

# Prozedurale Generierung von Wirbeltierskeletten

Masterarbeit von

**Nina Zimbel**

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

7. April 2020

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

# Inhaltsverzeichnis

<b>1. Bisherige Arbeiten</b>	<b>1</b>
1.1. Ziva . . . . .	1
1.2. ZSpheres in Zbrush . . . . .	1
1.3. 3DS MAX Biped . . . . .	1
1.4. Forensik und Archäologie . . . . .	1
1.5. No Man's Sky . . . . .	2
1.6. Avatar movie . . . . .	2
1.7. Mythologie . . . . .	2
<b>2. Biologie</b>	<b>3</b>
<b>3. Idee</b>	<b>5</b>
3.1. Skelett . . . . .	5
3.1.1. Extremitäten . . . . .	6
3.2. Muskeln . . . . .	6
3.3. Haut . . . . .	6
3.4. Interaktivität . . . . .	7
3.5. Datenstrukturen . . . . .	7
3.5.1. Terminale . . . . .	7
3.5.2. „Wachstum“ . . . . .	7
3.5.3. Balance . . . . .	8
<b>4. PCA</b>	<b>9</b>
4.1. Funktionsweise . . . . .	9
4.2. Datenerhebung . . . . .	10
4.3. Analyse der Eingabedaten . . . . .	11
4.4. Analyse der PCA-Ergebnisse . . . . .	17
4.5. Bedingte Verteilung der PCA Eingabedaten . . . . .	17
<b>5. Algorithmus zur Skelettgenerierung</b>	<b>22</b>
5.1. Generierung der einzelnen Extremitäten . . . . .	22
5.2. Ansatzpunkte für Extremitäten . . . . .	23
5.3. Knochenmodelle einfügen . . . . .	23
5.4. Ergebnisskelette . . . . .	25
5.5. Speichern und Laden von Skeletten . . . . .	25
<b>6. Implementierungsdetails</b>	<b>27</b>
6.1. Programmiersprache . . . . .	27
6.2. Dateiformate . . . . .	27
6.2.1. OpenSim . . . . .	27
6.2.2. OBJ . . . . .	28
6.2.3. FBX . . . . .	28
6.2.4. Alembic . . . . .	28

6.3. Interaktivität . . . . .	29
6.4. Technische Umsetzung . . . . .	29
6.5. Transformationsmatrizen . . . . .	29
6.6. PCA . . . . .	29
6.6.1. Annotation der Bilder . . . . .	29
6.6.2. Anpassung der PCA-Ergebnisse zur Weiterverarbeitung . . . . .	31
6.7. Algorithmus zur Extremitätengenerierung . . . . .	31
<b>Literatur</b>	<b>33</b>
<b>Anhang A. Vergleichende Anatomie</b>	<b>34</b>
<b>Anhang B. D'Arcy Thompson</b>	<b>37</b>
<b>Anhang C. Erhobene Daten für PCA</b>	<b>38</b>
C.1. Skelettbilder . . . . .	38
C.2. Gewicht der Tiere . . . . .	42



# 1. Bisherige Arbeiten

## 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/>

## 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.

## 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-D8C.html>
- Möglichkeit Skelett in einen fertig modellierten Körper einzupassen.
- Skelette sind schon vorgefertigt.
- v.a. für menschliche Skelette, aber auch (limitiert) anpassbar auf Tiere

## 1.4. 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 (?)

## 1.5. No Man's Sky

- Webseite [3]
- „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.

## 1.6. Avatar movie

- 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)
- Prolemuris 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)

## 1.7. Mythologie

- Zentauren haben Körper wie ein Pferd, aber statt einem Hals setzt ein menschlicher Oberkörper auf Schultergürtel auf.  
<https://tvropes.org/pmwiki/pmwiki.php/Main/OurCentaursAreDifferent?from=Main.CentauroidForm>
- Pegasus hat den Körper eines Pferdes + Flügel, die zusätzlich zu den Vorderbeinen an Schultergürtel ansetzen  
<https://tvropes.org/pmwiki/pmwiki.php/Main/Pegasus>

## 2. Biologie

- „Wirbeltiere (Vertebrata) [...] Von vielen Zoologen wird heute der Begriff Schädeltiere (Craniota) für dieses Taxon bevorzugt. Diese Auffassung berücksichtigt, dass die Rundmäuler, wie auch einige andere Wirbeltiere, als Achsenknochenkette keine Wirbelsäule, sondern eine Chorda dorsalis haben. Doch allen Wirbeltieren gemein ist ein verknöcherter oder knorpeliger Schädel; sein Vorhandensein gehört somit zu den gemeinsam abgeleiteten Merkmalen (Synapomorphien) dieser Chordaten-Gruppe.“ (<https://de.wikipedia.org/wiki/Wirbeltiere>) → Beschränkung auf Schädeltiere mit Wirbelsäule
- „Dem Skelett der Wirbeltiere sind viele Gemeinsamkeiten ansehbar, trotzdem unterscheidet es sich, je nach Lebensraum und Anforderungen, teilweise erheblich. Mit diesen Gemeinsamkeiten und Unterschieden beschäftigt sich die Vergleichende Anatomie.“ (<https://de.wikipedia.org/wiki/Skelett#Wirbeltiere>) Notizen zu [2] siehe Anhang A.
- Das Skelett eines Wirbeltiers ist nicht unbedingt zusammenhängend.
- „Säugetiere haben in der Regel sieben Halswirbel.“ Bei Wirbeltieren kann die Anzahl aber zwischen 6 und 31 variieren. Vögel haben zwischen 10 und 31 und zwei Tiere haben 6 Wirbel. (<https://de.wikipedia.org/wiki/Halswirbel>, weitere Quelle zu Wirbelanzahl bei Säugetieren: [https://archive.org/details/bub\\_gb\\_4JFmAAAAMAAJ/page/n35/mode/2up](https://archive.org/details/bub_gb_4JFmAAAAMAAJ/page/n35/mode/2up), S.10)
  - Halswirbel: 7 oder, falls Flügel 10 bis 30
  - Brustwirbel: 15
  - Lenden- + Kreuzwirbel: 10
  - Schwanzwirbel: 5 bis 20
- Auf der Rückenwirbelsäule liegen also insgesamt 25 Wirbel. Da der „Wurzelwirbel“ in der Mitte der Rückenwirbelsäule liegt (oder zumindest ungefähr, da Bezierkurve ausgewertet bei 0,5) werden 13 vorne und 12 hinten generiert.
- Form der Wirbelsäule siehe <https://de.wikipedia.org/wiki/Wirbels%C3%A4ule>
- Schultergürtel: Das Schlüsselbein ist bei den meisten Wirbeltieren zurückgebildet, bei manchen gar nicht vorhanden. Das Rabenbein ist bei vielen Wirbeltieren zu einem Fortsatz am Schulterblatt zurückgebildet. Das Schulterblatt ist meistens vorhanden.

→ Nur das Schulterblatt wird modelliert.

<https://de.wikipedia.org/wiki/Schulterg%C3%BCrtel>

### 3. Idee

1. Erzeugung eines Wirbeltierskeletts
2. Erzeugung von Muskeln (future work)
3. Erzeugung von Haut (future work / kurz ausprobieren)
4. Erzeugung von vielen Skelettvarianten bei Eingabe eines Skeletts (nur wenn es relativ leicht möglich ist)

#### 3.1. Skelett

- 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 paramterischen L-Systemen [1] 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**) ToDo  
Außerdem ist das Skelett nicht unbedingt zusammenhängend (siehe Biologie). → Darauf achten, dass das nicht von Algorithmus verlangt wird
- Regeln wichtig, die dafür sorgen, dass das Tier am Ende auch funktional ist.
- Darstellung des Skeletts in „sinnvoller“ Pose.
- Beachte, dass Tier nicht umfällt: berechne Drehmomente um Kontaktpunkte mit Boden. Dafür sind aber Muskeln bzw. Masse wichtig → Schätzung der Masse anhand der Knochengröße bzw. Größe der Bounding Box?
- Der Zufall ist eingeschränkt durch Eingabeparameter oder Benutzereingabe (siehe Interaktivität). Ähnliche Parameter sollten zu ähnlichem Tier führen. Parameter könnten folgendes beschreiben:
  - äußere Einflüsse (Klima, Terrain, Lebensraum, . . . )
  - Proportionen (länge der Extremitäten, Kopfgröße, . . . )
  - Anzahl vorhandener Gliedmaßen / Zehen / . . .

- Ein Skelett besteht aus Knochen, Gelenken und deren (relativen) Positionen und Orientierungen.
- Ein Knochen ist im einfachsten Fall ein Zylinder (Strecke) mit Länge und Radius, kann aber auch eine konvexe Kurve mit einem Radius sein.
- Ein Gelenk hat keine Ausdehnung (?). Es ist das Verbindungsstück zwischen zwei oder mehr Knochen und legt fest wie die Knochen sich relativ zueinander bewegen können. Werden mehr als zwei Knochen verbunden ist es einfach eine feste Verbindung.
- 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.
- Köpfe sind kompliziert → Auswahl an Köpfen bereitstellen (evtl. leicht skalier-/verformbar oder ineinander überführbar)
- Brustbein sorgt dafür, dass Skelett nicht mehr baumartig → erstmal weglassen, ist wahrscheinlich auch nicht unglaublich relevant

### 3.1.1. Extremitäten

- <https://de.wikipedia.org/wiki/Extremit%C3%A4tenevolution>
- „Die paarigen Flossen von Fischen und Extremitäten von Tetrapoden sind insofern homologe Skelettelemente, als sie bei beiden an Schulter- und Beckengürtel ansetzen und die Extremitäten aus den paarigen Flossen evolutionär hervorgegangen sind.[4] Sie unterschieden sich jedoch im Knochenaufbau und in der Embryonalentwicklung, so dass ein evolutionärer Übergang aus den Einzelementen schwer erklärbar ist.“  
→ einfach ignorieren und trotzdem (erstmal) so modellieren?

## 3.2. 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

## 3.3. 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.htm.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“

## 3.4. 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

## 3.5. Datenstrukturen

### 3.5.1. Terminale

- Terminale zunächst als Bounding Box. Später werden sie durch Knochenmodelle ersetzt. Für jeden möglichen Knochen wird ein Modell gespeichert. Diese Modelle werden so skaliert, dass sie einen 1x1-Würfel ausfüllen. So kann dann später leicht die Skalierung auf diesen Würfel übertragen werden.
- Knochen: Bounding Box bzw. Skalierung in drei Raumrichtungen, Position, Orientierungen und Name bzw. ID (wird für Ersetzung benötigt).
- Knochen in Hierarchie anordnen und Position und Rotation relativ zu Elternelement angeben, da damit Erzeugung von Kindelementen einfacher → verwende homogene Koordinaten.

Bei der Erzeugung von Elementen auf der Wirbelsäule ist das nicht einfacher. Aber alle Algorithmen, die Tiere animieren brauchen eine Hierarchie, deshalb ist das wichtig. ([Quelle](#), [Beispiele](#))

**ToDo**

- Gelenke: Position, Bewegungseinschränkungen für alle drei Richtungen. In einer Hierarchie verbindet das Gelenk ein Kindelement mit seinem Elternelement. Gelenk ist hier ein Punkt, der im Koordinatensystem des Elternelements angegeben wird, und um den das Koordinatensystem des Kindelements gedreht werden kann/soll.  
Das macht es sinnvoll ein Kindelement erst zu erzeugen, wenn das dazugehörige Elternelement terminal ist. Sonst weiß man noch nicht so genau wie das Gelenk aussehen soll. Mit etwas mehr Aufwand lässt sich das aber auch im Nachhinein noch anpassen...

