist, dass die meisten generierten Skelette seltsam wirken, wenn an der Hüfte Flügel oder Arme ansetzen und dafür an der Schulter Beine beginnen. Das liegt daran, dass existierende Tiere mit Flügeln oder Armen ihren Schwerpunkt im hinteren Bereich haben und sie auf den Hinterbeinen stehen. Deshalb wird die Wirbelsäule durch die PCA auch dementsprechend angelegt.

Eine Überlegung war auch zwischen Schulter und Hüfte weitere Extremitätengürtel anzu bringen, um z. B. Tiere wie asiatische Drachen erzeugen zu können. Das stellt sich aber als schwierig heraus. Die Wirbelsäule ist zwischen Hüfte und Schulter meist nach oben geschwungen und im Bauchraum befinden sich die meisten Organe des Tieres. Ein zusätzlicher Extremitätengürtel würde den Bauchraum einschränken. Außerdem wirkt dann auch die nach oben geschwungene Wirbelsäule anatomisch seltsam. „Verdoppelt“ man die Schwingung der Wirbelsäule und hängt einfach einen weiteren Rücken hinten oder vorne an, so wirkt es ebenso seltsam, da dann die „Höcker“ der Wirbelsäule für das Tier wahrscheinlich nicht wirklich ein Vorteil sind und nur die Fortbewegung erschweren. Asiatische Drachen sind ein „Spezialfall“. Sie besitzen langen, schlangenartigen Körper, bei dem die Wirbelsäule gerade verläuft. Deshalb treten die oben genannten Probleme bei ihnen nicht auf.

Eine weitere Idee, die auch umgesetzt wurde, ist, eine Art Zentauren zu ermöglichen. Hat das Tier einen Hals, der lang genug ist, kann darauf ein weiterer Schultergürtel kurz unterhalb vom Kopf angebracht werden. An diesem Schultergürtel dürfen dann alle Arten von Extremitäten außer Beinen ansetzen. Das wirkt tatsächlich meist auch anatomisch einigermaßen sinnvoll.

Ein solcher Schultergürtel wirkt aber nur sinnvoll, wenn auch der Hals lang genug ist. Deshalb wird die Benutzereingabe, die einen zweiten Schultergürtel erzwingt in eine Bedingung für die Halslänge umgewandelt und an die PCA weitergegeben. Gibt es keine Benutzereingabe, so wird ein zweiter Schultergürtel nur generiert, wenn der Hals lang genug ist.

### 7.2.4. Anordnung und Anzahl der Extremitäten

Ein PCA-Datenpunkt stellt die Länge der Vorder- und Hinterbeine, die Wahrscheinlichkeit für Flügel und die „Wahrscheinlichkeit“ für Beine zur Verfügung. Die genaue Anzahl (auch für fantastische Tiere) und die Anordnung müssen noch festgelegt werden.

Die Position jeder Extremität wird zufällig aus der Menge der möglichen Positionen gewählt. Beine dürfen nicht am zusätzlichen Schultergürtel am Hals generiert werden, Arme und Flügel dürfen nicht an der Hüfte generiert werden und Flossen sind überall erlaubt. Ist für eine Extremität kein Platz mehr, wird zunächst getestet, ob eine der anderen Extremitäten an ihre Position wechseln kann um Platz zu schaffen. Erst wenn das fehlschlägt kann die betreffende Extremität nicht platziert werden.

Da die Positionen nicht deterministisch gewählt werden, kommt es bei Tieren mit mehreren möglichen Anordnungen zu unterschiedlichen Ergebnissen bei gleicher Eingabe.

Die Anzahl und Art der Extremitäten orientiert sich zunächst an Benutzereingabe. Gegebenenfalls wurden diese Eingaben auch schon als Bedingungen an die PCA weitergeleitet (siehe Abschnitt 6.4).

Falls es keine Benutzereingabe für eine Art von Extremitäten gibt, so wird zunächst getestet ob es noch Platz gibt für diese Art. Falls noch Platz vorhanden ist, wird je nach Art unterschiedlich vorgegangen: Die Anzahl der Beine oder Flügel wird anhand der Wahrscheinlichkeiten des PCA-Datenpunktes generiert, aber maximal ein Paar pro Schulter- und Hüftgürtel. Die „Wahrscheinlichkeit“ der Beine liegt im Intervall [0, 2]. Hier wird, je nach dem ob die „Wahrscheinlichkeit“ in [0, 1] oder in [1, 2] liegt, ausgelost ob null oder ein bzw. ein oder zwei Beinpaare generiert werden.

Für Arme und Flossen gibt es keine Anhaltspunkte durch die PCA. Deshalb wurde einfach festgelegt, dass Arme, wie Flügel, mit der PCA-Flügelwahrscheinlichkeit generiert werden. Beine werden ganz zum Schluss in noch komplett leeren Extremitätengürteln generiert, falls die Länge der dort zu generierenden Extremitäten nicht zu lang ist.

## ToDo

**(nachträgliche Umverteilung der Extremitäten sinnvoll? (leere Gürtel auffüllen))**

Dieses Vorgehen führt dazu, dass, wenn ein PCA-Beispiel geladen wird, die Anzahl, Art und Position der Extremitäten nicht unbedingt dem realen Tier entsprechen muss. Zunächst kann es einen zusätzlichen Schultergürtel geben, falls der Nutzer das erlaubt und der Hals lang genug ist, und es können mehrere Extremitäten pro Extremitätengürtel platziert werden, wenn das erlaubt ist.

Die Anzahl der Flügel und Beine wird stimmen, da ihre Wahrscheinlichkeiten ja festgelegt sind, aber ihre Position wird zufällig bestimmt, falls möglich. Arme und Flossen werden nach den oben genannten Kriterien generiert, falls der Nutzer nichts anderes vorgibt.

Wenn ein Tier also möglichst genau reproduziert werden soll, sollte zumindest die Anzahl der Arme und Flossen angegeben werden und ein zweiter Schultergürtel und mehrere Extremitäten pro Gürtel verboten werden.

## 7.3. Wirbel und Rippen

Weitere Dinge, die festgelegt werden müssen, ist die Anzahl der Wirbel auf den einzelnen Teilen der Wirbelsäule und die Anzahl der Rippen.

Die Anzahl der Wirbel orientiert sich an echten Wirbeltierskeletten (siehe Absatz 2.1). Auf dem Hals werden 7 Wirbel generiert, außer das Tier hat Flügel, dann liegt die Anzahl zwischen 10 und 30. Auf dem Rücken liegen 25 Wirbel und auf dem Schwanz 5 bis 20. Da der „Wurzelwirbel“ in der Mitte der Rückenwirbelsäule liegt (oder zumindest ungefähr, da Bezierkurve bei 0,5 ausgewertet bei wird) werden 13 Wirbel auf der vorderen Hälfte und 12 hinten generiert.

Eine Bézierkurve ist ein Weg im  $\mathbb{R}^2$ . Sie ist parametrisiert auf [0, 1] und im Allgemeinen nicht nach Bogenlänge parametrisiert. Das heißt die Geschwindigkeit, mit der die Kurve durchlaufen wird, ist nicht konstant. Wertet man die Kurve also bei 0.5 aus, wurde nicht

notwendigerweise die Hälfte der Strecke zurückgelegt.

Für die Generierung von  $n$  Wirbeln wurde die Kurve einfach an den Stellen  $\frac{i}{n}$  ausgewertet, für  $0 \leq i \leq n$ . Durch den oben genannten Effekt variiert dann die Länge der Wirbel über den Kurvenverlauf.

Wird die Bézierkurve  $B$  nach Bogenlänge umparametrisieren, so erhält man die Kurve  $S$  mit:

$$S(t) = \int_0^t \|B'(s)\| \, ds$$

Das obenstehende Integral ist jedoch schwer zu berechnen, da im Allgemeinen keine Stammfunktion des Integranden zur Verfügung steht. Es gibt jedoch numerische Methoden, mit denen das Problem gelöst werden kann. [29]

Betrachtet man echte Wirbeltiere, so sind keine einfachen Regeln für die Länge ihrer Wirbel ersichtlich. Es gibt beispielsweise eine Studie, die die Beschaffenheit der Wirbel von Mäusen untersucht [23]. In dieser Studie auf Abbildung 5, Seite 19, ist sehr gut zu sehen, dass die Länge der Wirbel im Verlauf der Wirbelsäule sehr stark schwankt.

Die Anzahl der Rippen und auch die Ausdehnung des Brustkorbs variiert zwischen Wirbeltieren sehr stark. Es gibt Tiere, die an jedem Wirbel der Rückenwirbelsäule Rippen haben und es gibt Tiere die haben nur ein paar wenige auf dem vorderen Teil. (**Beispiele**)  
Deshalb wird ein zufälliges Intervall  $[0, x]$  auf der Bézierkurve des Rückens bestimmt. Jeder Wirbel, der in diesem Intervall liegt, bekommt auch eine Rippe. (**Bestimmung der Rippenlänge beschreiben?**)

ToDo  
ToDo

## 7.4. Knochenmodelle

Zunächst wird jeder terminale Knochen durch seine Bounding Box dargestellt. (**es sind nicht wirklich Bounding Boxen, eher „Proxyboxen“**) Diese Boxen lassen sich aber leicht durch die 3D-Modelle der entsprechenden Knochen ersetzen. Dazu müssen die 3D-Modelle nur im .obj-Format vorliegen und folgenden Bedingungen entsprechen:

ToDo

Das Modell ist korrekt an den Achsen ausgerichtet und so verzerrt, dass es einen Würfel mit 1(m) Kantenlänge in jeder Richtung möglichst gut ausfüllt.

Lässt man es hierbei bewenden, so ist es relativ schwierig herauszufinden wie man die einzelnen Knochen skalieren muss, dass sie an den Gelenken gut zusammenpassen. Außerdem ist es aufwändig herauszufinden wo die Gelenke an den Knochen ansetzen.

Setzt man sich dagegen etwas über den Gedanken der „Bounding Box“ hinweg, so kann die Positionierung und Skalierung einfacher werden. Hier wurden, je nach Knochen, einige der folgenden Punkte umgesetzt. (**Beispielbilder**)

ToDo

- Kleine Fortsätze, die nicht wirklich zur (optischen) Größe des Knochens beitragen, z. B. die Fortsätze der Wirbel, ragen aus der Bounding Box heraus.
- Kantenlängen, die von „außen“ vorgegeben werden, sind genau auf die Kantenlänge der Box skaliert (also 1). So beispielsweise die x-Länge der Wirbel, die auf der Wirbelsäule genau aneinander stoßen sollen. Dies können aber auch Längen sein, nur einen Teil des Knochens betreffen. Der Beinabstand an der Hüfte ist z. B. kleiner als die komplette Breite der Hüfte. Es ist aber einfacher den Beinabstand anzugeben, als die Hüftbreite. Auch die Skalierung der Hüfte in x-Richtung ist zunächst nicht klar, aber wenn die Breite der zugehörigen Wirbel gegeben ist, ist auch klar, wie breit die Hüfte sein soll. Deshalb ist die Hüfte in x-Richtung so skaliert, dass der Teil, an

dem der Wirbel ansetzt, schon die komplette Kantenlänge des Würfels ausfüllt. Bei Gelenken, die von der Breite her zusammenpassen sollen, ist dies auch sehr hilfreich. Aber das führt natürlich auch dazu, dass die „Bounding Box“ nicht mehr viel mit der resultierenden Größe des Knochens zu tun haben muss.

- Kantenlängen, die nicht vorgegeben werden, sind einfacher passend zu bestimmen, wenn sie nicht komplett unabhängig von den anderen Raumrichtungen sind. Ist z. B. die x- und y-Skalierung eines Knochens vorgegeben, und die Skalierung in z-Richtung soll nur möglichst gut dazu passen, so ist es sinnvoll, das 3D-Modell schon so zu speichern, dass die z-Richtung von einer der anderen Richtungen abhängt. Tut man dies nicht, so führt das relativ leicht dazu, dass die Knochen grundlos verzerrt werden.
- Längen die variiert werden können sollen z. B. die Länge der Wirbel oder der Beinabstand der Hüfte sollten genau eine Kantenlänge der 1x1 Box einnehmen.

## ToDo

### (Schwierigkeit bei stark gebogenen Knochen)

Liegen die Modelle in diesem Format vor, können sie einfach eingelesen werden und anhand der Skalierung der Bounding Box skaliert werden. Hier wurden vor allem Modelle von menschlichen Knochen verwendet, da sie leichter verfügbar sind. Manche Knochen sind jedoch auch von anderen Tieren. Das führt z. B. bei dem verwendeten Unterarmknochen des Pferdes dazu, dass er etwas überdimensionierte Fortsätze am Ellenbogen bekommen, wenn man ihn großskaliert. Das liegt daran, dass dieser Knochen beim Pferd eigentlich relativ kurz ist.

Eine Schwierigkeit daran Modelle in der oben genannten Form herzustellen ist, dass nicht unbedingt sofort klar ist, wie die Knochen ausgerichtet werden müssen. Die Hüfte muss z. B. so ausgerichtet werden, dass der Anfangs- und Endpunkt der durchgehenden Wirbelsäule auf gleicher Höhe liegen, damit nachfolgende Wirbel auch richtig anschließen. (Das funktioniert natürlich nur, weil die Wirbelsäule an der Stelle der Hüfte quasi gerade ist.)

## ToDo

### (wg Lizenz Pferdehüfte verwendet, die nicht mit Wirbelsäule verwachsen ist)

Auch die Positionierung der Rippen und des Oberarms in Kombination mit dem Unterarm ist anspruchsvoll. Dabei hilft es 3D-Modelle zu haben, in denen die anderen Knochen auch schon vorhanden sind, um sich die Ausrichtung abzuschauen. Außerdem können Bilder von Skeletten zu Rate gezogen werden. Und zuletzt muss man die genaue Positionierung einfach testen.

Zusätzlich muss beachtet werden wie die Knochen aneinander anschließen bzw. wie sie für die entsprechenden Gelenke korrekt positioniert sind. Das erfordert etwas „finetuning“. Für jeden Knochen sind dafür in Abhängigkeit zur Bounding Box zwei Offsets gespeichert: das Offset zu dem Gelenk, das ihn mit seinem Elternknochen verbindet und das Offset zu dem Gelenk, das ihn mit seinem Kindknochen verbindet (oder mehrere, falls vorhanden). Dies sorgt dafür, dass die Positionierung der Knochen stimmt, egal wie groß sie sind. Ist ein Knochen sehr groß und ein anschließender sehr klein (oder anders herum), so kommt es natürlich trotzdem vor, dass die Gelenke nicht wirklich ineinander passen. Für solche Situationen bräuchte man verschiedene 3D-Modelle, die je nach Gegebenheit eingesetzt werden.

Da der Kopf bzw. der Schädelknochen im Gegensatz zu anderen Knochen bei Wirbeltieren sehr stark variiert, ist es sinnvoll mehrere Schädelknochen zur Auswahl zu haben. Geht die Menge an verfügbaren Schädelknochen über „Tier mit Flügeln“ und „Tier ohne Flügel“ hinaus, so ist es außerdem sinnvoll die Auswahl des passenden Schädelknochens dem Nutzer zu überlassen.

Bei Händen und Füßen ist das Problem ebenso, dass es sehr viele verschiedene Ausprägungen davon gibt. Hier lassen sie sich jedoch grob nach Extremitätentyp unterscheiden. Diese

**ToDo**

Unterscheidung kann aber beliebig fein sein. Hier wurde nur nach Flügel, Flosse (?) und Extremität mit Bodenkontakt unterschieden. Und bei Extremitäten mit Bodenkontakt wurde nochmals danach unterschieden wie flach der Fuß oder die Hand auf dem Boden aufkommt (bei weniger als 45° ist es eine menschliche Hand, sonst ein Huf). (ggf. an **Implementierung anpassen**) (erwähnen, dass Knochen für Flügel, „hand“ nicht vollständig)

(Schwanzspitze: ein Modell für drei Wirbel. Das sieht bei wenigen Wirbeln komisch abgeknickt aus)

Setzten mehrere Extremitäten an einer Schulter oder einer Hüfte an, so ist ein „normales“ Modell der Schulter/Hüfte nicht mehr ausreichend, da nicht genug Gelenke vorhanden sind. Dieses Problem wurde an der Schulter so behoben, dass einfach mehrere Schulterblätte mit einem gewissen Abstand generiert wurden. Für die Hüfte wurde ein kombiniertes 3D-Modell aus zwei einzelnen Hüften erstellt, an dem nun zwei Gelenke vorhanden sind.

(3D-Modelle leider nicht ganz einfach Austauschbar wegen Offsets -> erklären wie Offsets eingelesen werden müssten) (Wenn Muskeln generiert werden sollen, müssen noch mehr Ansatzpunkte bestimmt werden und Positionierung der Knochen hängt von Muskeln ab -> future work)

**ToDo**  
**ToDo**

## 8. Zusätzliche Features

### 8.1. Speichern und Laden von Skeletten

Um ein Skelett zu speichern reicht es jene Metadaten herauszuschreiben, aus denen der Algorithmus das Skelett wieder reproduzieren kann. Zu den benötigten Daten gehören die Daten, die die PCA liefert (Position der Wirbelsäule, Länge der Knochen in den Extremitäten, Gewicht) und Daten, die durch den Algorithmus generiert werden (Anzahl der Wirbel, Intervall auf der Wirbelsäule, in dem sich Rippen befinden, Anzahl und Art der Extremitäten an den jeweiligen Extremitätengürteln und die Winkel an Gelenken der Extremitäten, Art des einzusetzenden Kopfes). Das alles wird in ein paar wenigen Java-Klassen gebündelt und über Java-Serialisierung in eine Textdatei geschrieben. Diese Datei kann dann wieder eingelesen werden um die Klassen wieder herzustellen. (**PCA Beispiele können auch geladen werden**)

**ToDo**

Das funktioniert natürlich nur über das implementierte Programm und liefert keine Metainformationen zu dem generierten Skelett nach außen. (**Ideen für Zusatzinfos, die sinnvoll sein könnten, aufschreiben / future work (zB Knochenhierarchie); wären aber wsh nicht super hilfreich, weil für Animation etc. sowieso nochmal alles angefasst werden muss**)

**ToDo**

(**Möglichkeiten Zusatzinfos abzuspeichern (fbx, alembic, universal scene description <https://graphics.pixar.com/usd/docs/index.html>), python script für Blender etc. um Knochen anzulegen**)

### 8.2. Erzeugung von Variationen

- was genau wird variiert
- PCA Daten normalverteilt variieren (Verteilung bleibt Gauß, Erwartungswert bleibt gleich, Überlegungen dazu wie sie sich ändert durch Aufaddieren der Variation / Faltung mit anderem Gauß)
- Variation der Daten, die nicht von PCA abhängen, beschreiben; generische Algorithmen zitieren (auch hinzufügen und löschen oder ändern von Features um Variationen/Verbesserungen zu erzeugen)
- create variations on existing models: simmons wilhelms van gelder 2002 [https://www.researchgate.net/publication/234776823\\_Model-based\\_reconstruction\\_for\\_creature\\_animation](https://www.researchgate.net/publication/234776823_Model-based_reconstruction_for_creature_animation)

## 8.3. GUI

(**Funktionen beschreiben + wie verknüpft mit Programm**) JavaView zum anzeigen für groben Überblick (Lizenz checken) **ToDo**

Interaktivität

- Eine Anwendung, bei der nach Eingabe von Parametern sofort das komplette Tier generiert wird, ist weniger hilfreich als eine, bei der schrittweise Teile davon generiert werden können (und auch rückgängig gemacht werden können)
- Teile, die einem nicht gefallen, sollten geändert werden können
- Änderungen können Auswirkungen auf den Rest des Körpers haben (durch Regeln) bzw. manche Änderungen sind nicht möglich
- Könnte verwendet werden um schnell verschiedene Möglichkeiten zu testen

# 9. Implementierungsdetails

## 9.1. Programmiersprache

- Rust: nicht geeignet, da Datenstrukturen die zyklische Referenzen auf veränderbare Objekte verwenden nicht oder nur kompliziert umsetzbar sind.
- Java: scheint gut zu funktionieren. Es gibt Bibliotheken zum im-/exportieren von obj-Dateien und Unterstützung für OpenGL  
Außerdem gibt es Bibliotheken, die PCA (bzw. Kovarianzmatrizen etc.) unterstützen und JavaView um objs darzustellen

## 9.2. Dateiformate

- Einfachstes Format (nur für die Darstellung von 3D-Objekten ohne Zusatzinformationen): obj
- Erster Schritt: einfaches .obj erzeugen und mit Blender darstellen; einfach Knochen als Bounding Box darstellen, später Verwendung von OpenGL mit vertex shadern etc. (Plan erwähnen? wurde ja doch nicht gemacht...)
- Jeder Editor geht mit Muskeln und Gelenken anders um. Gibt es ein Dateiformat, das nicht speziell zu einem Editor gehört, dass Bedingungen an die Rotation von Gelenken speichern kann?
- Eigenes Format erzeugen? Dann bräuchte man Plugins um es in verschiedenen Editoren laden zu können. Viel verwendeter Editor: Houdini (kostenlos für Studenten aber nicht Open Source). Oder selbst darstellen (siehe Interaktivität).
- Vorschlag von Jo: „Memory dumps“, also direkt die structs aus dem Speicher auf Platte rauschreiben. Am besten wenn sie am Stück liegen mit einem fwrite() und zurücklesen mit einem fread(). Es ist nützlich dazu am Anfang der Datei ein bisschen Metadaten zu speichern (magic number, version, array size etc.).

### 9.2.1. OpenSim

- <https://simtk-confluence.stanford.edu:8443/display/OpenSim/OpenSim+Documentation>
- Open Source Software Platform für die Modellierung und Simulation von Menschen, Tieren, etc.  
aber vor allem gedacht zur Auswertung von experimentellen Daten

- Import von .obj Dateien möglich. Außerdem zusätzliche Daten wie Winkel von Gelenken über .mot oder .sto Dateien (eigenes Format von OpenSim, siehe <https://simtk-confluence.stanford.edu:8443/display/OpenSim/Preparing+Your+Data>)
- Export in andere Dateiformate nicht möglich (?)
- für Download und Zugang zur „Community“ Account nötig
- für Windows und Mac OS (Linux Support gibt es auch, ist aber schwieriger: <https://simtk-confluence.stanford.edu:8443/display/OpenSim/Linux+Support>)

### 9.2.2. OBJ

- Beschreibung des Formats: <https://www.fileformat.info/format/wavefrontobj/egff.htm>
- Erzeugung mit Rust: obj\_exporter <https://docs.rs/obj-exporter/0.2.0/obj-exporter/index.html>
- Erzeugung mit Java: javagl Obj <https://github.com/javagl/Obj>, unterstützt auch Umwandlung von obj-Daten in Daten, die direkt für vertex buffer objects in OpenGL verwendet werden können
- Reicht wahrscheinlich für die ersten Dinge aus. Finetuning wird sowieso mit anderer Software gemacht

### 9.2.3. FBX

- Verwendung am besten über Autodesk FBX SDK für C++.
- Dokumentation: <http://help.autodesk.com/view/FBX/2019/ENU/> und <http://docs.autodesk.com/FBX/2014/ENU/FBX-SDK-Documentation/index.html>
- Es gibt auch fbxcel, eine FBX library für Rust. Ist aber relativ low level und nicht ganz offensichtlich wie zu verwenden.
- Einschränkungen für Gelenke können in FBX nicht gespeichert werden [http://docs.autodesk.com/FBX/2014/ENU/FBX-SDK-Documentation/index.html?url=cpp\\_ref/class\\_fbx\\_constraint.html&topicNumber=cpp\\_ref\\_class\\_fbx\\_constraint\\_htmlc57a3f99-513a-44a0-a24f-445e9077c99f](http://docs.autodesk.com/FBX/2014/ENU/FBX-SDK-Documentation/index.html?url=cpp_ref/class_fbx_constraint.html&topicNumber=cpp_ref_class_fbx_constraint_htmlc57a3f99-513a-44a0-a24f-445e9077c99f)

### 9.2.4. Alembic

- [www.alembic.io](http://www.alembic.io)
- Wird u.a. dafür verwendet Knochen (+ Animationen) in Ziva zu importieren
- Es kann mit Python (PyAlembic) und C++ verwendet werden.  
PyAlembic Doku: <http://docs.alembic.io/python/examples.html#pyalembic-intro>  
C++ API Refernce (enthält sehr wenig Infos): <http://docs.alembic.io/reference/index.html>
- Für Rust gibt es keine Bibliothek (?)