([Bewegungseinschränkungen für bestimmtes Gelenk bei jedem Tier gleich? ToDo](#)  
[Wie Einschränkungen erzeugen?](#)),

- Erzeugung von Kindelementen nicht nur von direktem Elternteil abhängig → Abhängigkeiten zwischen entfernten Teilen nötig; auch für Balance nützlich. „Kommunikation“ über Überobjekt, das auch alle Einzelteile speichert. (SkeletonGenerator)

### 3.5.2. „Wachstum“

- Lage der Wirbelsäule am Anfang festlegen, da sie der zentrale Teil des Körpers ist und die Lage der anderen Körperteile grob festlegt. Form variiert von Gerade (Fisch) bis geschwungener Hals und Schwanz (z.B. Giraffe). Mensch passt hier nicht ganz ins Schema → weglassen?
- Position der Wirbelsäule wird zu Beginn durch ein, oder mehrere, Beziérkurve(n) festgelegt. Außerdem wird gesteckt wo Elemente ansetzen, die nicht einfach Wirbel sind, z. B. Extremitäten. Von diesen Punkten aus, kann das Skelett dann weiter

„wachsen“. Diese Positionierung der Wirbelsäule legt die Länge und die Höhe des Tiers fest. Außerdem natürlich den Verlauf der Wirbelsäule, also z. B. auch die Länge des Halses.

Regel, die zunächst sinnvoll klingt: Ansatzpunkte der Extremitäten an der Wirbelsäule können nur in Bereichen der Kurve sein, an denen die Steigung nahe null ist.

- Wachstum unter Berücksichtigung verschiedener Randbedingungen wie Bodenposition, Anzahl Extremitäten etc. Bounding Box für Nichtterminale ist dafür aber nicht nötig.
- Die Knochen sollen eine Hierarchie (Baum) bilden. Am besten ist ein Knochen in der Nähe des Schwerpunkts (oft wird die Hüfte verwendet). Die Hüfte gibt es aber nicht immer. Verwende ersten Wirbel nach dem Brustkorb (falls existent).

### 3.5.3. Balance

Berechnung der Balance: Drehmomente der Knochen (benötigt Position der Knochen und Gewicht) → zunächst nicht nötig, da Position der Wirbelsäule zu Beginn mit Beziérkurve festgelegt wird

- Drehmoment  $\vec{M} = \vec{r} \times \vec{F}$ , mit Ortsvektor  $\vec{r}$  vom Bezugspunkt des Drehmoments zum Angriffspunkt der Kraft  $\vec{F}$ . Betrag des Drehmoments  $M = r \cdot F \cdot \sin(\alpha)$ , mit dem Winkel  $\alpha$  zwischen  $\vec{F}$  und  $\vec{r}$ .  
(<https://de.wikipedia.org/wiki/Drehmoment>)
  - Kraft, die auf Knochen wirkt: resultierende Kraft der Streckenlast, die durch zugewiesenes Gewicht erzeugt wird. Der Einfachheit halber einfach rechteckige Streckenlast annehmen, dann greift Kraft am Mittelpunkt der Strecke und  $\vec{F} = \frac{1}{2}mg^1$ , mit Masse  $m$  und Erdbeschleunigung  $g$   
(<https://www.der-wirtschaftsingenieur.de/index.php/streckenlast/>, <https://www.youtube.com/watch?v=T1Sf8GPD4dc>)
  - Reduzierung des Problems auf zwei Dimensionen, da Körper so aufgebaut ist, dass er nur nach vorne oder hinten umkippen kann und nicht zur Seite.
  - Damit Körper immer in Balance ist, muss auch nicht terminalen Teilen ein Drehmoment (und damit auch implizit ein Gewicht) zugewiesen werden. Werden aus diesen Nichtterminalen dann neue Teile generiert, so muss das Drehmoment bzw. die Summe der einzelnen Drehmomente gleich bleiben.
  - Der Bezugspunkt für die Berechnung der Drehmomente ist der Schwerpunkt des Körpers, da sich dort alle durch das Eigengewicht wirkenden Drehmomente des Körpers zu null aufzufaddieren.  
(<https://www.grund-wissen.de/physik/mechanik/drehmoment-und-gleichgewicht.html>)
- Das heißt zu Beginn der Generierung muss der Schwerpunkt des Körpers festgelegt werden, damit die Drehmomente berechnet werden können. Das ist am Anfang aber sehr einfach, da das initiale Element die Bounding Box des Gesamten Skeletts ist. Also ist der Schwerpunkt der Mittelpunkt.

---

<sup>1</sup>Die Formel zu rechteckigen Streckenlasten, die man so findet ist  $\vec{F} = \frac{1}{2}Lq_0$ , mit Länge der Strecke  $L$  und den auf der Strecke verteilten Einzellasten  $q_0$ , aber  $Lq_0 = mg$ . Daraus folgt obige Formel.

## 4. PCA

(einheitliche Benennung von PCA Dimension, erhobenen Daten etc.)

ToDo

Eine *Principal Component Analysis* (PCA) oder auch Hauptkomponentenanalyse wird auf einer Menge von Datenpunkten (oder Beispielen) mit jeweils  $n$  Dimensionen ausgeführt. Sie findet diejenigen Richtungen im  $n$ -dimensionalen Raum, in denen sich die Beispiele am meisten unterscheiden, die „principal components“. Das Ziel ist in den meisten Fällen die Dimensionalität der Daten zu reduzieren. Das wird dadurch erreicht, dass die Eingabedaten nur noch durch die „principal components“ dargestellt werden, da die anderen Dimensionen optimalerweise nur einen marginalen Einfluss auf die Daten haben.

Das Ziel im Bezug auf die Generierung von Skeletten war vor allem die Position und Krümmung der Wirbelsäule bei unterschiedlichen Tieren zu untersuchen, da diese sehr charakteristische Merkmale eines Skeletts sind. Aus dem gleichen Grund soll der Algorithmus zur Generierung der Skelette mit der Generierung der Wirbelsäule starten und daraus dann den Rest der Knochen „wachsen“ lassen. Die Hoffnung hier ist auch, dass dadurch Skelette generiert werden, die ausbalanciert wirken. Es soll nichts generiert werden, das aussieht, als würde es sofort umfallen. (Hat das funktioniert?)

ToDo

Eine PCA, die als Eingabe viele Beispiele für Wirbelsäulen und ein paar weiteren Daten zum zugehörigen Tier bekommt, ist hier hilfreich, da sie Zusammenhänge für die Positionierung der Wirbelsäule liefert.

### 4.1. Funktionsweise

(PCA Quelle) (PCA-Algorithmus mit Pseudocode und Ein- und Ausgabe als Referenz (Eingabe ist Kovarianzmatrix, Rest Vorberechnung?))

ToDo  
ToDo

Die Eingabe für eine PCA ist eine Menge von Punkten. Diese Punkte repräsentieren einzelne Instanzen dessen, was untersucht werden soll. Hier sind die Datenpunkte Skelette von Wirbeltieren. Für jede Eigenschaft des Skeletts, die untersucht werden soll, hat der Punkt eine Dimension. Gegeben ist also eine Menge von Punkten mit jeweils  $n$  Dimensionen. Die PCA nimmt an, dass die Punkte in jeder Dimension normal-/gaußverteilt sind. In diesem Fall bilden die Punkte im  $n$ -dimensionalen Raum einen Ellipsoid.

Das Ziel der PCA ist herauszufinden wo die Achsen des Ellipsoids liegen, also wie die Eingabedimensionen miteinander korreliert sind. Interessant sind dabei die Achsen in deren Richtung die Daten die große Streuung aufweisen.

Um diese Achsen zu berechnen, wird zunächst die Kovarianzmatrix aufgestellt, deren Einträge die Kovarianz zwischen den verschiedenen Achsen beschreibt. Die Eigenvektoren dieser Kovarianzmatrix sind dann die Achsen des Ellipsoids und die zugehörigen Eigenwerte geben an wie groß die Varianz in dieser Richtung ist, also wie weit ausgedehnt das Ellipsoid in dieser Richtung ist. Der Mittelpunkt des Ellipsoids ist der Mittelwert der Daten.

Will man nun herausfinden was die Haupteigenschaften eines Datenpunktes sind, stellt man ihn im neuen Koordinatensystem des Ellipsoids dar, also als gewichtete Summe der Eigenvektoren. Dann betrachtet die Dimensionen mit den größten Eigenwerten. Dazu zieht man zunächst den Mittelwert vom Datenpunkt ab und multipliziert ihn dann mit der transponierten Basiswechselmatrix, also der Matrix, in der in den Zeilen die Eigenvektoren stehen.

## 4.2. 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 stammen Zoologiebüchern, in denen sie als Beispiele für bestimmte Erklärungen angegeben waren (Bildquellen siehe Anhang C.1). 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 oder die Winkel der Gelenke an den Beinen. Auch Informationen zu sehr kleinen Knochen, wie Handwurzelknochen oder die unterschiedlichen Fingerknochen, sind schwierig zu bekommen, da sie teilweise schwierig zu erkennen und zu markieren sind. Deshalb haben wir 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.
- Die Tierklasse, also ob das Tier ein Fisch, ein Amphib, 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.
- Ob Flügel vorhanden sind („ja“ repräsentiert durch die Zahl eins und „nein“ repräsentiert durch null).
- Die Anzahl der Beine mit Bodenkontakt.
- Das ungefähre Gewicht eines ausgewachsenen Exemplars des Tieres, zu dem das Skelett gehört, in Kilogramm. Hier wurde oft das maximale Gewicht 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.2 zu finden.)

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. Details dazu wie die Daten annotiert und ausgelesen wurden sind in Abschnitt 6.6 zu finden.

Die Lage der Wirbelsäule wird durch drei kubische Bézierkurven erfasst. Jeweils einer für Hals, Rücken und Schwanz. Hals und Rücken gehen an der Schulter ineinander über und Rücken und Schwanz beim Becken. Diese Bézierkurven sind die ersten 20 Eingabedimensionen für die 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.

Zusätzlich wird jeweils durch eine Gerade im Bild die Länge des Ober- und Unterarms, der Hand, des Ober- und Unterschenkels und des Fußes eingetragen, falls vorhanden. Die Bezeichnung der Extremität als Arm oder Bein ist nur zur Unterscheidung zwischen Vorder- und Hinterextremitäten gedacht. Sie hat in keiner Weise etwas mit der Funktion der Gliedmaßen zu tun.

Zusätzlich zum Bild gibt es noch eine Textdatei, in der die anderen Daten, die oben schon erwähnt wurden, erfasst werden.

Bei der Annotation der Bilder sind folgende Schwierigkeiten aufgetreten: (**in Text ein- ToDo betten**)

- Bei Fischen ist nicht klar wo Rücken in Schwanz übergeht, da der Beckengürtel sich teilweise beim Kopf befindet oder auch gar nicht vorhanden ist. Bei der Datenerhebung wurde der Übergang ungefähr bei der Rücken- oder der Afterflosse festgelegt, da dies relativ gut zum Algorithmus passt.
- Hals und Schwanz von manchen Tieren ist mit einer kubischen Bézierkurve nicht darstellbar. Das ist unter den verwendeten Beispielen der Hals von Ichthyornis und vom Schwan und der Schwanz vom Ichthyosaurus und vom Koboldmaki. In diesem Fällen wurde versucht die Form möglichst gut anzunähern oder Fortsätze (wie am Schwanz vom Ichthyosaurus) die im Algorithmus wahrscheinlich sowieso nicht abgebildet werden, 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 irgendwie eingerollt ist.