## 9.3. Aufbau der Software

- ein Wort zur Softwarearchitektur?
- Erweiterbarkeit der Software?

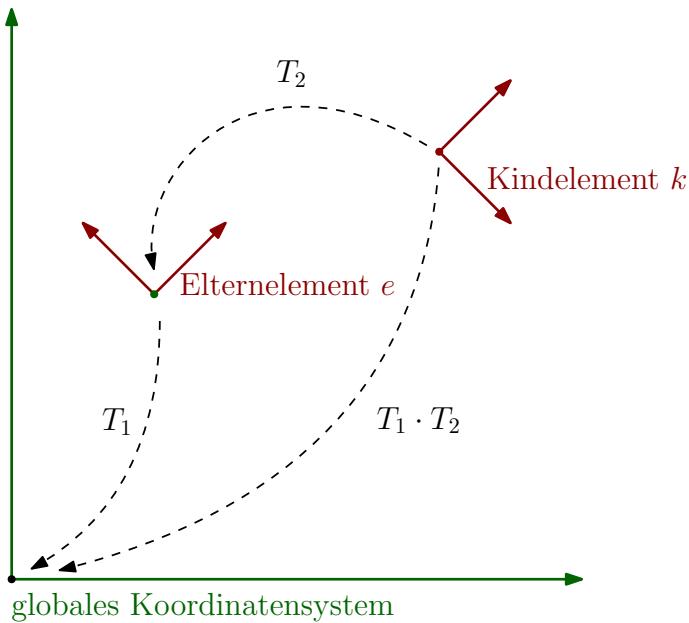


Abbildung 9.1.: Gegeben sei das Element  $e$ . Die Abbildung, die die lokalen Koordinaten von  $e$  in globale Koordinaten umrechnet sei  $T_1$ . Jedes Kinderelement  $k$  von  $e$  speichert eine Transformationsmatrix  $T_2$ , die angibt wo der Ursprung des Koordinatensystems von  $k$  relativ zum Koordinatensystem von  $e$  liegt. Will man nun Koordinaten von  $k$  in globale Koordinaten umrechnen, benötigt man die Abbildung  $T_1 \cdot T_2$ .

## 9.4. Zufall

Linearer Kongruenzgenerator reicht für Zufallszahlen aus, da nur wenige erzeugt werden (erkennbare Muster entstehen erst bei mehr Zufallszahlen) (**erwähnen?**)

**To Do**

## 9.5. Transformationsmatrizen

**To Do**

(**Recherchiere warum rechtshändige Koordinatensysteme bzw. konsistente Koordinatensysteme wichtig sind (back face culling, Rotationsrichtung der Matrizen bei positiven Winkeln)**)

Jedes Element im Skelett speichert, relativ zu seinem Elternelement, die Position des Ursprungs seines Koordinatensystems. Um den Überblick über die Transformationsmatrizen bzw. Abbildungen behalten, die vom einen ins andere Koordinatensystem umwandeln, hier zwei Übersichtsgrafiken: (**Erzeugung von Elementen auf der Wirbelsäule mit gegebener Weltposition + gespiegelte Elemente**)

## 9.6. PCA

Die Bilder der Skelette wurden folgendermaßen für die Datenerhebung vorbereitet:

1. Zuschneiden des Bildes, so dass möglichst nur das Skelett mit wenig Rand außen herum zu sehen ist.
2. Einfügen in eine  $1000 \times 1000$  Pixel große Bildumgebung.
3. Verschieben innerhalb der Bildumgebung an den unteren Rand und horizontal in die Mitte.

Ist das geschehen kann die Lage der Wirbelsäule und die Länge der Knochen der Extremitäten annotiert werden.

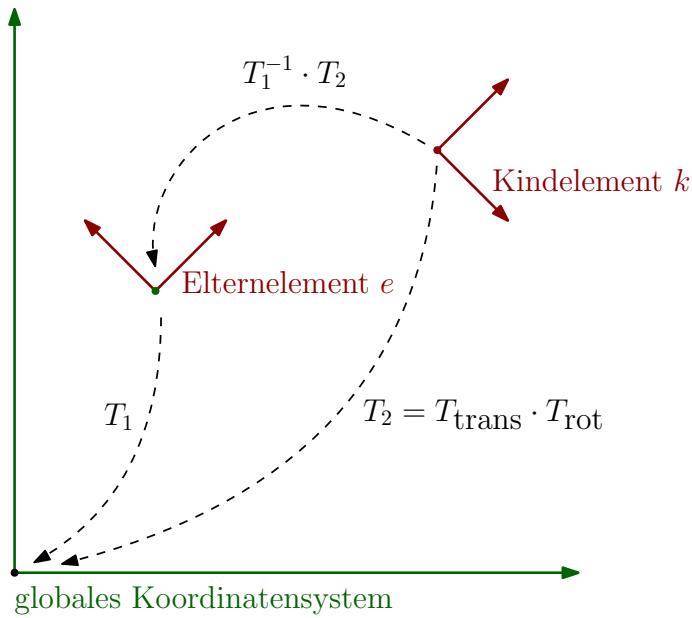


Abbildung 9.2.: Will man ein Element  $k$  erzeugen, das Kindelement von  $e$  ist und dessen globale Koordinaten bekannt sind, muss man die Abbildung berechnen, die die relative Position von  $k$  angibt. Seien  $T_1$  und  $T_2$  jeweils die Transformationen in das globale Koordinatensystem von  $e$  bzw.  $k$ . Dann ist die gesuchte Abbildung  $T_1^{-1} \cdot T_2$ .

### 9.6.1. Annotation der Bilder

Die Annotation der Bilder wurde per Hand mit dem Programm Inkscape<sup>1</sup> durchgeführt. Für jedes zu markierende Element wurde eine Strecke oder eine Bézierkurve eingefügt und mit einem vorher festgelegten Namen benannt. Diese Elemente wurden dann durch Inkscape als Pfade in der erzeugten svg-Datei gespeichert. Aus dieser Datei wurden dann automatisiert die eingetragenen Pfade mit ihren Koordinaten ausgelesen.

Folgende Details sind wichtig zu beachten, damit dieser Vorgang reibungslos abläuft.

- Man kann in Inkscape einstellen, dass Koordinaten immer absolut angegeben werden. Das ist sinnvoll um die Koordinaten leichter auslesen zu können.
- Der Ursprung des Koordinatensystems in Inkscape ist unten links, im svg-Format ist er aber oben links.
- Ebenen sollten in Inkscape nicht verschoben sein, sonst verschieben sich mit ihnen auch die Koordinaten.

### 9.6.2. Anpassung der PCA-Ergebnisse zur Weiterverarbeitung

- Um Wirbelsäule aus PCA Daten differenzierbar zu machen, wurden jeweils der vorletzte und der zweite Punkt von Hals und Rücken bzw. Rücken und Schwanz um den Kontaktspunkt um den gleichen Winkel in gegensätzliche Richtungen gedreht, damit beide Bézierkurven an dem Kontaktspunkt die gleiche Steigung haben. (**beste ToDo Lösung?**)
- An den Kontaktspunkten der Wirbelsäulenteile ist die Steigung nicht unbedingt gleich (die Wirbelsäule als ganzes ist nicht  $C^1$ ). Deshalb müssen vor der Weiterverwendung die jeweils nächsten Kontrollpunkte nach dem Kontaktspunkt verschoben werden. Das

<sup>1</sup>Programm zum erstellen und bearbeiten von Vektorgrafiken, <https://inkscape.org/de>

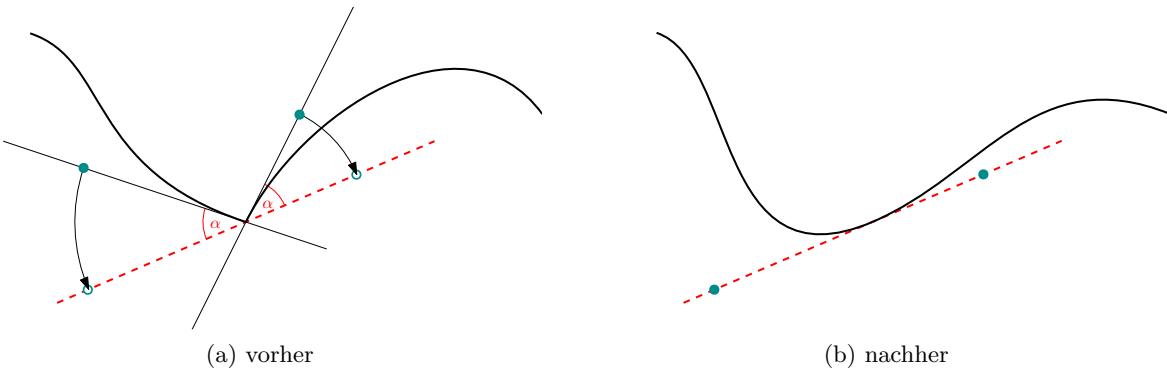


Abbildung 9.3.: Anpassung der Kontrollpunkte der Wirbelsäulenteile, wenn die Steigung an den Kontaktstellen ungleich ist. Die beiden Teile der Wirbelsäule und die Steigung am Kontaktstellen sind hier in schwarz dargestellt. Die zu drehenden Kontrollpunkte in cyan. In rot ist die resultierende Steigung und der Winkel, der für die Drehung verwendet wird, zu sehen.

wurde hier so gemacht, dass die zu verschiebenden Kontrollpunkte um den Kontaktstellen in gegensätzliche Richtungen rotiert wurden. Beide werden um den gleichen Winkel rotiert (siehe Abbildung 9.3). Grundsätzlich könnten die Kontrollpunkte beliebig auf der Tangente des Kontaktstellen platziert werden (rot im Bild) um eine übereinstimmende Steigung zu bekommen. Um den Verlauf der Wirbelsäulenteile möglichst wenig zu verändern ist es von Vorteil auch die Kontrollpunkte möglichst wenig zu verschieben. Man könnte die Punkte z. B. auch senkrecht auf die Tangente projizieren. Welches Verfahren angewendet wird, ist jedoch nicht von großer Bedeutung.

## 9.7. Gelenke

**ToDo**

(Für jede Art von Extremität und jede Position ein anderes Gelenk mit anderen Einschränkungen (nicht alle Möglichkeiten in einem))

## 9.8. Probleme der Beinpositionierung bei kurzen Beinen

Wie in Absatz 7.2.2 zur Ausrichtung der Beine bereits beschrieben, treten bei sehr kurzen Beinen ein paar Probleme auf. Diese werden im Folgenden kurz umrissen.

- Bei Berechnung der Bodenhöhe wird die Beinlänge von den entsprechenden Kontrollpunkten der Bézierkurve aus gemessen, da die Position der Hüfte/Schulter in diesem Stadium noch nicht klar ist. Deshalb kann es bei sehr kurzen Beinen sein, dass der Abstand zwischen Boden und Gelenk zu groß ist und der Boden nicht erreichbar ist.

In diesem Fall wird das Bein einfach komplett ausgestreckt senkrecht nach unten positioniert. Da der Abstand zwischen dem Hüft- / Schultergelenk und dem Kontaktstellen der Bézierkurve nicht sehr groß ist, ist auch der Abstand zu Bodenhöhe nicht enorm und fällt nicht allzusehr auf.

- Bei sehr kurzen Knochen ändert sich der Abstand zum Boden durch Drehung der Gelenke nicht so stark wie bei langen Knochen. Wenn der Winkel sich zu wenig ändert, wird aber davon ausgegangen, dass die Winkeländerung zu klein ist und alle Änderungen sofort wieder zurückgesetzt wurden. Deshalb wird die Winkeländerung

dann für die nächste Iteration stärker verkleinert. Das ist aber in diesem Fall kontraproduktiv. Der Algorithmus schafft es dann nämlich nicht mehr die Knochen in die richtige Lage zu bringen, da der Bewegungsspielraum dadurch zu stark eingeschränkt wird.

Dem könnte man natürlich entgegenwirken, indem man, statt die Änderung des Abstandes zum Boden zu messen, abfragt wie oft die Drehungen in der jeweiligen Iteration zurückgesetzt wurden.

3. Bei wirklich sehr kurzen Beinen (hier eine Gesamtlänge unter 5) macht es gar keinen Sinn sie anzuzuordnen, da man sie kaum sieht. Außerdem tritt hier der Effekt auf, der in Punkt 2 schon beschrieben wurde.
  4. Die Beinstartposition kann schon vor der ersten Iteration unterhalb der Bodenhöhe liegen. Das tritt auf, wenn mindestens ein Paar Beine sehr kurz ist und die Wirbelsäule an den Ansatzpunkten auf sehr unterschiedlicher Höhe liegt (siehe auch Absatz 7.2.1 zur Berechnung der Bodenhöhe). (**Beispiel: Dimetrodon**) ToDo
- In diesem Fall ist der Algorithmus natürlich nicht anwendbar. Das Bein wird dann einfach senkrecht nach unten ausgerichtet und mit einem Fuß versehen, der mit der Sohle auf dem Boden steht.
5. Bei kurzen Beinknochen kann es dazu kommen, dass der Oberschenkel nicht näher zum Boden kann, weil er schon aufliegt, aber Schienbein und Hand durch das Gelenkoffset (siehe Abschnitt 7.4 zur Positionierung der Knochenmodelle) über dem Boden schweben und nicht näher zum Boden bewegt werden können. (**Beispiel Krokodil** ToDo (**siehe auch Screenshot**))

# 10. Erweiterungs- und Verbesserungsmöglichkeiten

- Muskeln, Haut etc. anbauen
- austauschbare 3D-Modelle
- mehr verschiedene 3D-Modelle
- Beinalgo verbessern / IK
- mehr Daten für PCA sammeln
- Gewicht in Algo verwenden
- Erweiterung auf Mensch? Mensch auch als PCA Datenpunkt?

## 10.1. Muskeln

- Die „Hauptmuskeln“ verlaufen bei Wirbeltieren wahrscheinlich ähnlich, relativ zu den Knochen. Trotzdem unterscheiden sie sich recht stark.
- Knochen/Gelenke bekommen Zusatzattribute für Start- und Zielpunkte der Muskeln.
- Muskeln haben eine „Dicke“ und aus Start- und Zielpunkt muss Kurve des Muskels generiert werden.
- Wie wird die genaue Form festgelegt? Muskeln irgendwie auf ihre „Dicke“ aufblähen + Interaktion mit vorhandenen Elementen (andere Muskeln und Knochen) → future work

## 10.2. Haut

- Was für Algorithmen gibt es, die zu einem vorhandenen 3D-Modell eine Hülle mit gewissen Eigenschaften generieren?  
es gibt eine solche Funktion z.B. in AutoCAD <https://knowledge.autodesk.com/de/support/autocad/learn-explore/caas/CloudHelp/cloudhelp/2016/DEU/AutoCAD-Core/files/GUID-B7F99810-765E-4E7E-ABDD-275C64147CCC-.html>
- Einfach nur eine Hülle mit gewissem Abstand sieht wahrscheinlich sehr unrealistisch aus. „Bony Landmarks“ (Stellen an denen das Gewebe über den Knochen sehr dünn ist) könnten helfen (siehe <https://www.proko.com/landmarks-of-the-human-body/>) oder „bone weights“

# Literatur

- [1] A. Aristidou u. a. “Inverse Kinematics Techniques in Computer Graphics: A Survey”. In: *Computer Graphics Forum* 37.6 (2018), S. 35–58. URL: <https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.13310>.
- [2] Esben Bach und Andreas Madsen. “Building and Generating Facial Textures using Eigen Faces”. Masterarbeit. Harbin Institute of Technology und Linköping University, 2016.
- [3] Nuno Barreto und Licinio Roque. “A Survey of Procedural Content Generation tools in Video Game Creature Design”. In: *Second Conference on Computation Communication Aesthetics and X*. Juni 2014. URL: [https://www.researchgate.net/publication/298307183\\_A\\_Survey\\_of\\_Procedural\\_Content\\_Generation\\_tools\\_in\\_Video\\_Game\\_Creature\\_Design](https://www.researchgate.net/publication/298307183_A_Survey_of_Procedural_Content_Generation_tools_in_Video_Game_Creature_Design).
- [4] Volker Blanz und Thomas Vetter. “A Morphable Model for the Synthesis of 3D Faces”. In: *SIGGRAPH’99 Proceedings of the 26th annual conference on Computer graphics and interactive techniques* (Sep. 2002). URL: <https://dl.acm.org/doi/10.1145/311535.311556>.
- [5] Demoszene. URL: <https://www.demoscene.info/> (besucht am 06.05.2020).
- [6] Morris L. Eaton. *Multivariate Statistics. A Vector Space Approach*. Lecture Notes-Monograph Series. Beachwood, Ohio: Institute of Mathematical Statistics, 1983. ISBN: 9780940600690. URL: <https://epdf.pub/multivariate-statistics-a-vector-space-approacheed2a525dca43e48f58a88b289ff519f66368.html>.
- [7] Wilhelm Ellenberger. *Handbuch der Anatomie der Tiere für Künstler*. Bd. 1. Leipzig: Dietrichsche Verlagsbuchhandlung, 1900. URL: [https://archive.org/details/bub\\_gb\\_4JFmAAAAMAAJ/page/n19/mode/2up](https://archive.org/details/bub_gb_4JFmAAAAMAAJ/page/n19/mode/2up).
- [8] Jonas Freiknecht und Wolfgang Effelsberg. “A Survey on the Procedural Generation of Virtual Worlds”. In: *Multimodal Technologies and Interaction*. Bd. 1. 4. Artikel 27. 2017. URL: <https://doi.org/10.3390/mti1040027>.
- [9] Ernst Hadorn und Rüdiger Wehner. *Allgemeine Zoologie*. 21. Aufl. Stuttgart: Georg Thieme Verlag, 1986. ISBN: 3-423-04061-0.
- [10] James Scott Hanan. “Parametric L-systems and Their Application to the Modelling and Visualization of Plants”. AAIINN83871. Diss. 1992. ISBN: 0-315-83871-X.
- [11] Milton Hildebrand und George E. Goslow. *Vergleichende und funktionelle Anatomie der Wirbeltiere*. Berlin, Heidelberg: Springer-Verlag, 2004.

- [12] Jon Hudson. "Creature Generation using Genetic Algorithms and Auto-Rigging". Masterarbeit. Bournemouth University, National Centre for Computer Animation, 2013.
- [13] I.T. Jolliffe. *Principal Component Analysis*. 2. Aufl. New York: Springer-Verlag, 2002. ISBN: 0-387-95442-2.
- [14] Robert Krogh. "Procedural Character Generation. Implementing Reference Fitting and Principal Components Analysis". Masterarbeit. Aalborg University, 2007. URL: [https://projekter.aau.dk/projekter/en/studentthesis/procedural-character-generation-implementing-reference-fitting-and-principal-components-analysis\(8a268f9e-a382-496b-8a94-a94fe6606d2e\).html](https://projekter.aau.dk/projekter/en/studentthesis/procedural-character-generation-implementing-reference-fitting-and-principal-components-analysis(8a268f9e-a382-496b-8a94-a94fe6606d2e).html).
- [15] Prof. Dr. Jörn Müller-Quade u. a. *Theoretische Grundlagen der Informatik*. Wintersemester 2013/2014. URL: [https://crypto.iti.kit.edu/fileadmin/User/TGI\\_WS13/Skript\\_Beth.pdf](https://crypto.iti.kit.edu/fileadmin/User/TGI_WS13/Skript_Beth.pdf).
- [16] *No Man's Sky*. URL: <https://www.nomanssky.com/> (besucht am 08.10.2019).
- [17] *Our Centaurs Are Different aka: Centauroid Form*. URL: <https://tvtropes.org/pmwiki/pmwiki.php/Main/CentauroidForm> (besucht am 07.05.2020).
- [18] *Pegasus*. URL: <https://tvtropes.org/pmwiki/pmwiki.php/Main/Pegasus> (besucht am 07.05.2020).
- [19] Martin Petratsch. "Prozedurale Städtegenerierung mit Hilfe von L-Systemen". Großer Beleg. Technische Universität Dresden, 2008. URL: [http://www.inf.tu-dresden.de/index.php?node\\_id=2234&ln=de](http://www.inf.tu-dresden.de/index.php?node_id=2234&ln=de).
- [20] Hartmut Prautzsch, Wolfgang Boehm und Marco Paluszny. *Bezier and B-Spline Techniques*. Berlin, Heidelberg: Springer-Verlag, 2002. ISBN: 3540437614.
- [21] Eric N. Rittmeyer u. a. "Ecological Guild Evolution and the Discovery of the World's Smallest Vertebrate". In: *PLoS ONE* 7 (Jan. 2012). URL: <https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0029797>.
- [22] Hyewon Seo und Nadia Thalmann. "An Automatic modeling of human bodies from sizing parameters". In: Jan. 2003, S. 19–26. URL: <https://dl.acm.org/doi/10.1145/641480.641487>.
- [23] Harumichi SHINOHARA. "The Mouse Vertebrae: Changes in the Morphology of Mouse Vertebrae Exhibit Specific Patterns Over Limited Numbers of Vertebral Levels". In: *Okajimas folia anatomica Japonica* 76 (Juni 1999), S. 17–31. URL: [https://www.jstage.jst.go.jp/article/ofaj1936/76/1/76\\_17/\\_pdf%EF%BB%BF](https://www.jstage.jst.go.jp/article/ofaj1936/76/1/76_17/_pdf%EF%BB%BF).
- [24] *SpeedTree*. URL: <https://store.speedtree.com/> (besucht am 05.05.2020).
- [25] D'Arcy Wentworth Thompson [Verfasser]. *Über Wachstum und Form*. Hrsg. von John Tyler Bonner und Adolf Portmann [Verfasser eines Geleitwortes]. In gekürzter Fassung neu hrsg. vom John Tyler Bonner. Wissenschaft und Kultur ; 26. Neu hrsg. und gekürzte Fass. der Ausg. Cambridge 1966. Basel: Birkhäuser, 1973. ISBN: 3764305363.
- [26] *Vertebrate with Extra Limbs*. URL: <https://tvtropes.org/pmwiki/pmwiki.php/Main/VertebrateWithExtraLimbs> (besucht am 07.05.2020).
- [27] Günter Vogel und Hartmut Angermann. *dtv-Atlas zur Biologie*. 6. Aufl. Band II. München: Deutscher Taschenbuch Verlag GmbH & Co. KG, 1968. ISBN: 3-423-03012-7.
- [28] *Vogelskelett*. URL: <https://de.wikipedia.org/wiki/Vogelskelett> (besucht am 06.05.2020).