## 4.3. Analyse der Eingabedaten

Insgesamt wurden 44 Datenpunkte erhoben. Das entspricht bei 29 Dimensionen  $44 \cdot 44 = 1276$  Zahlen als Eingabe. Das Ergebnis der PCA ist im wesentlichen die  $29 \times 29$  große

Kovarianzmatrix. Das sind 841 bzw.  $\frac{29 \cdot 30}{2} = 435$  Zahlen als Ausgabe. Da  $1276 < 435$ , ist das eine ausreichende Menge an erhobenen Beispielen. (**bessere Begründung?**)

**ToDo**

Der Mittelwert aller Eingabedaten ist in Abbildung 4.3 visualisiert. (**Werte der Mittelwertvisualisierung angeben wenn zuletzt aktualiert, damit Bild nicht so groß sein muss; und Rand abschneiden**) Wie die bei der Datenerhebung ist hier die Position der Wirbelsäule auf einem  $1000 \times 1000$  Pixel Bild gezeigt. Da von den Knochen der Beine und Arme nur die Längen erhoben wurden, sind ihre Positionen nicht realistisch. Die Ober- und Unterschenkel und die Ober- und Unterarme, sind von den Übergangspunkten Hals-Rücken und Rücken-Schwanz senkrecht nach unten gezeichnet, Hand und Fuß sind vom jeweiligen Endpunkt von Arm oder Bein horizontal nach links gezeichnet. Die restlichen Daten sind nicht visualisiert, sondern nur in Textform in der oberen linken Ecke angegeben.

Eine Voraussetzung dafür, dass die PCA korrekt funktioniert, ist, dass die Eingabedaten in jeder Dimension normalverteilt sind. Im Folgenden sollen die Eingabemerkmale daraufhin genauer untersucht werden.

Es gibt die beiden Merkmale *Flügel* und *Beine mit Bodenkontakt*, die offensichtlich nicht normalverteilt sind, da sie diskret nur ein bzw. zwei 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 liegen soll, sollen diese beiden Merkmale keinen großen Einfluss haben. Deshalb wurden diese Merkmale kleinskaliert. Dazu weiter unten mehr.

**ToDo**

Die anderen Dimensionen wurden mithilfe eines Quantil-Quantil-Diagramms mit der Normalverteilung verglichen. (**QQ Diagramme erklären, negative Werte sind ganz normal, da inverse CDF bei einem Mittelwert von 0 bei Werten < 0.5 negativ ist.**) Die Diagramm zeigen für alle Merkmale außer dem Gewicht, dass sie mehr oder weniger gut normalverteilt sind (siehe Beispiele in Abbildung 4.1 und eine vollständige Aufzählung im Anhang in den Abbildungen C.1 und C.2).

Das Merkmal *Gewicht* ist überhaupt nicht normalverteilt (siehe Abbildung 4.2 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 wieder normalverteilt (siehe Abbildung 4.2 c). Deshalb wird das Gewicht zunächst logarithmiert, 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. Eine solche Skalierung bestimmt die „Wichtigkeit“ des Merkmals für die PCA. Da das Gewicht, wie die Merkmale *Flügel* und *Beine mit Bodenkontakt* nicht die Hauptmerkmale sind, die hier untersucht werden sollen, wurde es ebenfalls kleinskaliert.

Generell bewirkt die Skalierung einer Dimension eine Gewichtung. Denn durch eine Skalierung ändert sich auch die (Co-)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 mit 1000

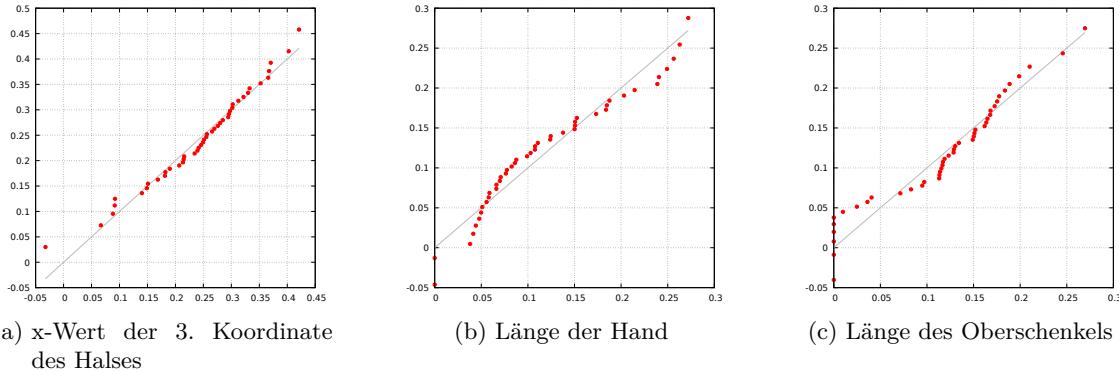


Abbildung 4.1.: Beispielhaft ausgewählte Quantil-Quantil-Diagramme von drei Eingabedimensionen. (a) weicht nicht stark von der Normalverteilung ab, (b) und (c) hingegen schon mehr, sind aber trotzdem noch akzeptabel verteilt.

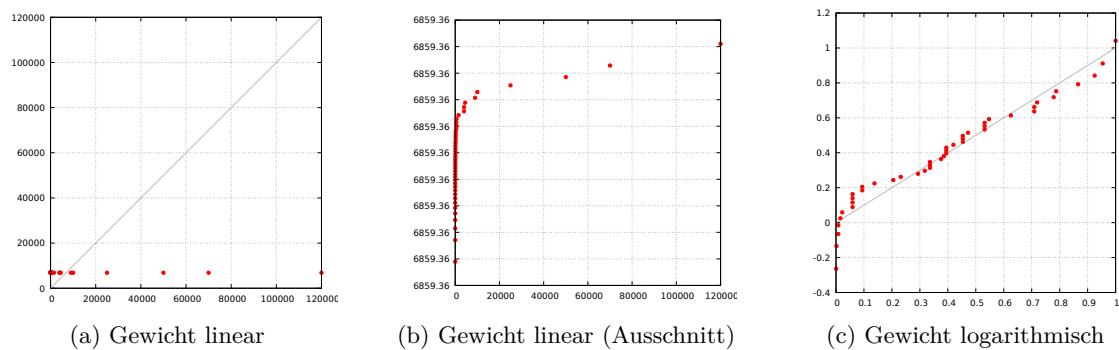


Abbildung 4.2.: Quantil-Quantil-Diagramme des Gewichts, einmal linear (a,b) und einmal mit logarithmischer Skala (b)

skaliert werden, da sie alle im Intervall  $[0, 1000]$  liegen, da sie in Pixeln dargestellt werden und das Bild eine Größe von  $1000 \times 1000$  Pixel hat. Bei Längen wären theoretisch auch Werte  $> 1000$  möglich. Solche Längen wären aber unrealistisch und werden deshalb ignoriert. Koordinaten und Längen im Bild sind diejenigen Merkmale, die uns am meisten interessieren. Deshalb sollten sie den größten Einfluss auf das Ergebnis der PCA haben. Alle anderen Merkmale wurden deshalb kleiner skaliert.

Man könnte statt einer Skalierung durch 1000 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. Wir haben uns aber für die oben beschriebene Variante entschieden, da es natürlich wirkt, dass kleine Merkmale im Bild auch weniger wichtig sind. Falls in Zukunft Gründe für eine andere Gewichtung auftauchen 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 Gewicht  $w$ :  $\frac{\log(w+1)}{\log(\max+1)}$ . Das schwerste Wirbeltier ist der Blauwal mit bis zu 120 Tonnen (siehe C.2). Deshalb ist hier  $\max = 120.000$ .

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 ersten paar Eigenvektoren 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.

Berachtet man nun die so skalierten Eingabedaten, so hat der Klippschliefer den minimalen Abstand zum Mittelwert (siehe Abbildung 4.4a). 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 Bild des Skeletts gibt. Das liegt daran, dass es keine seitlichen Abbildungen von ausgestreckten Schlangen gibt. Sie werden eigentlich immer gekrümmmt dargestellt, da sonst das Bild sehr lang und schmal werden würde. Deshalb wurde für die Schlange nur eine horizontale Gerade 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 ja keine.

Der Punkt mit dem zweitgrößten Abstand zum Mittelwert ist das Känguru (siehe Abbildung 4.4b). Zusätzlich zum Bild gibt es hier folgende Daten: *Tierklasse* Säugetier, *Flügel* nein, *Anzahl Beine mit Bodenkontakt* 2, *Gewicht* 50kg

Zuletzt betrachten wir noch die Projektion der Eingabedaten auf die ersten zwei Eigenvektoren. In Abbildung 4.5 ist noch einmal gut zu vergleichen was die Effekte der Skalierung der Eingabedaten ist. Ganz links sind die Ergebnisse zusehen, die entstehen, wenn man alle Merkmale nur auf das Intervall  $[0, 1]$  skaliert. In der Mitte geht das Gewicht nicht mehr linear, sondern logarithmisch ein und ganz rechts sind *Flügel*, *Anzahl Beine* und *Gewicht* zusätzlich kleinskaliert. Gut zu sehen ist, wie sich die Clusterbildung durch die Anpassungen verringert. (**aAnpassung der Stile der Plots**)

In Abbildung 4.6 ist noch einmal jeweils die Projektion der Daten auf die ersten beiden Eigenvektoren zu sehen. Diesmal mit unterschiedlich markiert mit den diskret erhobenen Daten. Hier sieht man z.B. schön dass alle Tiere mit Flügeln auch Vögel sind und dass fast alle Tiere mit zwei Beinen Vögel sind. Die vier Tiere, die zwei Beine haben, aber keine Vögel sind, sind die Ohrenrobbe, der Seehund, der Tyrannosaurus Rex und das Känguru.

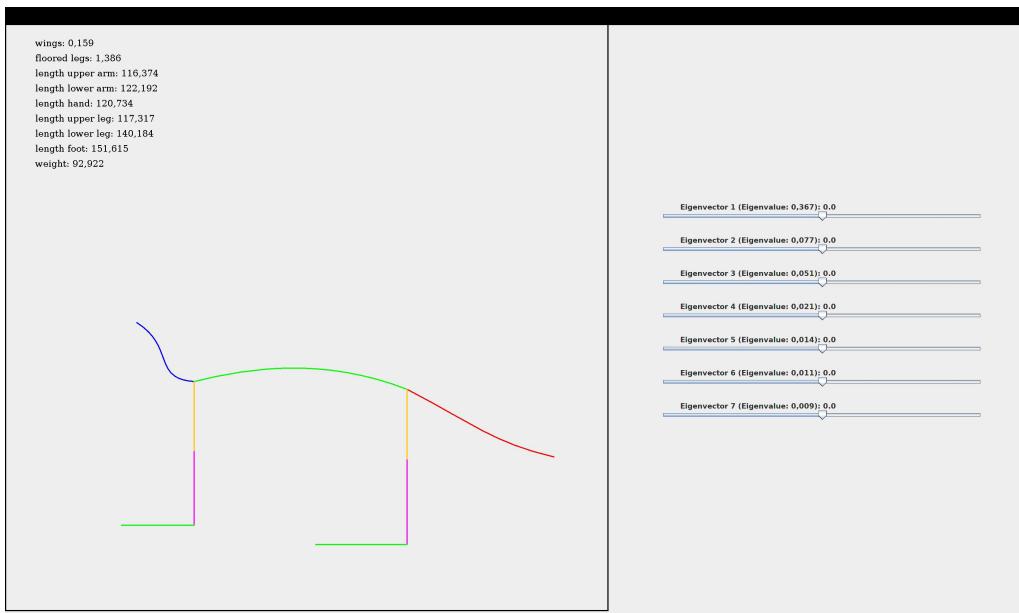


Abbildung 4.3.: Visualisierung des Mittelwerts der Eingabedaten. Die Werte, die nicht visualisiert sind, sind folgende: ...