- [29] Hongling Wang, Joseph Kearney und Kendall Atkinson. "Arc-length parameterized spline curves for real-time simulation". In: (Jan. 2002). URL: [https://www.researchgate.net/publication/228718576\\_Arc-length\\_parameterized\\_spline\\_curves\\_for\\_real-time\\_simulation](https://www.researchgate.net/publication/228718576_Arc-length_parameterized_spline_curves_for_real-time_simulation).
- [30] Rüdiger Wehner und Walter Gehring. *Zoologie*. 24. Aufl. Stuttgart: Georg Thieme Verlag, 2007. ISBN: 978-3-13-367424-9.
- [31] Rüdiger Wehner und Walter Gehring. *Zoologie*. 25. Aufl. Stuttgart: Georg Thieme Verlag, 2013. ISBN: 978-3-13-367425-6.
- [32] Wilfried Westheide und Reinhard Rieger. *Spezielle Zoologie. Teil 2: Wirbel- oder Schädeltiere*. München: Spektrum Akademischer Verlag, 2004. ISBN: 3-8274-0307-3.

# A. Vergleichende Anatomie

Notizen aus „Vergleichende und funktionelle Anatomie der Wirbeltiere“ [11]:

## Kapitel 1 - Einleitung

- „Nichts anderes in der Natur hat eine herrlichere Struktur als der Körper der Wirbeltiere.“ (S. 1)
- Anpassung des Körpers an äußere Gegebenheiten (S. 3). → Eingabeparameter?
- (phylogenetische) Homologie von Strukturen: gemeinsame Abstammung (meist auch gleiche Funktion) (S. 4 und 6)
- Analogie von Strukturen: gleiche Funktion aber nicht gleiche phylogenetische Herkunft (S. 6)
- serielle Homologie: homologe Gene an verschiedenen Körperteilen aktiviert (z. B. Wirbel) (S. 7) → gleiche/sehr ähnliche Regeln
- rudimentäre/degenerierte Organe: haben keine Funktion mehr, waren aber bei Vorfahren funktionell (z. B. Beckengürtel des Finnwals) (S. 9/10)
- Veränderungen von Körperproportionen (z. B. Schädelknochen) können durch eine fortschreitende Verzerrung eines Gitters beschrieben werden (S. 18/19), siehe auch Anhang B

## Kapitel 2 - Charakterisierung , Ursprung und Einteilung der Vertebraten

- „[...] ein Tier mit einem Cranium, also einer skelettartigen Schädelkapsel, [ist] ein Vertebrat.“ (S. 27)
- Teil der allgemeine Beschreibung der (meisten) Vertebraten (S. 27):
  - „Der Körper ist bilateralsymmetrisch, d. h. er weist eine rechte und eine linke Seite, ein anteriores und ein posteriores Ende und eine dorsale und eine ventrale Oberfläche auf.“
  - Sie haben ein inneres Skelett.

## Kapitel 3 - Fische

- Agnathen (Kieferlose): erste bekannte Vertebraten und einzige ohne Kiefer. Einzige rezente Arten: Neunauge und Schleimaal (S. 41/42)

- Kiefertragende Fische: 1. Kiemenbogen entwickelte sich zu Kiefer, haben paarige Flossen (S. 45/46)
  - Knorpelfische: Haie, Rochen, Chimären; verkalkte Knorpel aber wenige/keine Knochen (S. 47) (Unterschied Knochen/Knorpel ignorieren oder nur Knochenfischskelette generieren?)
  - Knochenfische: Strahlenflosser (Flossen mit knöchernen Strahlen), Fleischflosser (Flossen mit fleischigen Stielen) (S. 52/53)

#### Kapitel 4 - Tetrapoden

- an Land ist Stromlinienform und sind Flossen kein Vorteil mehr, ein Hals ist nützlich und der Körper muss von Beinen getragen werden → Extremitätengürtel fester mit Axialskelett verbunden (S. 59)
- Amphibien: nicht vollkommen terrestrisch (Haut feucht, Eier im Wasser / feucht). Unterklasse Lissamphibia (dazu gehören alle rezenten Amphibien) haben nur vier Zehen am Vorderfuß. Es gibt drei Ordnungen: Anura (Schwanzlose), Urodela (Schwanzlurche) und Apoda (Beinlose) (S. 61)
- Reptilien: erste Klasse mit allen Strukturen für vollkommen landgebundenes Leben, leben aber auch teilweise wieder im Wasser (S. 62)
- Vögel: alle rezenten Vögel fliegen oder sind Nachkommen von Fliegern. Erste Zehe ist opponiert (siehe Diagramm S. 71). (S. 67)
- Säugetiere (S. 71 f)

#### Kapitel 8 - (Kopfskelett)

- „Das innere, gelenkige Skelettsystem der Vertebraten ist einzigartig im Tierreich. Es ist das wichtigste aller Organsysteme für das Studium der Wirbeltiermorphologie.“ (S. 131)
- Muskeln haben Ansatzstellen an Knochen, die Lage und Ausmaße zeigen. (S. 131)

#### Kapitel 9 - Körperskelett

- „Die Wirbelsäule ist älter als jeder andere Teil des postcranialen Skeletts mit Ausnahme der Chorda dorsalis. Dennoch ist sie nicht so alt wie die Hauptmerkmale der weichen Organsysteme, und sie fehlt praktisch bei den ältesten, bekannten Vertebraten.“ (S. 163)
- Wirbel können viele verschiedene Merkmale haben, z. B. Dornfortsatz, Ansatz der Rippe (siehe Bild S. 165) (S. 163)
- Evolution der Wirbelsäule: zunächst Chorda dorsalis mit stützenden Knorpeln (S. 166)
- Rippen
  - ursprünglich über die gesamte Länge der Wirbelsäule vorhanden. Zwei Arten: Dorsalrippen und Ventralrippen (siehe Abb. S. 173) (S. 172)
  - keine Rippen bei kieferlosen Vertebraten und Placodermi
- Mediane Flossen (175 f)
  - Dorsalflosse, Analflosse und Schwanzflosse
  - treten bei fast allen Agnatha und kiefertragenden Fischen auf

- Wirbelsäule kann unterschiedlich in Schwanzflosse liegen (gerade, nach oben/unten geknickt oder sie hört davor auf), meistens nach oben abgeknickt (heterocerk).
- Schultergürtel der Fische ist mit Kopf verbunden (S. 178) (also kein Hals)
- Beckengürtel der Tetrapoden viel größer als bei Fischen. Bei verschiednene Gruppen unterschiedlich aber speziell (z. B. Vogel, Säugetier,...) (S. 180 f) → diskret repräsentieren?
- S. 183 Abbildung zur Evolution der Vordergliedmaßen
- Amphibien haben meistens kurze Gliedmaßen, die seitlich des Körpers nach außen gestellt sind (S. 187)
- Gliedmaßen der Reptilien oft seitlich aber manche auch unter dem Körper (wie bei Säugetieren) (S. 188 f)

**Kapitel 21 - Strukturelemente des Körpers**

- „allgemein nützliche“ Strukturen: Kiefer, zwei Paare von Extremitäten (S. 433)
- Skelett kann nicht einfach „groß skaliert“ werden. Belastung ist sonst möglicherweise zu groß (S. 444, siehe auch S. 478 und S. 481)
- Knochen halten am besten Druckkraft aus → Minimierung anderer Kräfte (S. 444)(schlecht bei Zug oder seitlicher Belastung)

**Kapitel 22 - Mechanik von Stützung und Bewegung**

- Balance und Gegenmoment (S. 466)

## B. D'Arcy Thompson

(siehe auch [https://en.wikipedia.org/wiki/On\\_Growth\\_and\\_Form](https://en.wikipedia.org/wiki/On_Growth_and_Form))

Notizen aus „Über Wachstum und Form“ von D'Arcy Wentworth Thompson [25]:

### Kapitel 2 - Über die Größe

- Maximale und minimale Größe eines Tieres sind durch unterschiedliche Faktoren, wie z. B. das Atmungssystem festgelegt. Bei Säugetieren: min. 5g (Maus) (S. 68) und max. sowas wie Elefant

### Kapitel 4 - Über die Theorie der Transformationen oder den Vergleich verwandter Formen

- „Es handelt sich hier um das berühmteste Kapitel des Buches, das in der biologischen Literatur schon vielfach besprochen worden ist.“ (S. 325, Kommentar des Herausgebers (?))
- Kartesische Transformationen (S. 334 ff)
  1. entlang einer Achse ausdehnen
  2. logarithmische Verlängerung
  3. einfache Scherung
  4. radiale Koordinaten
- Beispiele z. B. S. 366 f, Säugetierschädel ab S. 371

## C. Erhobene Daten für PCA

### C.1. Mit PCA erzeugte Datenpunkte

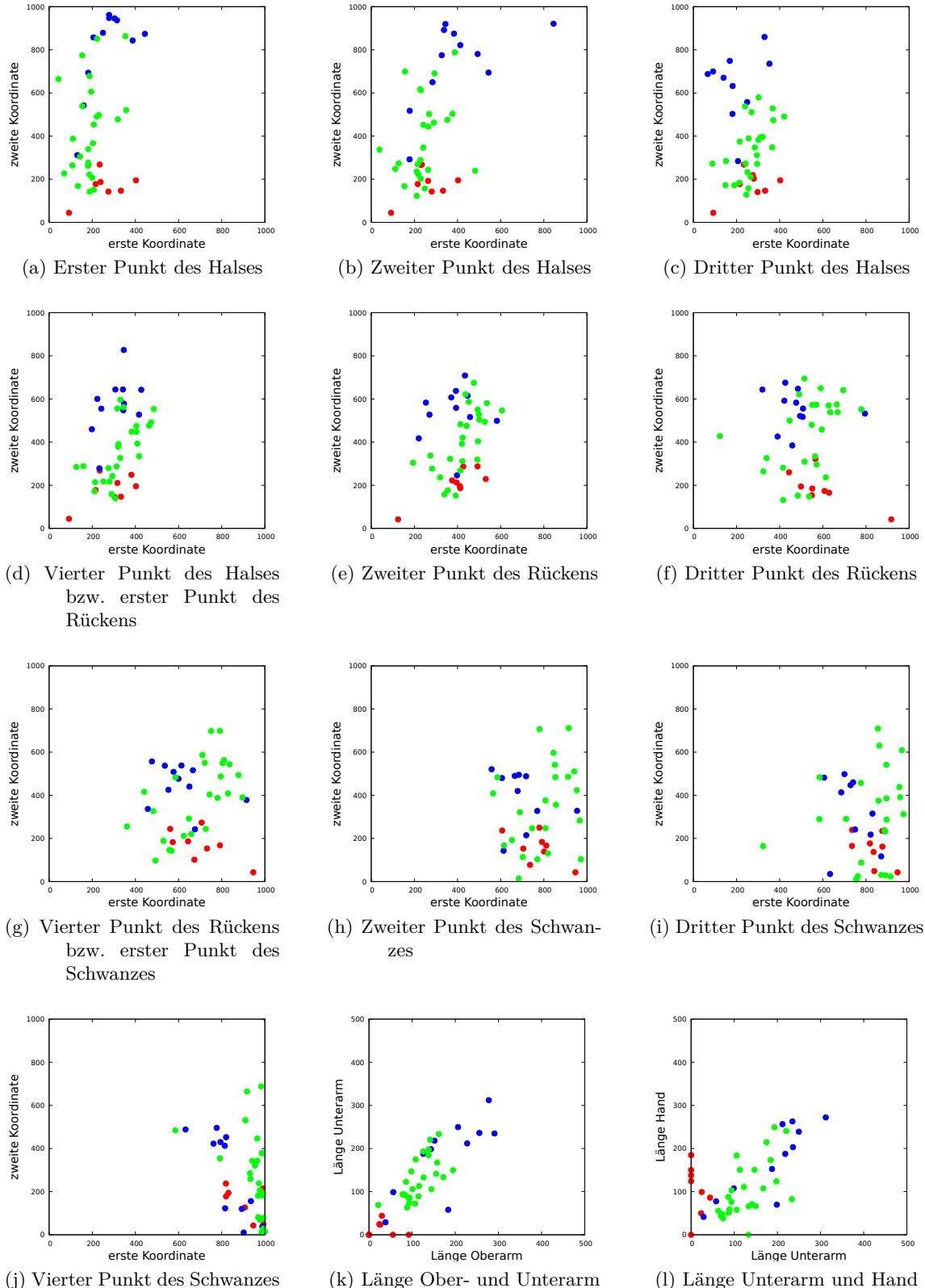
### C.2. Skelettbilder

**ToDo**

(alphabetisch sortieren)

alle Bilder aus [32] außer:

- Sinornis und Taube aus [11]
- Seekuh aus [31]
- Archaeopteryx, Eusthenopteron, Ichthyosaurus, Ichthyostega, Muraenosaurus, Ur-pferdchen aus [30]
- Pferd: <https://www.kosmos.de/content/buecher/ratgeber/pferde-reiten/vorwaerts-abwaerts>
- Känguru <http://www.bildwoerterbuch.com/tierreich/beuteltiere/kaenguru/skelett-eines-kaengurus.php>
- Schwan <https://www.alamy.de/skelett-eines-schwans-osteographia-oder-die-anatomie-der.html>
- Chamäleon <https://www.madcham.de/de/anatomie/>
- Gnu <https://lutzmoeller.net/Afrika/2007/Lutz-Juli/8-Gnu.php>
- Tyrannosaurus Rex [https://upload.wikimedia.org/wikipedia/commons/9/9f/Tyrannosaurus\\_skeleton.jpg](https://upload.wikimedia.org/wikipedia/commons/9/9f/Tyrannosaurus_skeleton.jpg)
- Dormedar [https://upload.wikimedia.org/wikipedia/commons/a/ac/Camel\\_Skeleton\\_-\\_Richard\\_Owen\\_-\\_On\\_the\\_Anatomy\\_of\\_Vertebrates\\_%281866%29.jpg](https://upload.wikimedia.org/wikipedia/commons/a/ac/Camel_Skeleton_-_Richard_Owen_-_On_the_Anatomy_of_Vertebrates_%281866%29.jpg)
- Strauß <https://www.alamy.de/stockfoto-skelett-von-strauss-24658845.html>
- Blauwal <https://www.quagga-illustrations.de/wp-content/uploads/2014/05/h0001705.jpg>
- Krokodil <https://de.depositphotos.com/210906852/stock-illustration-skeleton-crocodile.html>
- Giraffe [https://de.wikipedia.org/wiki/Giraffen#/media/Datei:Giraffe\\_skeleton.jpg](https://de.wikipedia.org/wiki/Giraffen#/media/Datei:Giraffe_skeleton.jpg)



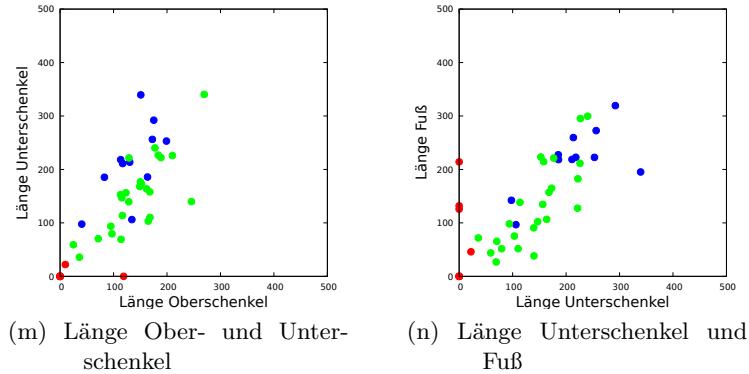


Abbildung C.1.: Erhobene Daten: Punkte der Bézierkurven der Wirbelsäule und Längen der Extremitäten. Bei den Extremitäten ist jeweils gegeneinander abgetragen Ober- und Unterarm, Ober- und Unterschenkel, Unterarm und Hand, Unterschenkel und Fuß. Markiert ist jeweils ob die Datenpunkte 0 (rot), 2 (blau) oder 4 (grün) Beine haben.

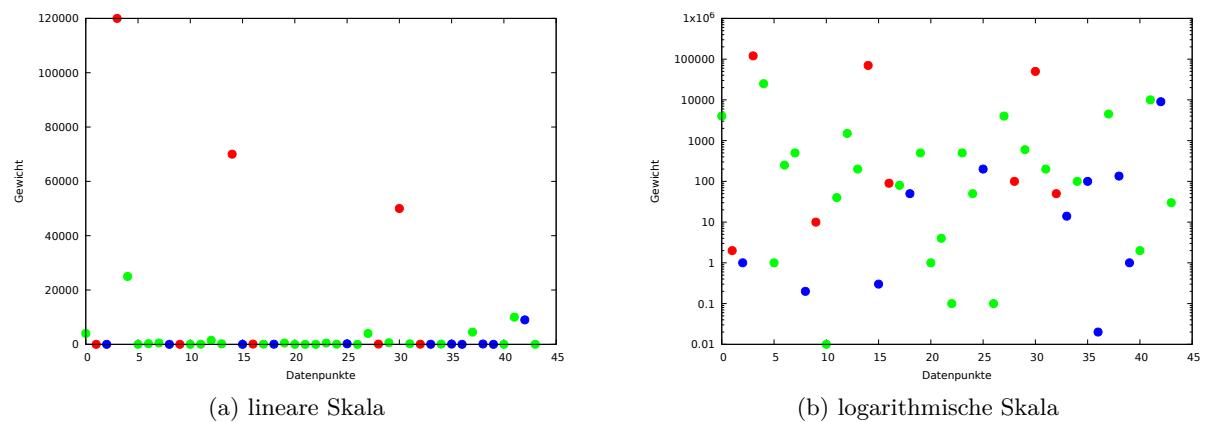
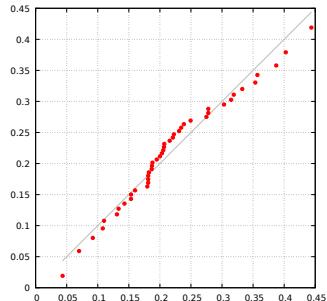
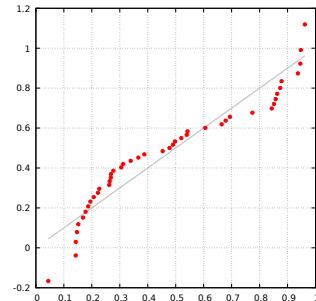


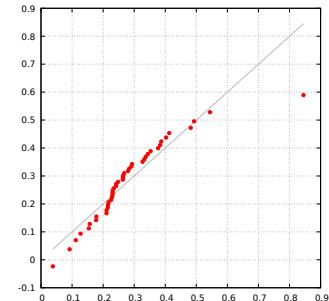
Abbildung C.2.: Erhobene Daten: Gewicht. Markiert ist jeweils ob die Datenpunkte 0 (rot), 2 (blau) oder 4 (grün) Beine haben.



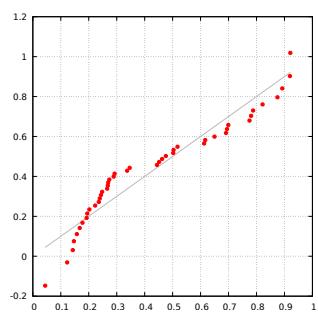
(a) Erster Punkt des Halses, x-Koordinate



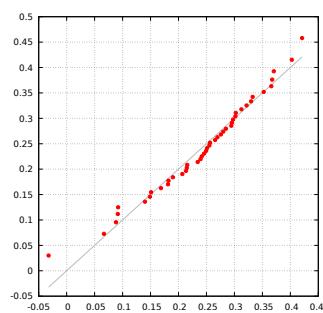
(b) Erster Punkt des Halses, y-Koordinate



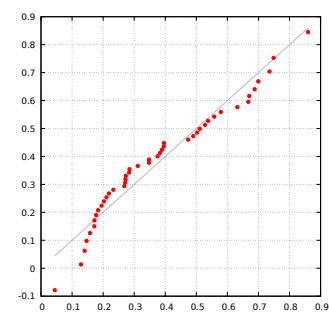
(c) Zweiter Punkt des Halses, x-Koordinate



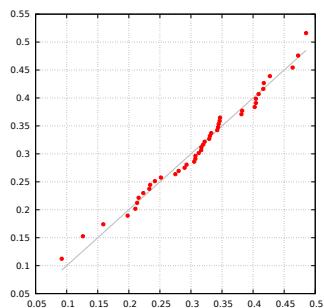
(d) Zweiter Punkt des Halses, y-Koordinate



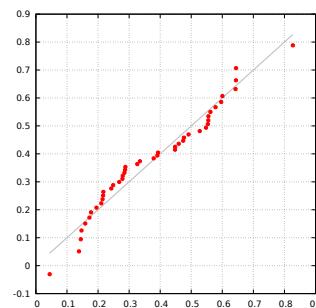
(e) Dritter Punkt des Halses, x-Koordinate



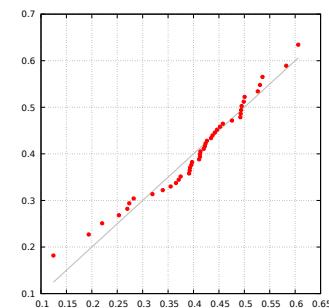
(f) Dritter Punkt des Hales, y-Koordinate



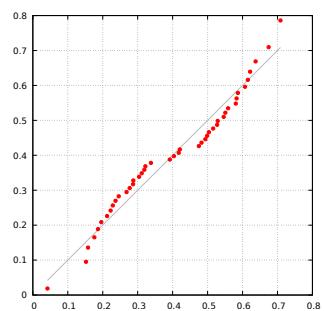
(g) Erster Punkt des Rückens, x-Koordinate



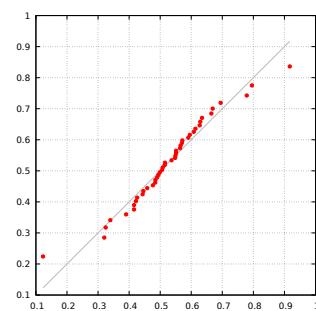
(h) Erster Punkt des Rückens, y-Koordinate



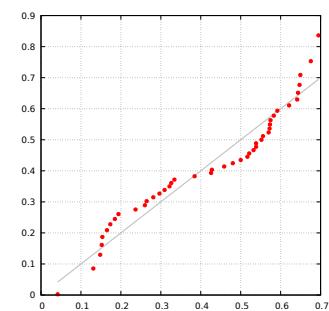
(i) Zweiter Punkt des Rückens, x-Koordinate