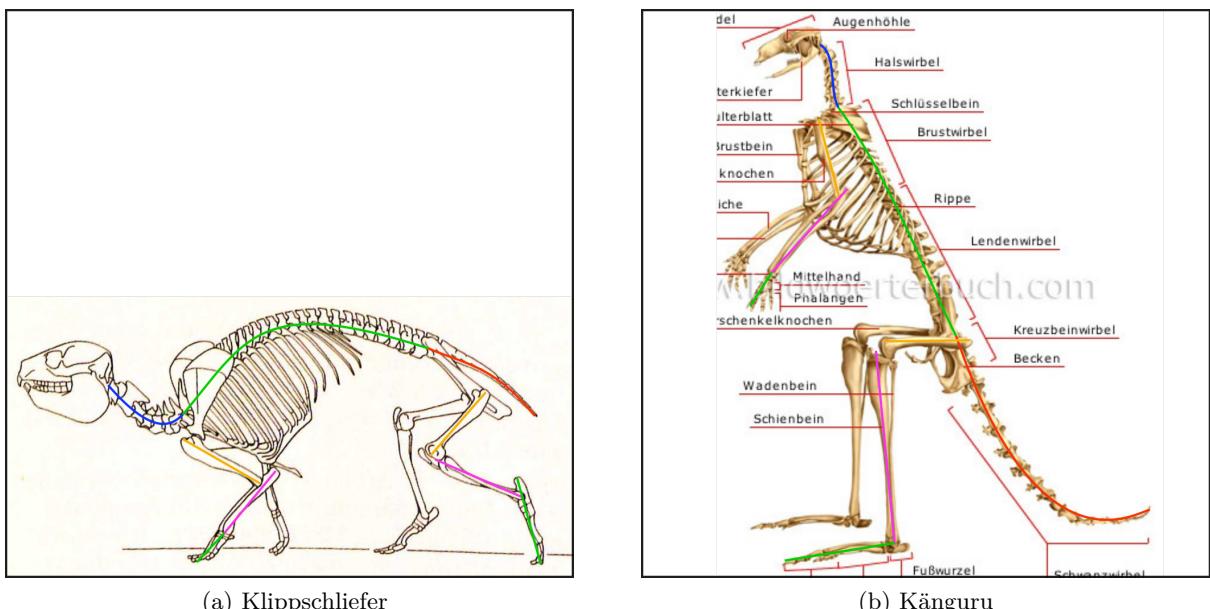


Abbildung 4.4.: Annotierte Bild des Skeletts eines Klippschließers (a) und eines Kängurus(b). Die Teile der Wirbelsäule und die Knochen der Extremitäten sind hier jeweils mit der gleichen Farbe markiert wie in der Visualisierung des Mittelwerts (Abbildung 4.3)

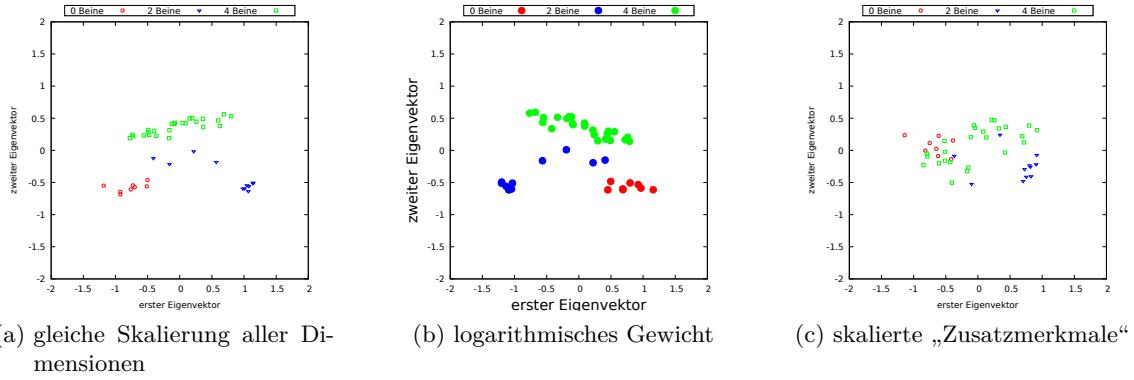


Abbildung 4.5.: Dargestellt sind hier jeweils die Projektionen der Eingabedaten auf die ersten beiden Eigenvektoren. Für jede Version wurden die Eingabedaten unterschiedlich vorverarbeitet. (a) Skalierung aller erhobenen Daten auf das Intervall  $[0, 1]$ , (b) zusätzlich Verwendung von logarithmischem Gewicht, statt linearem, (c) zusätzliche Skalierung der Merkmale *Flügel*, *Anzahl Beine* und *Gewicht* durch 100. 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.

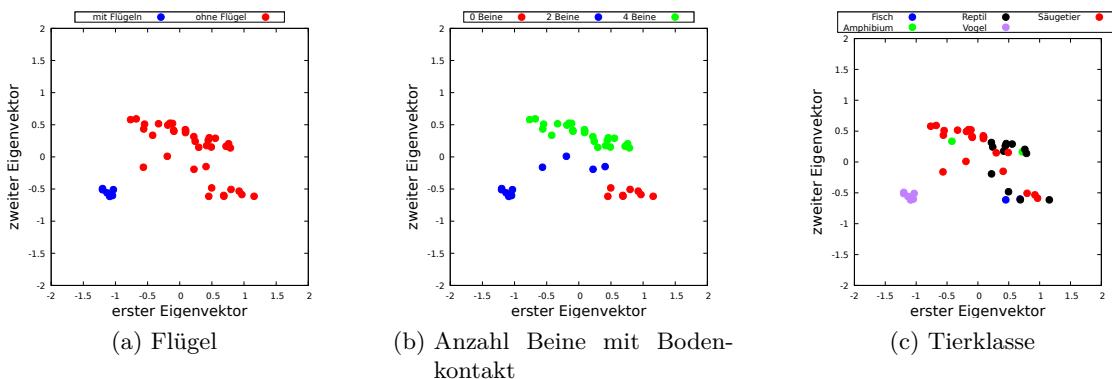


Abbildung 4.6.: Projektion der Eingabedaten auf die Ebene, die durch den ersten und zweiten Eigenvektor aufgespannt wird. Markiert sind jeweils verschiedene Merkmale der Punkte.

## 4.4. Analyse der PCA-Ergebnisse

Von 29 Eingabedimensionen gibt es auch 29 Eigenvektoren mit Eigenwerten größer 0. Der kleinste hat einen Wert von 0,000024. Von den Eigenwerten sind 7 größer als 0,01. Leider reichen diese 7 Dimensionen aber noch nicht aus, um die Eingabedaten hinreichend anzunähern. Bei manchen Tieren funktioniert das ganz gut (siehe Archaeopteryx, Abbildung 4.7), bei anderen aber eher schlechter (siehe Frosch, Abbildung 4.8). (**updaten**) Die Daten konnten somit von der PCA nicht besonders gut komprimiert werden. Trotzdem sind die berechneten Eigenvektoren hilfreich für die (Weiter-)Entwicklung des Algorithmus. (**wenn Algo weiterentwickelt, beschreiben**)

**ToDo**  
**ToDo**

In den Abbildungen, die die Eingabedaten im transformierten Koordinatensystem der PCA darstellen (Abbildung 4.6) ist gut zu erkennen, dass die Koordinatentransformation der PCA die Daten tatsächlich nach den Hauptachsen des mehrdimensionalen Ellipsoids, den die Datenpunkte bilden, ausrichtet.

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

Leider gibt es keine gute Möglichkeit die Qualität der Ergebnisse der PCA zu messen. 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, haben wir uns dafür entschieden alle Merkmale zu behalten.

Außerdem gibt es die Möglichkeit die Eingabedaten in mehrere Mengen aufzuteilen und diese von verschiedenen Instanzen der PCA analysieren zu lassen. Hierbei gibt es zunächst das Problem, dass sich dann die Anzahl der Datenpunkte noch weiter reduziert, was die Ergebnisse nicht mehr repräsentativ macht. Ein Merkmal, das sich zur Unterteilung in Mengen anbietet würde, ist die Angabe ob das Tier Flügel hat oder nicht, da sich dadurch nur zwei Gruppen ergeben. Außerdem ist die Gruppe der Tiere mit Flügeln in den Eingabedaten bei anderen Skalierungen (siehe Abbildung 4.5) klar als Cluster zu erkennen (siehe auch Abschnitt 4.3). Tatsächlich liefert eine solche Aufteilung bessere Rekonstruktionen aus den größten Eigenvektoren, aber das liegt natürlich vor allem daran, dass die zu untersuchende Datenmenge, jeweils verkleinert wurde.

Ein Problem dabei ist, dass dann keine Skelette mehr erzeugt werden können, die zwischen den beiden Gruppen liegen. Tatsächlich scheinen die Datenpunkte, die „zwischen“ den Gruppen erzeugt werden, relativ sinnvoll auszusehen. Das wäre ein Argument dafür keine Aufteilung vorzunehmen. Dasselbe gilt für die Cluster, die durch die Aufteilung anhand der Anzahl der Beine mit Bodenkontakt entstehen.

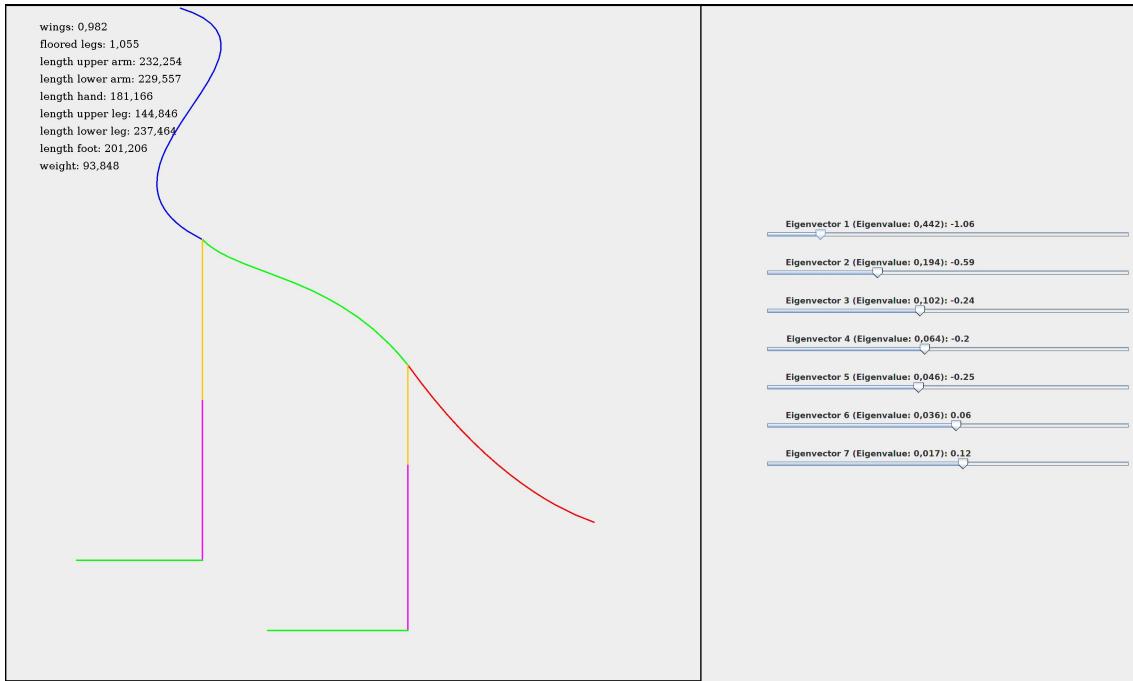
(**Schaubild mit verschiedenen erzeugten Wirbelsäulen?**)