(j) Zweiter Punkt des Rückens, y-Koordinate



(k) Dritter Punkt des Rückens, x-Koordinate



(l) Dritter Punkt des Rückens, y-Koordinate

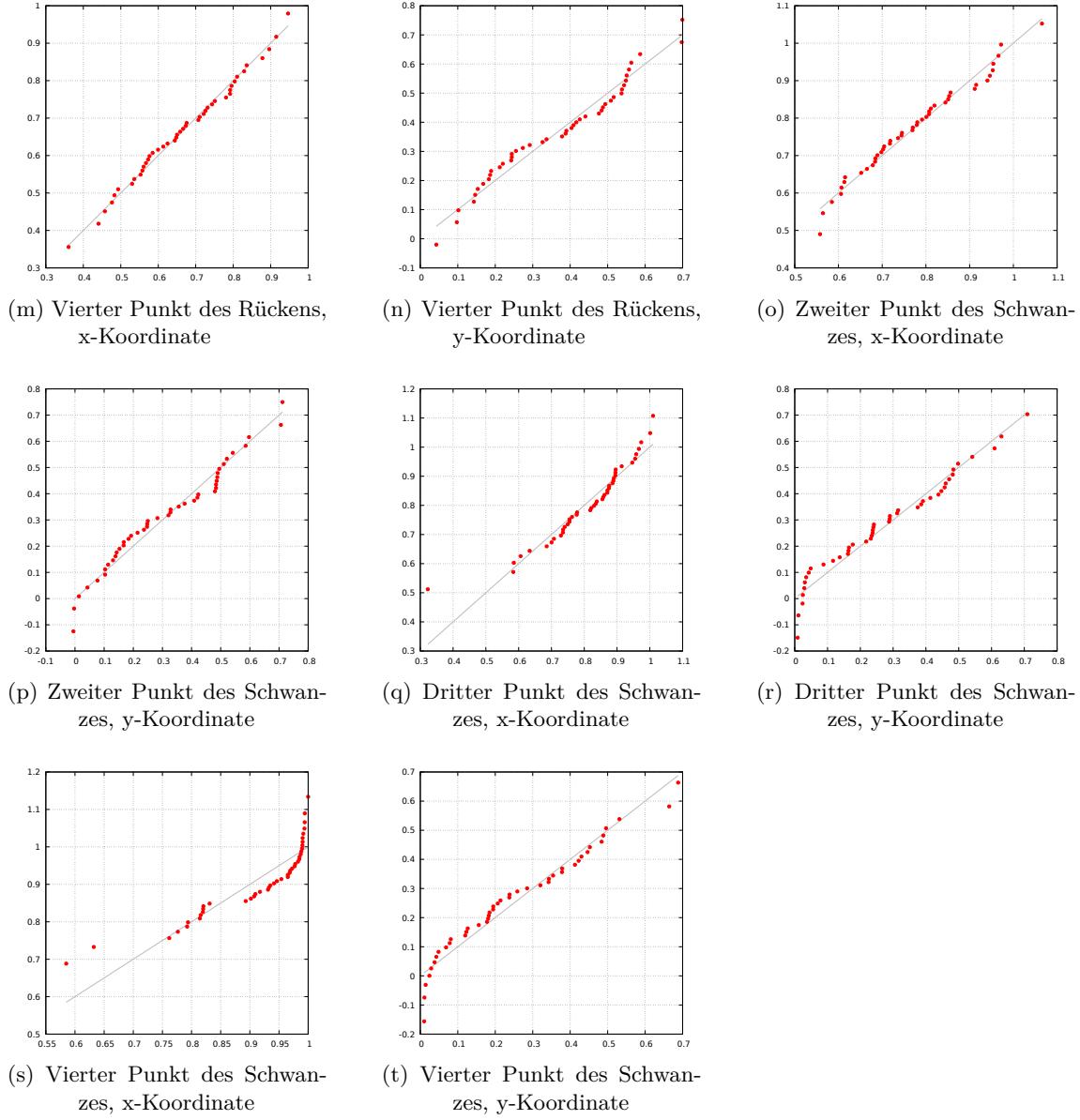


Abbildung C.3.: Quantil-Quantil-Diagramme für alle Dimensionen der Wirbelsäule

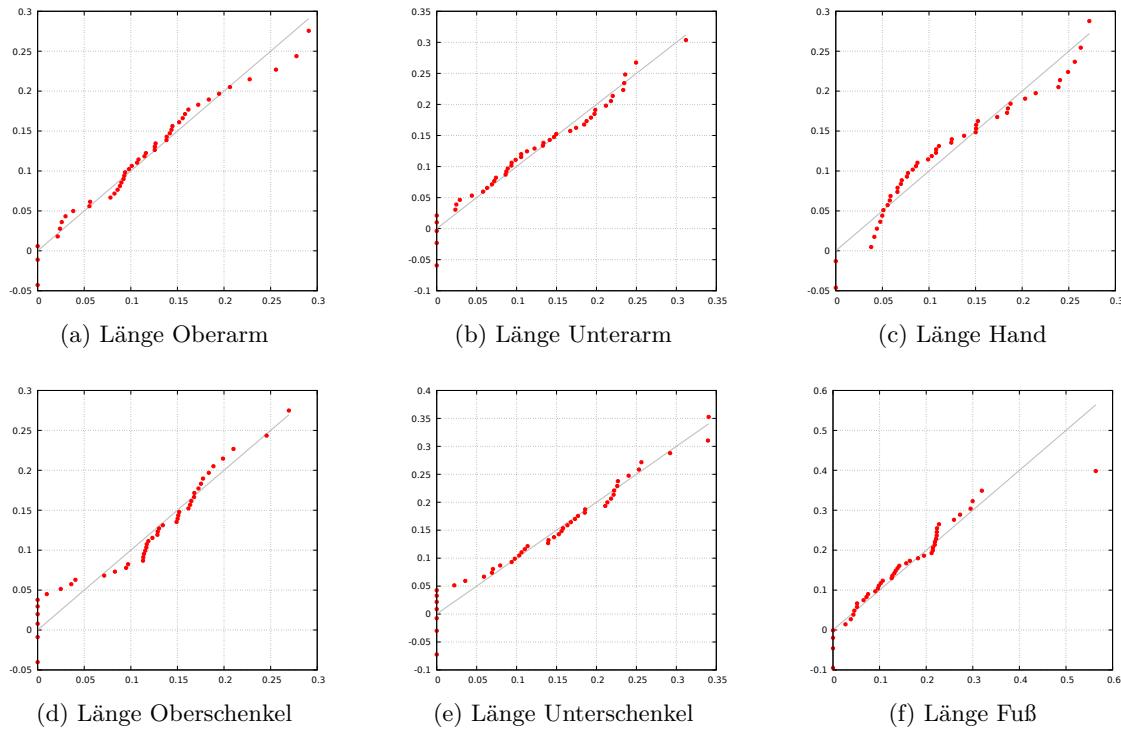


Abbildung C.4.: Quantil-Quantil-Diagramme für die Dimensionen der Eingabedaten, die zusätzlich zur Wirbelsäule erhoben wurden. Nicht dargestellt ist das binäre Attribut *Flügel* und die Anzahl der Beine mit Bodenkontakt.

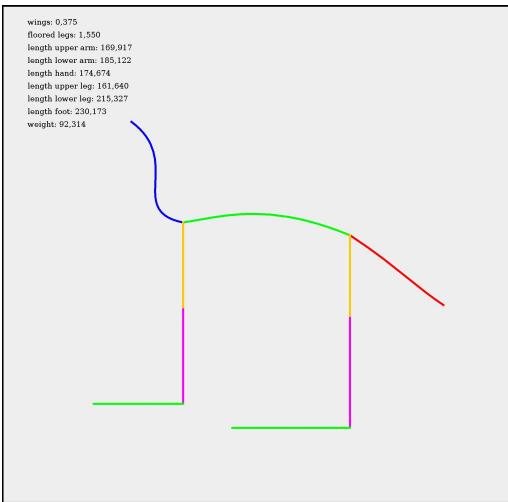
- Schlange: zu Schlangen gibt es keine Bilder, die deren Skelett in ausgestrecktem Zustand von der Seite darstellen. Deshalb wurde ein leeres Bild genommen und der Verlauf des Rückens durch eine Gerade angenähert (Extremitäten besitzt eine Schlange nicht. Außerdem ist nicht erkennbar ob bzw. wo Hals in Rücken und Rücken in Schwanz übergeht. Deshalb wurde die komplette Wirbelsäule als Rücken markiert.)

## C.3. Gewicht der Tiere

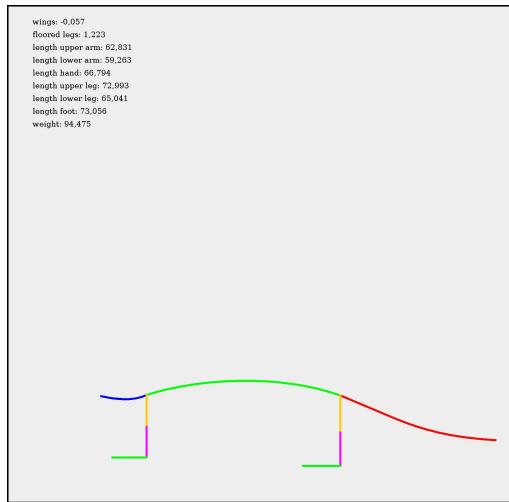
(alphabetisch sortieren)

**To Do**

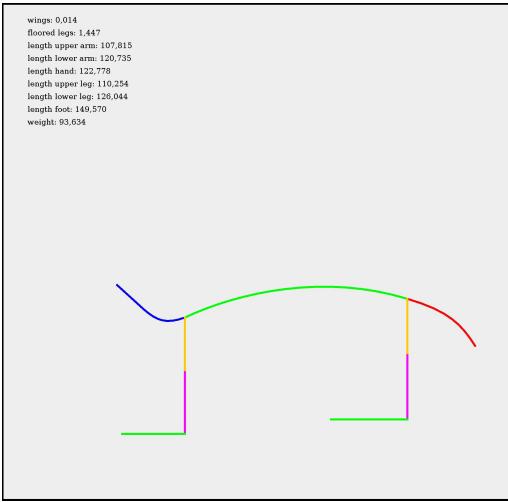
- Blauwal 120 Tonnen, <http://tierdoku.com/index.php?title=Blauwal>, „das schwerste bekannte Tier der Erdgeschichte“ <https://de.wikipedia.org/wiki/Blauwal>
- Durchschnittsgewicht (Warmblut-)Pferd 600 kg, <https://www.reitarena.com/de/blog/blog-post/2015/03/03/das-pferd-grundlegende-fakten.html>
- Afrikanischer Elefant 4000kg, <https://de.upali.ch/gewicht-und-grosse/>
- Amerikanischer Flussbarsch 2kg, [http://tierdoku.com/index.php?title=Amerikanischer\\_Flussbarsch](http://tierdoku.com/index.php?title=Amerikanischer_Flussbarsch)
- Archaeopteryx 1kg, <https://de.wikipedia.org/wiki/Archaeopteryx>
- Brachiosaurus 23-44 Tonnen, <https://de.wikipedia.org/wiki/Brachiosaurus>
- Dimetrodon 250kg, <https://de.wikipedia.org/wiki/Dimetrodon>
- Elster 0,2kg, <https://de.wikipedia.org/wiki/Elster>



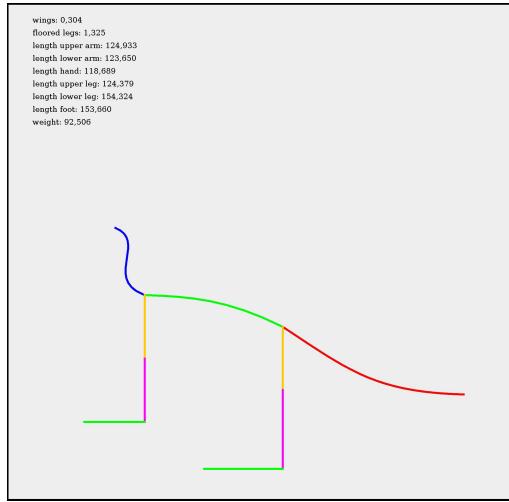
(a) 1-, Flügel 0,38, Beine 1,6, Gewicht 92kg



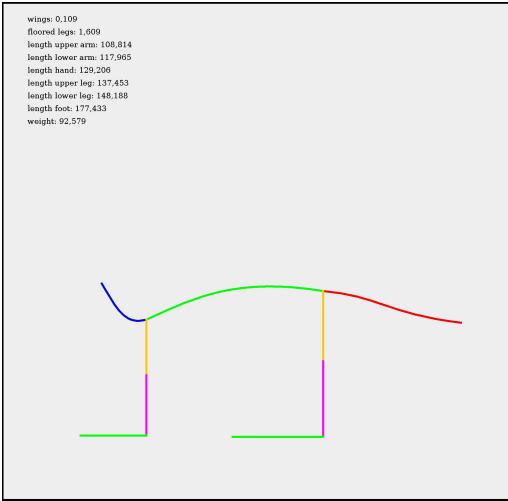
(b) 1+, Flügel -0,057, Beine 1,2, Gewicht 94kg



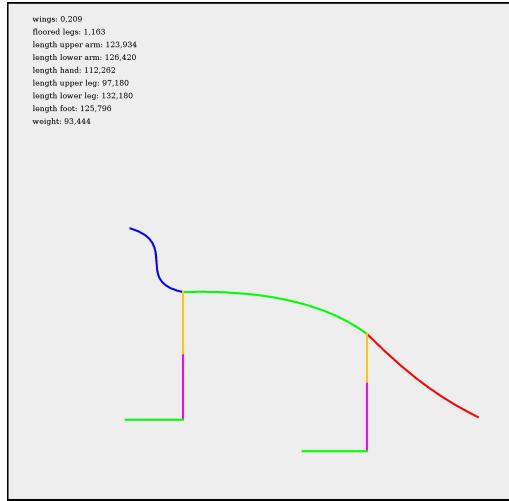
(c) 2-, Flügel 0,014, Beine 1,4, Gewicht 94kg



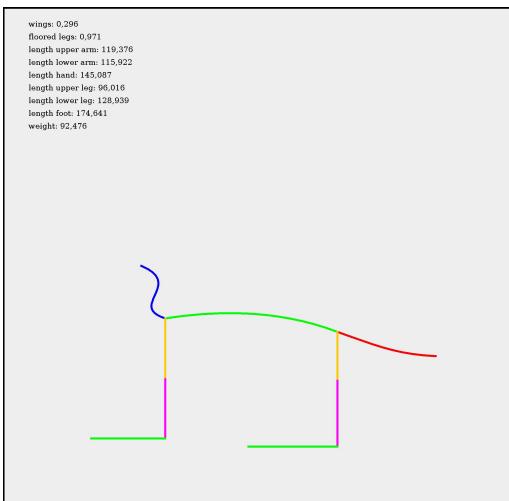
(d) 2+, Flügel 0,3, Beine 1,3, Gewicht 93kg



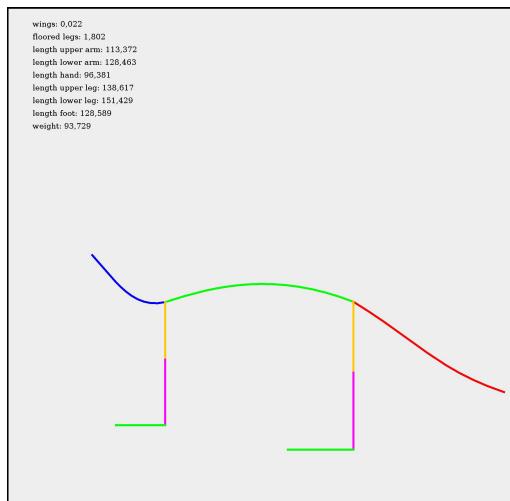
(e) 3-, Flügel 0,11, Beine 1,6, Gewicht 93kg



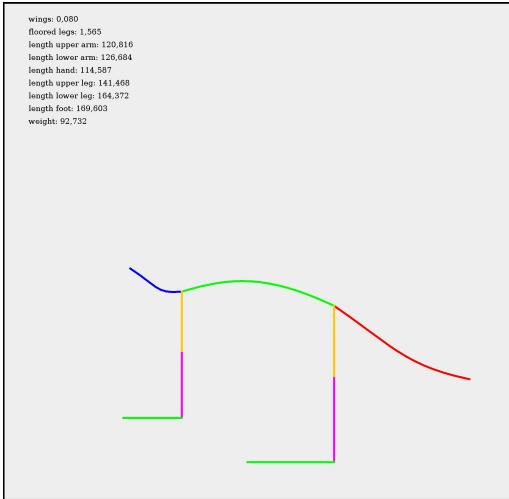
(f) 3+, Flügel 0,21, Beine 1,2, Gewicht 93kg



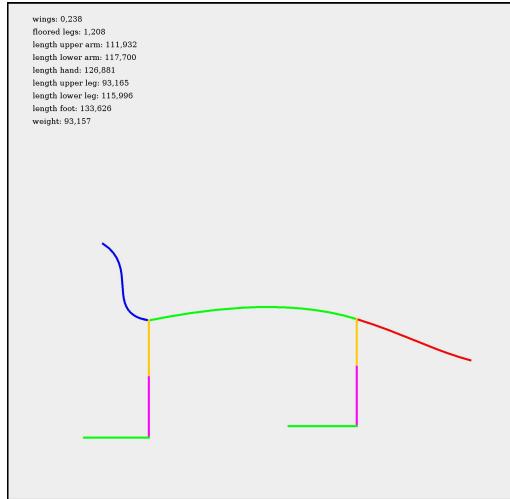
(g) 4-, Flügel 0, 3, Beine 1, Gewicht 92kg



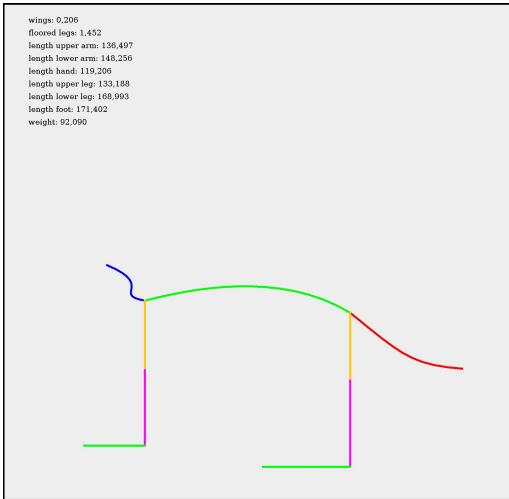
(h) 4+, Flügel 0, 022, Beine 1, 8, Gewicht 94kg



(i) 5-, Flügel 0, 08, Beine 1, 6, Gewicht 93kg



(j) 5+, Flügel 0, 24, Beine 1, 2, Gewicht 93kg



(k) 6-, Flügel 0, 21, Beine 1, 5, Gewicht 92kg



(l) 6+, Flügel 0, 11, Beine 1, 3, Gewicht 98kg

Abbildung C.5.: Datenpunkte im PCA-Koordinatensystem. Eine Koordinate nimmt den Wert der positiven (+) bzw. negativen (-) Standardabweichung in die entsprechende Richtung an, alle anderen sind null. Von oben nach unten sind die Ergebnisse für den größten (1) bis zum sechstgrößten (6) Eigenwert dargestellt. *Beine* steht für *Beine mit Bodenkontakt*.

- Forelle 10-50kg (je nach Art), <https://de.wikipedia.org/wiki/Forelle>
- Grönlandwal 50-100 Tonnen, <https://de.wikipedia.org/wiki/Gr%C3%B6nlandwal>
- Ichthyornis 0.3kg, [http://dinodata.de/animals/birds/pages\\_i/ichthyornis.php](http://dinodata.de/animals/birds/pages_i/ichthyornis.php)
- Ichthyosaurus 90kg, <https://www.tiere-online.de/sonstige-tiere/dinosaurier/ichthyosaurus/>
- Ichthyostega 80kg, <https://dinosaurierwelt.com/ichthyostega/>
- Kaffernbüffel 350-900kg, <https://de.wikipedia.org/wiki/Kaffernb%C3%BCffel>
- Kaninchen je nach Art, ganz grob 1kg
- Klippschliefer 2-5kg, <https://de.wikipedia.org/wiki/Klippschliefer>
- Koboldmaki 0,1kg, <https://de.wikipedia.org/wiki/Koboldmakis>
- Landschildkröte je nach Art, grob 50kg
- Ohrenrobbe 25-500kg, <https://de.wikipedia.org/wiki/Ohrenrobben>
- Panzerspitzmaus 100g ,<https://de.wikipedia.org/wiki/Panzermaus>
- Parasaurolophus walkeri 4-5 Tonnen, [http://tierdoku.com/index.php?title=Parasaurolophus\\_walker](http://tierdoku.com/index.php?title=Parasaurolophus_walker)
- Peloneustes philarchus 100kg, <https://de.wikipedia.org/wiki/Peloneustes>
- Pottwal bis 50 Tonnen, <https://de.wikipedia.org/wiki/Pottwal>
- Rothirsch 80-350kg, <https://de.wikipedia.org/wiki/Rothirsch>
- Seehund 100-150kg, <https://de.wikipedia.org/wiki/Seehund>
- Sinornis 20g, [http://dinodata.de/animals/birds/pages\\_s/sinornis.php](http://dinodata.de/animals/birds/pages_s/sinornis.php)
- Stegosaurus 4,5 Tonnen, <https://de.wikipedia.org/wiki/Stegosaurus>
- Taube je nach Art, grob 1-2kg
- Thrinaxodon Reptil „ein paar Pfund“, <https://www.thoughtco.com/thrinaxodon-1091887>
- Triceratops 6-12 Tonnen, <https://de.wikipedia.org/wiki/Triceratops>
- Urpferdchen (Propalaeotherium) 30kg, <https://de.wikipedia.org/wiki/Propalaeotherium>
- Schwan 14kg, <https://de.wikipedia.org/wiki/Schw%C3%A4ne>
- Chamäleon 0,1-2kg, <https://www.tierchenwelt.de/echsen/128-chamaeleon.html>
- Gämse 25-50kg, <https://de.wikipedia.org/wiki/G%C3%A4mse>
- Gnu 140-250kg, <https://de.wikipedia.org/wiki/Gnu>
- Schwein 100kg, <https://de.wikipedia.org/wiki/Hausenschwein>
- Känguru 2-90kg ,<https://de.wikipedia.org/wiki/K%C3%A4ngurus>
- Tyrannosaurus 9 Tonnen, <https://de.wikipedia.org/wiki/Tyrannosaurus>
- Dromedär 300-700kg, <https://de.wikipedia.org/wiki/Dromedär>
- Afrikanischer Strauß bis 135kg, [https://de.wikipedia.org/wiki/Afrikanischer\\_Stra%C3%9F](https://de.wikipedia.org/wiki/Afrikanischer_Stra%C3%9F)
- Frosch 10g, <http://www.biologie-schule.de/frosch-steckbrief.php>
- Krokodil 100-1000kg, <https://de.wikipedia.org/wiki/Krokodile>
- Schlange bis 100kg bei Riesenschlangen, <https://de.wikipedia.org/wiki/Schlangen>
- Giraffe bis 2 Tonnen, <https://www.tierchenwelt.de/huftiere/73-giraffe.html>

# **Erklärung**

Ich versichere, dass ich die Arbeit selbstständig verfasst habe und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe, die wörtlich oder inhaltlich übernommenen Stellen als solche kenntlich gemacht und die Satzung des KIT zur Sicherung guter wissenschaftlicher Praxis in der jeweils gültigen Fassung beachtet habe. Die Arbeit wurde in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegt und von dieser als Teil einer Prüfungsleistung angenommen.

Karlsruhe, den 13. Mai 2020

(Nina Zimbel)