**ToDo**

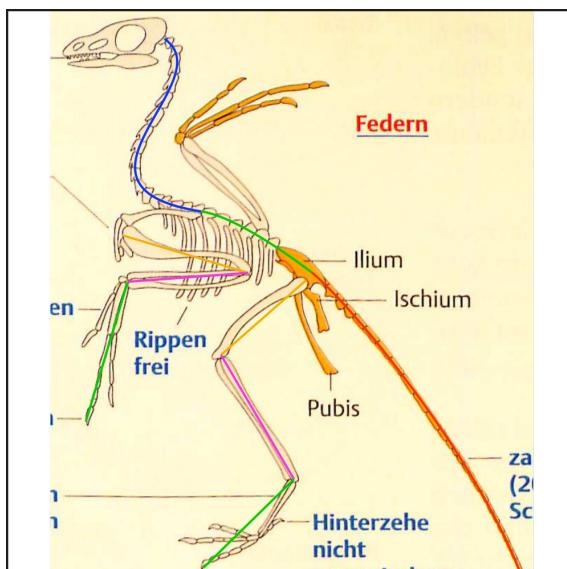
## 4.5. Bedingte Verteilung der PCA Eingabedaten

Aus den Ergebnissen der PCA lassen sich sehr gut zufällige Skelette erzeugen. Es ist aber schwer gezielt Eigenschaften festzulegen. Der erste Schritt ist Eigenschaften auf gewählte Werte festzulegen, die so schon in den erhobenen Daten vorkommen. 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 bedingten Verteilungen bilden. Dazu muss der bedingte Mittelwert und die bedingte Kovarianzmatrix bestimmt werden.



(a) Rekonstruktion

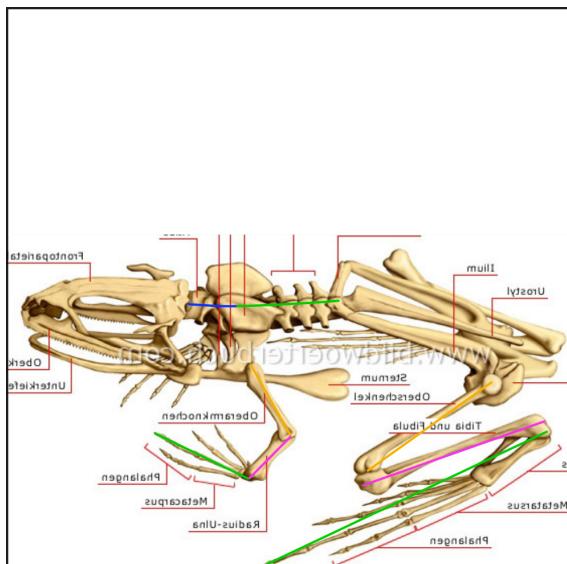


(b) Eingabebild

Abbildung 4.7.: Archaeopteryx (a) Rekonstruktion aus den größten 7 Eigenvektoren, (b) Eingabebild, zusätzlich erhobene Daten sind: Tierklasse: Vogel, Flügel, Paare von Beinen mit Bodenkontakt: 1, ungefähres Gewicht: 1kg



### (a) Rekonstruktion



(b) Eingabebild

Abbildung 4.8.: Frosch (a) Rekonstruktion aus den größten 7 Eigenvektoren, (b) Eingabebild, zusätzlich erhobene Daten sind: Tierklasse: Amphib, keine Flügel, Paare von Beinen mit Bodenkontakt: 2, ungefähres Gewicht: 0,01kg

**ToDo** (**Berechnung beschreiben: [https://en.wikipedia.org/wiki/Multivariate\\_normal\\_distribution#Conditional\\_distributions](https://en.wikipedia.org/wiki/Multivariate_normal_distribution#Conditional_distributions)**) Verwendet man nun die bedingte Kovarianzmatrix als Eingabe für die PCA, so erhält man nur noch Daten für Skelette mit den vorher festgelegten Bedingungen. 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 ganz neu ausgerechnet werden. (**ist das relevant für mein Benutzerinterface?**)

Betrachtet man erzeugte Beispiele aus bedingten Verteilungen, zeigt sich sehr deutlich, dass das Festlegen einer Dimension auch die anderen Dimensionen sehr stark einschränkt. 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 deutlich daran zu erkennen, dass der Verlauf der Wirbelsäule immer sehr ähnlich ist.

**ToDo** Um das etwas zu umgehen, wird die Eingabe als Intervall aufgefasst, aus dem zufällig ein Wert gezogen wird. Legt z. B. der Benutzer einen ganzzahligen Wert fest, so wird ein kleiner zufälliger Wert aufaddiert oder abgezogen. (**welches Intervall?**) So wird nicht immer genau der gleiche Wert verwendet.

**ToDo** Schaut man sich nun die erzeugten Skelette mit zwei tragenden Beinen an, so sind darunter nicht nur Vögel, sondern auch Skelette, die kurze Extremitäten haben und deren Wirbelsäule sich nahe über dem Boden befindet. (**Beispielbild**) Das liegt daran, dass es einen Übergang von bodennahen Vierbeinern wie Krokodilen oder Fröschen zu Fischen geben muss. Da es hier aber in der Natur eher wenige Beispiele für Zweibeiner gibt (in den erhobenen Daten sind das nur der Seehund und die Ohrenrobbe), wirkten diese Tiere eher ungewöhnlich. Da das Problem hier aber inhärent in der Vorgehensweise der PCA ist, lässt es sich nicht wirklich beheben.

Nun möchte man vielleicht Bedingungen an das zu generierende Skelett stellen, die nicht schon genau so in den erhobenen Daten repräsentiert sind. Das könnten beispielsweise die Schwanz- oder Halslänge sein.

Sind es Bedingungen an Dimensionen, die nicht erhoben wurden, können sie 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.

Für Bedingungen, die schon implizit in den erhobenen Daten enthalten sind, könnte man es ebenso machen und einfach eine zusätzliche Dimension einfügen, die aus schon bestehenden Dimensionen errechnet wird. Dabei wäre aber das Problem, dass nur die Dimensionalität erhöht wird, nicht aber die eingegebenen Daten. Es wären also mehr Eingabebeispiele nötig, nur dafür, dass die PCA eine offensichtliche Korrelation der Dimensionen erkennt.

Die bessere Alternative ist es 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 Beziérkurve 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.

**ToDo** Betrachtet man nun konkret den Wunsch die Schwanzlänge festzulegen, ist das nicht ganz so einfach. Die Länge in x- und y-Richtung lässt sich zwar leicht festlegen, diese Länge sagen aber noch nicht unbedingt viel über die tatsächliche Länge der Beziérkurve aus. Verlangt man z. B. eine Länge von null für den Schwanz in x-Richtung, so haben die generierten Skelette einen Schwanz der zwar auf gleicher Höhe beginnt und endet, aber trotzdem vorhanden ist und einen kleinen Bogen beschreibt. (**Beispielbild**) 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.

## 5. Algorithmus zur Skelettgenerierung

### 5.1. Generierung der einzelnen Extremitäten

Extremitäten werden zunächst danach unterschieden, ob sie Bodenkontakt haben sollen oder nicht. Wenn nicht, dann sind es entweder Flossen, Arme oder Flügel:

- Flossen: gerade nach hinten (orientiert an Welt-x-Achse)
- Arme: Oberarm gerade nach unten (orientiert an negativer Welt-y-Achse), Unterarm im 90° Winkel nach vorne, Hand verlängert Unterarm
- Flügel: spezielles Winkelintervall für jedes beteiligte Gelenk, daraus jeweils zufällig gewählte Winkel

Bei Extremitäten mit Bodenkontakt wird iterativ vorgegangen. Der allgemeinste Ansatz wäre hier inverse Kinematik zu verwenden. Das ist hier aber nicht nötig, da in jedem Schritt klar ist, wie die Winkel verändert werden müssen, dass der Endpunkt näher zum Boden kommt.

**To Do** (**Absatz über IK, lcp (linear complementary problem) (nicht Hauptaugenmerk / Ziel der Arbeit ist etwas anderes / reicht für Proof of Concept, bei Animationen muss große Maschinerie sowieso nochmal angeworfen werden)**)

**To Do** (**Absatz über Darstellung der Gelenke (auch mit zwei Freiheitsgraden), Problematik mit lokalen Winkelconstraints vs. globalen Berechnungen für Abstand zum Boden**) Von der Startposition aus, werden die Winkel an den Gelenken in jedem Schritt jeweils so vergrößert oder verkleinert, dass sich der Punkt, der zum Schluss den Boden berühren soll, sich der Bodenoberfläche nähert. Ob die Winkel jeweils vergößert oder verkleinert werden sollen, wird bestimmt, indem die Ausrichtung des Kindelements mit der y-Achse des Weltkoordinatensystems verglichen wird. Soll der Endpunkt der Extremität dem Boden nähern, so wird der Winkel so verändert, dass die Ausrichtung des Kindelements sich der Senkrechten nähert. Wenn nicht, so wird der Winkel in die entgegengesetzte Richtung verändert. (**das passiert aber nicht**)

**To Do** Je nach Ausgangsposition sieht das Ergebnis aber nicht unbedingt natürlich aus. Zum Beispiel kann es passieren, dass das Fußgelenk nicht gedreht wird, also der Fuß das Schienbein einfach verlängert und die Spitze des Fußes Bodenkontakt hat. Wenn dann die Oberseite des Fußes näher am Boden ist als die Unterseite, dann ist das keine sinnvolle Position. Um so etwas zu verhindern, wird die Startposition der Extremität so gewählt, dass alle Gelenke stark angewinkelt sind. (**Abbildung**) Außerdem wird während der Iteration verboten, dass Knochen unterhalb der Bodenhöhe enden. (**Zweiter Freiheitsgrad an**

**Hüft- und Schultergelenk macht es schwieriger, da Winkel, je nach Winkel der anderen Richtung, vergrößert oder verkleinert werden muss um den Boden zu erreichen. Wegelassen. Bei mehr Anforderungen doch IK verwenden.)**

In jedem Schritt werden die Winkel um eine bestimmte Gradzahl verändert. Diese Gradzahl verkleinert sich mit jedem Schritt bis zu einer Minimalgröße. Zu Beginn werden die Winkel stark verändert um die grobe Ausrichtung des Beines festzulegen und in den kleiner werdenen Schritten wird die Extremität genauer ausgerichtet, so dass der Endpunkt zum Schluss auf dem Boden steht. Zusätzlich wird nicht in jedem Schritt jeder Freiheitsgrad jedes Gelenks verändert. Für jeden Freiheitsgrad wird eine Wahrscheinlichkeit (kleiner als eins) festgelegt, dass dieser ausgewählt wird. Dadurch können bestimmte Richtungen oder Gelenke priorisiert werden um ein besseres Ergebnis zu erzielen. (**Was sind die guten Einstellungen? bzw braucht man diese Wkten überhaupt? (Konkrete Einstellungen erwähnen (in Implementierungsdetails?)**)

To Do  
To Do

## 5.2. Ansatzpunkte für Extremitäten

Ansatzpunkte für Extremitäten sind natürlich zunächst der Hüftgürtel und der Schultergürtel. Um auch die Generierung fantastischer Tiere 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 einfach an der Hüfte bzw. der Schulter mehrere Gelenke mit ein wenig Abstand angelegt, an denen Extremitäten ansetzen können. 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.

Eine Überlegung war auch zwischen Schulter und Hüfte weitere Extremitätengürtel anzubringen. Das stellt sich aber als schwierig heraus. Die Wirbelsäule ist zwischen Hüfte und Schulter 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.

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 keine alle Arten von Extremitäten außer Beinen ansetzen. Das wirkt tatsächlich meist auch anatomisch einigermaßen sinnvoll.

## 5.3. Knochenmodelle einfügen

Zunächst wird jeder terminale Knochen durch seine Bounding Box dargestellt. 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:

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**)

- 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.

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.)

**(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 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  $45^\circ$  ist es eine menschliche Hand, sonst ein Huf). (ggf. **an Implementierung anpassen**)

ToDo

ToDo

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.

## 5.4. Ergebnisskelette

- 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

## 5.5. 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.

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

**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)

# 6. Implementierungsdetails

## 6.1. 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

## 6.2. 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
- 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 verwendeteter 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 raus schreiben. 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.).

### 6.2.1. 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>)

### 6.2.2. 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

### 6.2.3. 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)

### 6.2.4. 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 (?)

## 6.3. Interaktivität

- OpenGL
  - SDL + OpenGL Tutorials  
<http://headerphile.com/sdl2/opengl-part-1-sdl-opengl-awesome/>,  
<http://www.sdl tutorials.com/sdl-opengl-tutorial-basics>
  - Daten direkt mit OpenGL erzeugen (laden als vertex und index array)
- Benutzeroberfläche
  - imgui (opengl/vulcan/3D view integriert) mit Rust oder C++: <https://github.com/ocornut/imgui>
    - \* OpenGL und ImGui für Rust: <https://nercury.github.io/rust/opengl/tutorial/2018/02/08/opengl-in-rust-from-scratch-00-setup.html>,  
<https://github.com/michaelfairley/rust-imgui-sdl2>
    - \* es gibt Java Bindings (<https://github.com/ice1000/jimgui>), aber Swing ist wahrscheinlich einfacher
    - \* OpenGL scene → imgui: <https://gamedev.stackexchange.com/questions/140693/how-can-i-render-an-opengl-scene-into-an-imgui-window>
  - Java Swing Bibliothek und JOGL (Java OpenGL Binding) (<http://www.jogl.info>)

## 6.4. Technische Umsetzung

- Repräsentation des Zustands als Hierarchie von einzelnen Komponenten (terminale sowie nichtterminale).
- Übersetzung in ein 3D-Modell: zunächst .obj, später Verwendung von OpenGL mit vertex shadern etc.
- linearer Kongruenzgenerator reich für Zufallszahlen aus, da nur wenige erzeugt werden (erkennbare Muster entstehen erst bei mehr Zufallszahlen) (erwähnen?)
- 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?)**

## 6.5. Transformationsmatrizen

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:

## 6.6. PCA

### 6.6.1. 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

---

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

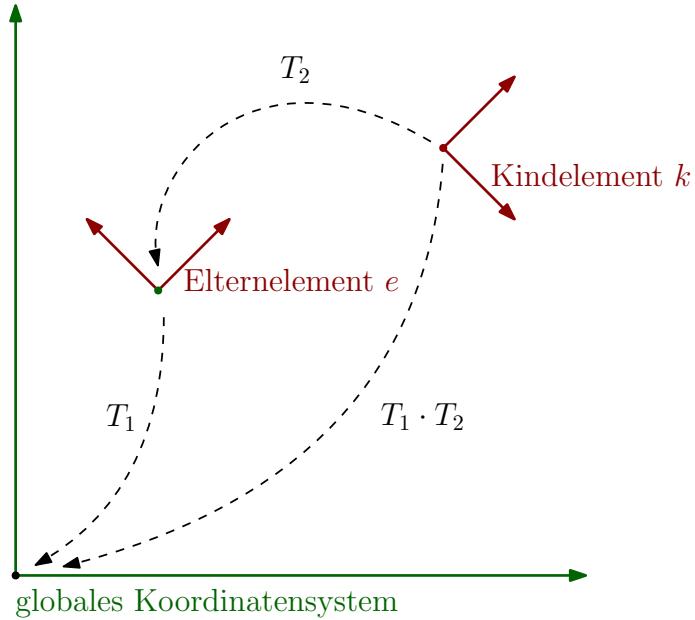


Abbildung 6.1.: Gegeben sei das Element  $e$ . Die Abbildung, die die lokalen Koordinaten von  $e$  in globale Koordinaten umrechnet sei  $T_1$ . Jedes Kindelement  $k$  von  $e$  speichert eine Transformationsmatrix  $T_2$ , die angibt wo der Ursprung des Koordinatensystems von  $k$  relativ zum Koordinatensystem von  $e$  liegt. Will man nun Koordinaten von  $k$  in globale Koordinaten umrechnen, benötigt man die Abbildung  $T_1 \cdot T_2$ .

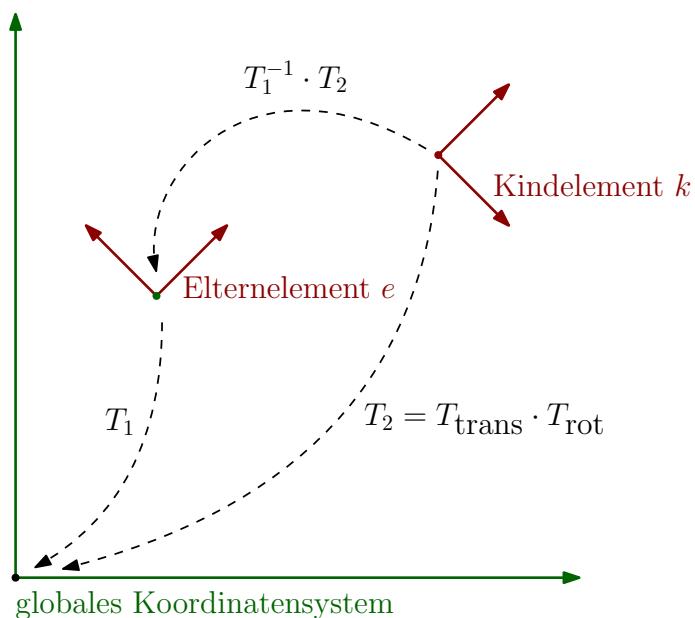


Abbildung 6.2.: Will man ein Element  $k$  erzeugen, das Kindelement von  $e$  ist und dessen globale Koordinaten bekannt sind, muss man die Abbildung berechnen, die die relative Position von  $k$  angibt. Seien  $T_1$  und  $T_2$  jeweils die Transformationen in das globale Koordinatensystem von  $e$  bzw.  $k$ . Dann ist die gesuchte Abbildung  $T_1^{-1} \cdot T_2$ .

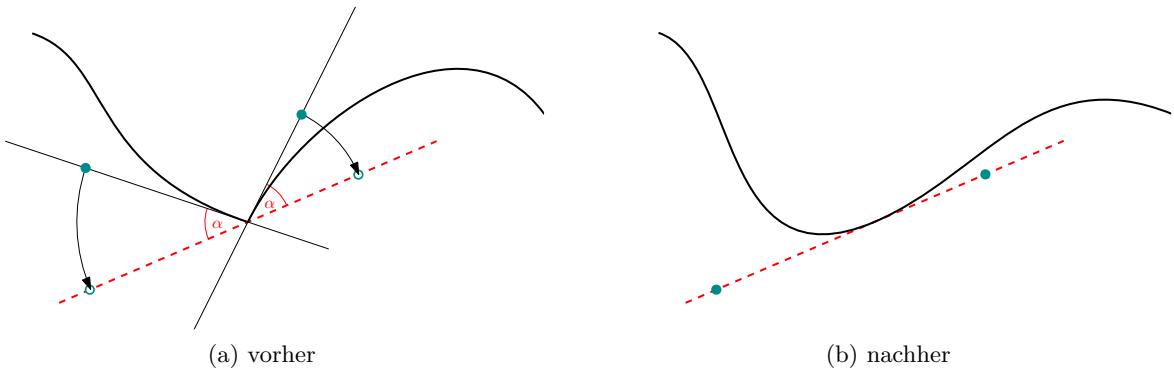


Abbildung 6.3.: Anpassung der Kontrollpunkte der Wirbelsäulenteile, wenn die Steigung an den Kontaktunkten ungleich ist. Die beiden Teile der Wirbelsäule und die Steigung am Kontaktunkt sind hier in schwarz dargestellt. Die zu drehenden Kontrollpunkte in cyan. In rot ist die resultierende Steigung und der Winkel, der für die Drehung verwendet wird, zu sehen.

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.

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

- An den Kontaktpunkten 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 Kontaktpunkt verschoben werden. Das wurde hier so gemacht, dass die zu verschiebenden Kontrollpunkte um den Kontaktpunkt in gegensätzliche Richtungen rotiert wurden. Beide werden um den gleichen Winkel rotiert (siehe Abbildung 6.3). Grundsätzlich könnten die Kontrollpunkte beliebig auf der Tangente des Kontaktpunkts 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.

## 6.7. Algorithmus zur Extremitätengenerierung

Wie in Abschnitt 5.1 bereits beschrieben, ist die Extremität in der initialen Position für den Algorithmus möglichst stark angewinkelt. Die Gelenke beginnen also mit ihren kleinst- bzw. größtmöglichen Winkeln. In den Iterationen des Algorithmus wird dann derjenige Endpunkt der Extremität dem Boden genähert, der zum Schluss Bodenkontakt haben

soll. Die Winkel der Gelenke werden also im Wesentlichen nur in eine Richtung verändert. (Würden die Gelenke nicht in der extremsten Position starten, wären diese Winkel nie wieder erreichbar.) Wenn sich der Endpunkt eines Knochens durch das Drehen unter die Bodenhöhe verschieben würde, so wird jene Drehung verworfen. In der nächsten Iteration könnte es wieder möglich sein, da eine kleinere Drehung verwendet wird.

In jeder Iteration werden die Winkel um eine bestimmte Gradzahl verändert. Zu Beginn sind das  $30^\circ$ . In den nächsten Iterationen sind es dann jeweils  $\frac{4}{5}$  davon. Falls sich der Abstand zum Boden dadurch aber kaum verändert (konkret: weniger als 0.1), liegt die Vermutung nahe, dass die Gradzahl zu groß ist und deshalb alle möglichen Winkeländerungen zu einer Positionen führen würden, in der mindestens ein Knochen unterhalb der Bodenhöhe läge. Deshalb wird in diesem Fall die Gradzahl für die nächste Iteration halbiert.

Außerdem ist es möglich für jedes Gelenk und jeden Winkel eine Wahrscheinlichkeit festzulegen mit der jener Winkel verändert wird. Das wird aber tatsächlich nur für den Winkel des Oberarms/-schenkels gemacht, der die Abspreizung der Extremität in z-Richtung festlegt. Würde man hier die Wahrscheinlichkeit im Vergleich zu den anderen nicht verringern, hätten die meisten Tiere stark abgespreizte Beine. Da dies oft eher unnatürlich wirkt und auch in der Natur nicht bei dem Großteil der Tiere vorkommt, ist hier die Anpassung der Wahrscheinlichkeit sinnvoll.

Um für jedes Gelenk die Drehrichtung zu bestimmen, die den Endpunkt näher Richtung Boden bringt, wird die negative lokale y-Achse in Weltkoordinaten betrachtet und mit der Welt-y-Achse verglichen. (**für beide Gelenktypen Schaubilder einfügen**) Dies funktioniert allerdings nur, wenn die Gelenke nicht zu weit drehen, also über die Welt-y-Achse hinaus. Zusätzlich ergibt sich auch keine sinnvolle Position des Knochens, wenn die Gelenke so weit gedreht werden. Deshalb wird auch das verhindert. Das funktioniert, indem Gelenke, die zu weit gedreht wurden, auf eine Position zurückgesetzt werden, die gerade noch erlaubt ist.

**To Do**

# Literatur

- [1] James Scott Hanan. "Parametric L-systems and Their Application to the Modelling and Visualization of Plants". AAINN83871. Diss. 1992. ISBN: 0-315-83871-X.
- [2] Milton Hildebrand und George E. Goslow. *Vergleichende und funktionelle Anatomie der Wirbeltiere*. Springer-Verlag Berlin Heidelberg New York, 2004.
- [3] *No Man's Sky*. URL: <https://www.nomanssky.com/> (besucht am 08.10.2019).
- [4] Lennart Olsson und Uwe Hoßfeld. "Homology, Genes, and Evolutionary Innovation.—Günter P. Wagner." In: *Systematic Biology* 64.2 (Dez. 2014), S. 365–367. ISSN: 1063-5157. DOI: 10.1093/sysbio/syu127. eprint: <http://oup.prod.sis.lan/sysbio/article-pdf/64/2/365/24587311/syu127.pdf>. URL: <https://doi.org/10.1093/sysbio/syu127>.
- [5] 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.
- [6] Rüdiger Wehner und Walter Gehring. *Zoologie*. 24. Aufl. 2007. ISBN: 978-3-13-367424-9.
- [7] Rüdiger Wehner und Walter Gehring. *Zoologie*. 25. Aufl. 2013. ISBN: 978-3-13-367425-6.
- [8] Wilfried Westheide und Reinhard Rieger. *Spezielle Zoologie. Teil 2: Wirbel- oder Schädeltiere*. 2004. ISBN: 3-8274-0307-3.

# A. Vergleichende Anatomie

Notizen aus „Vergleichende und funktionelle Anatomie der Wirbeltiere“ [2]:

## 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)

## 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 [5]:

### 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

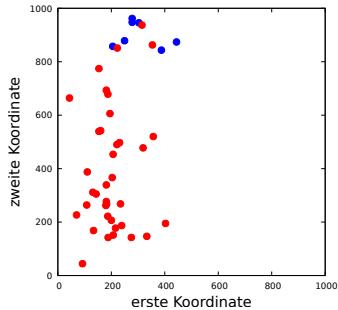
### C.1. Skelettbilder

**ToDo**

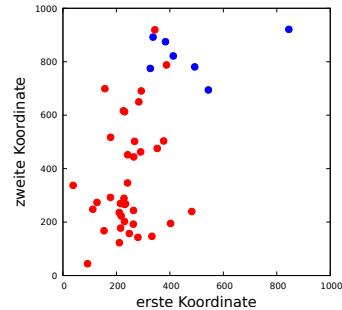
(alphabetisch sortieren)

alle Bilder aus [8] außer:

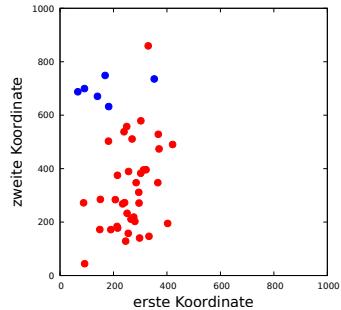
- Sinornis und Taube aus [2]
- Seekuh aus [7]
- Archaeopteryx, Eusthenopteron, Ichthyosaurus, Ichthyostega, Muraenosaurus, Ur-pferdchen aus [6]
- Pferd: <https://www.kosmos.de/content/buecher/ratgeber/pferde-reiten/vorwaerts-abwaerts>
- Känguru <http://www.bildwoerterbuch.com/tierreich/beuteltiere/kaenguru-skelett-eines-kaengurus.php>
- 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)
- Dromedär [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/File:Giraffe\\_skeleton.jpg](https://de.wikipedia.org/wiki/Giraffen#/media/File:Giraffe_skeleton.jpg)



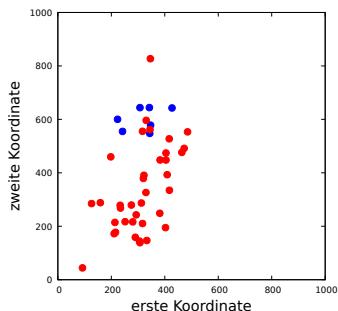
(a) Erster Punkt des Halses



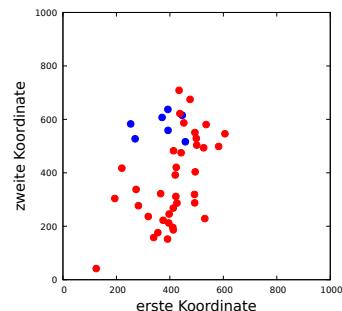
(b) Zweiter Punkt des Halses



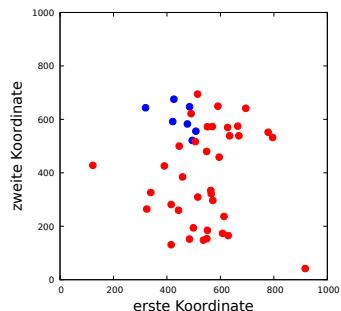
(c) Dritter Punkt des Halses



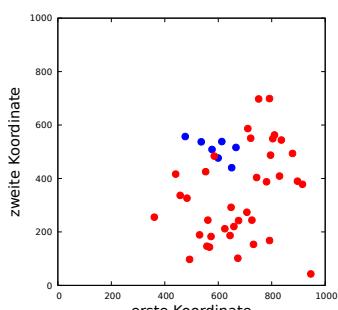
(d) Vierter Punkt des Halses bzw. erster Punkt des Rückens



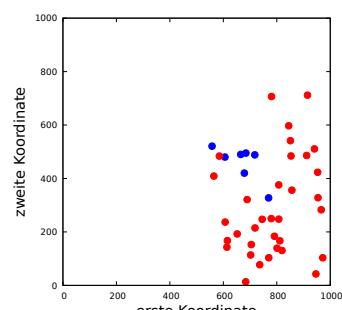
(e) Zweiter Punkt des Rückens



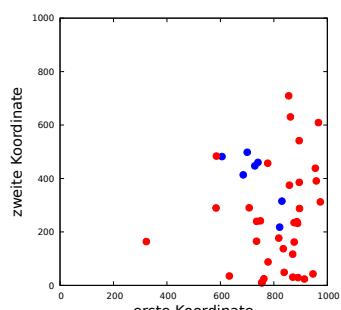
(f) Dritter Punkt des Rückens



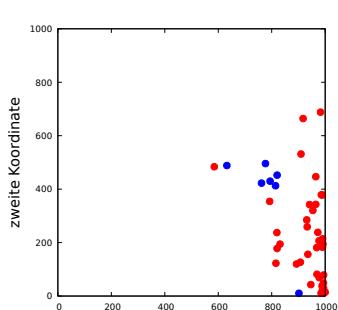
(g) Vierter Punkt des Rückens bzw. erster Punkt des Schwanzes



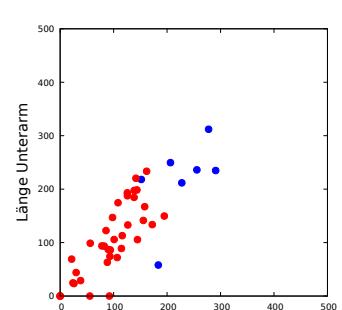
(h) Zweiter Punkt des Schwanzes



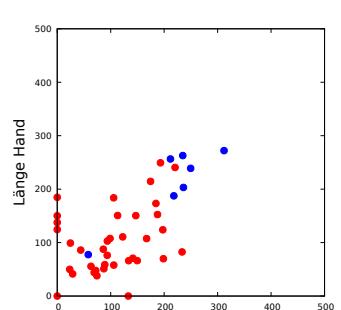
(i) Dritter Punkt des Schwanzes



(j) Vierter Punkt des Schwanzes



(k) Länge Ober- und Unterarm



(l) Länge Unterarm und Hand

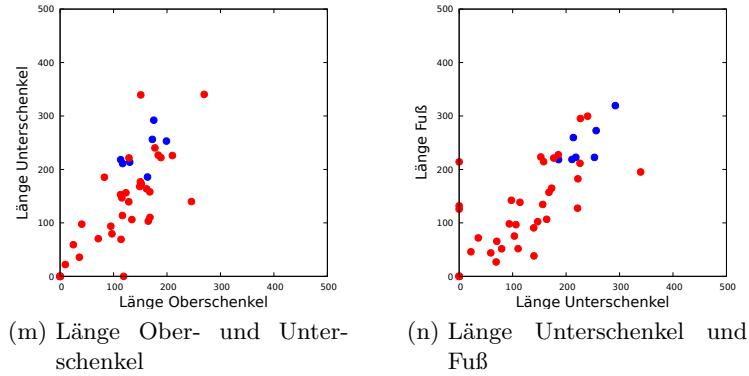


Abbildung C.0.: Erhobene Daten: Punkte der Bézierkurven der Wirbelsäule und Längen der Extremitäten. Bei den Extremitäten ist jeweils gegeneinander abgetragen Ober- und Unterarm, Ober- und Unterschenkel, Unterarm und Hand, Unterschenkel und Fuß. Markiert ist jeweils ob die Datenpunkte Flügel haben (blau)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

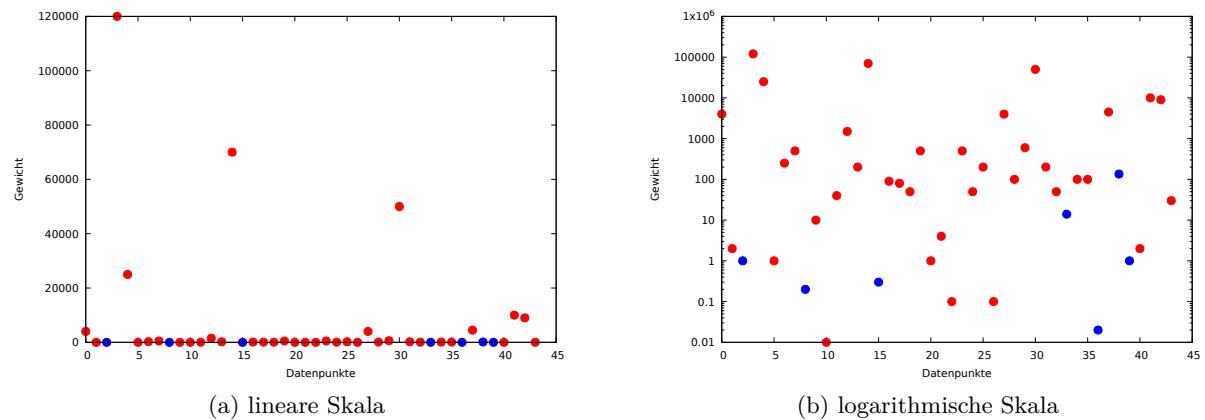
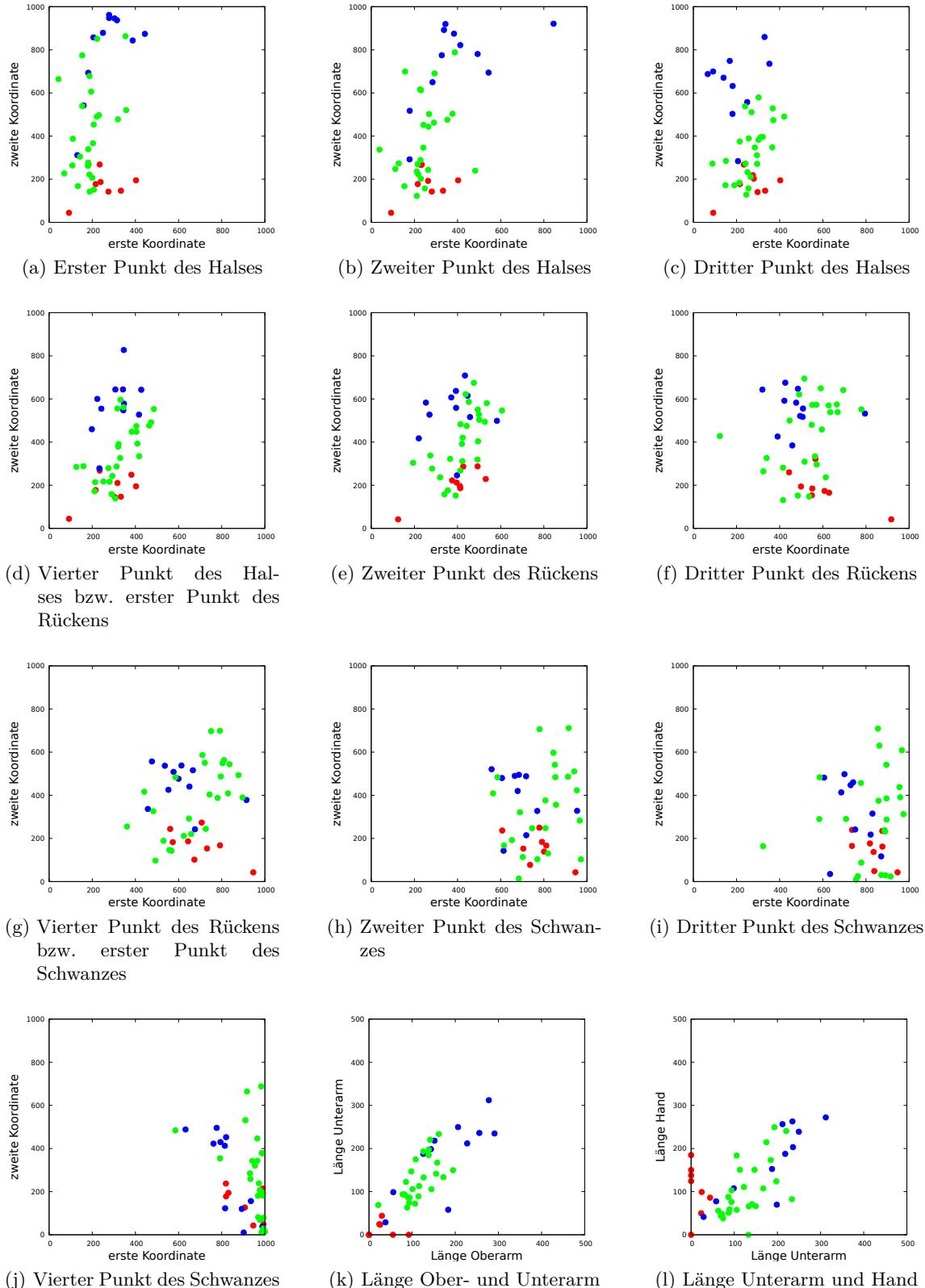


Abbildung C.1.: Erhobene Daten: Gewicht. Markiert ist jeweils ob die Datenpunkte Flügel haben (blau) oder nicht (rot).



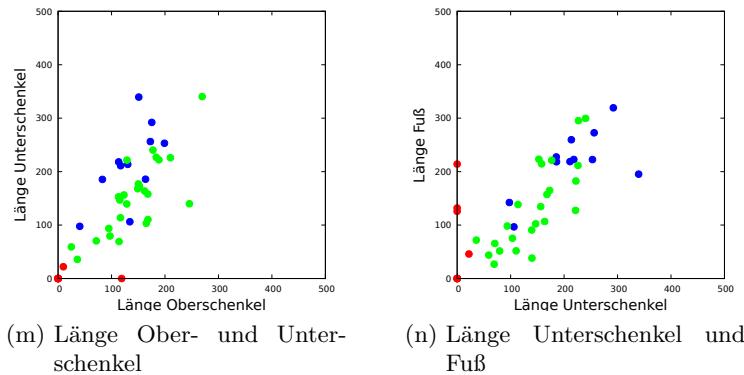


Abbildung C.1.: Erhobene Daten: Punkte der Bézierkurven der Wirbelsäule und Längen der Extremitäten. Bei den Extremitäten ist jeweils gegeneinander abgetragen Ober- und Unterarm, Ober- und Unterschenkel, Unterarm und Hand, Unterschenkel und Fuß. Markiert ist jeweils ob die Datenpunkte 0 (rot), 2 (blau)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

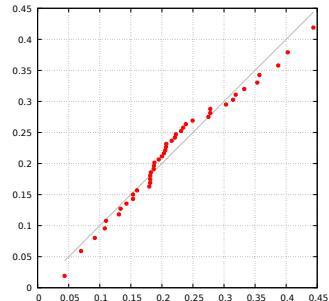
- Schlange: zu Schlangen gibt es keine Bilder, die deren Skelett in ausgestrecktem Zustand von der Seite darstellen. Deshalb wurde ein leeres Bild genommen und der Verlauf des Rückens durch eine Gerade angenähert (Extremitäten besitzt eine Schlange nicht). Außerdem ist nicht erkennbar ob bzw. wo Hals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

## C.2. Gewicht der Tiere

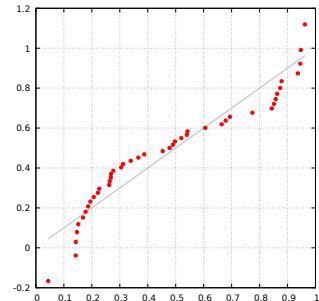
**ToDo**

### (alphabetisch sortieren)

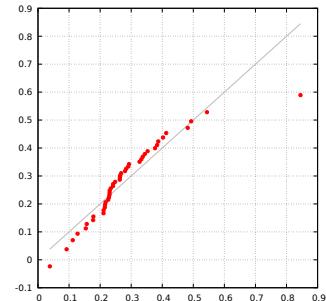
- Blauwal 120 Tonnen, <http://tierdoku.com/index.php?title=Blauwal>, „das schwerste bekannte Tier der Erdgeschichte“ <https://de.wikipedia.org/wiki/Blauwal>
- Durschnittsgewicht (Warmblut-)Pferd 600 kg, <https://www.reitarena.com/de/blog/blog-post/2015/03/03/das-pferd-grundlegende-fakten.html>
- Afrikanischer 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>
- 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/>



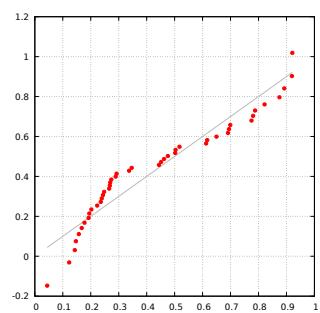
(a) Erster Punkt des Halses, x-Koordinate



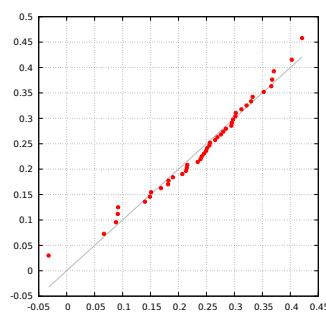
(b) Erster Punkt des Halses, y-Koordinate



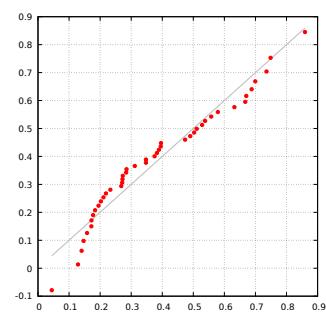
(c) Zweiter Punkt des Halses, x-Koordinate



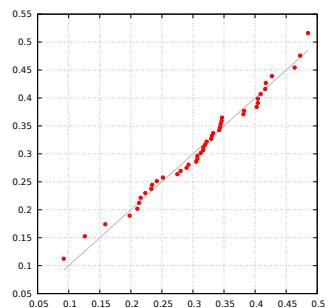
(d) Zweiter Punkt des Halses, y-Koordinate



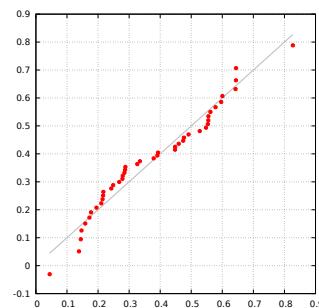
(e) Dritter Punkt des Halses, x-Koordinate



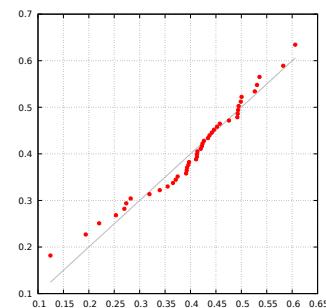
(f) Dritter Punkt des Hales, y-Koordinate



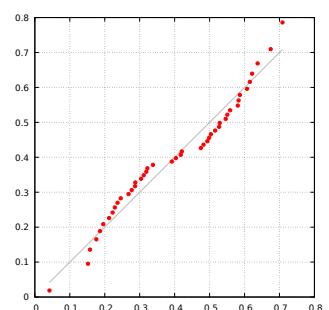
(g) Erster Punkt des Rückens, x-Koordinate



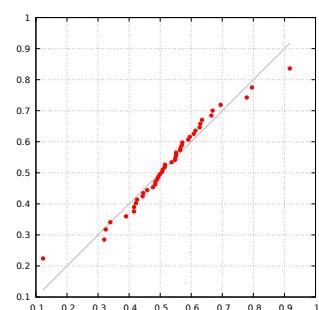
(h) Erster Punkt des Rückens, y-Koordinate



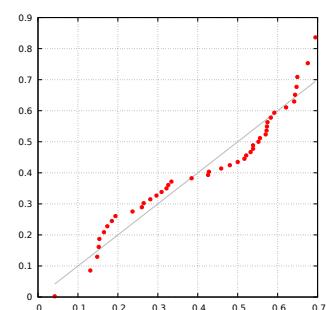
(i) Zweiter Punkt des Rückens, x-Koordinate



(j) Zweiter Punkt des Rückens, y-Koordinate



(k) Dritter Punkt des Rückens, x-Koordinate



(l) Dritter Punkt des Rückens, y-Koordinate

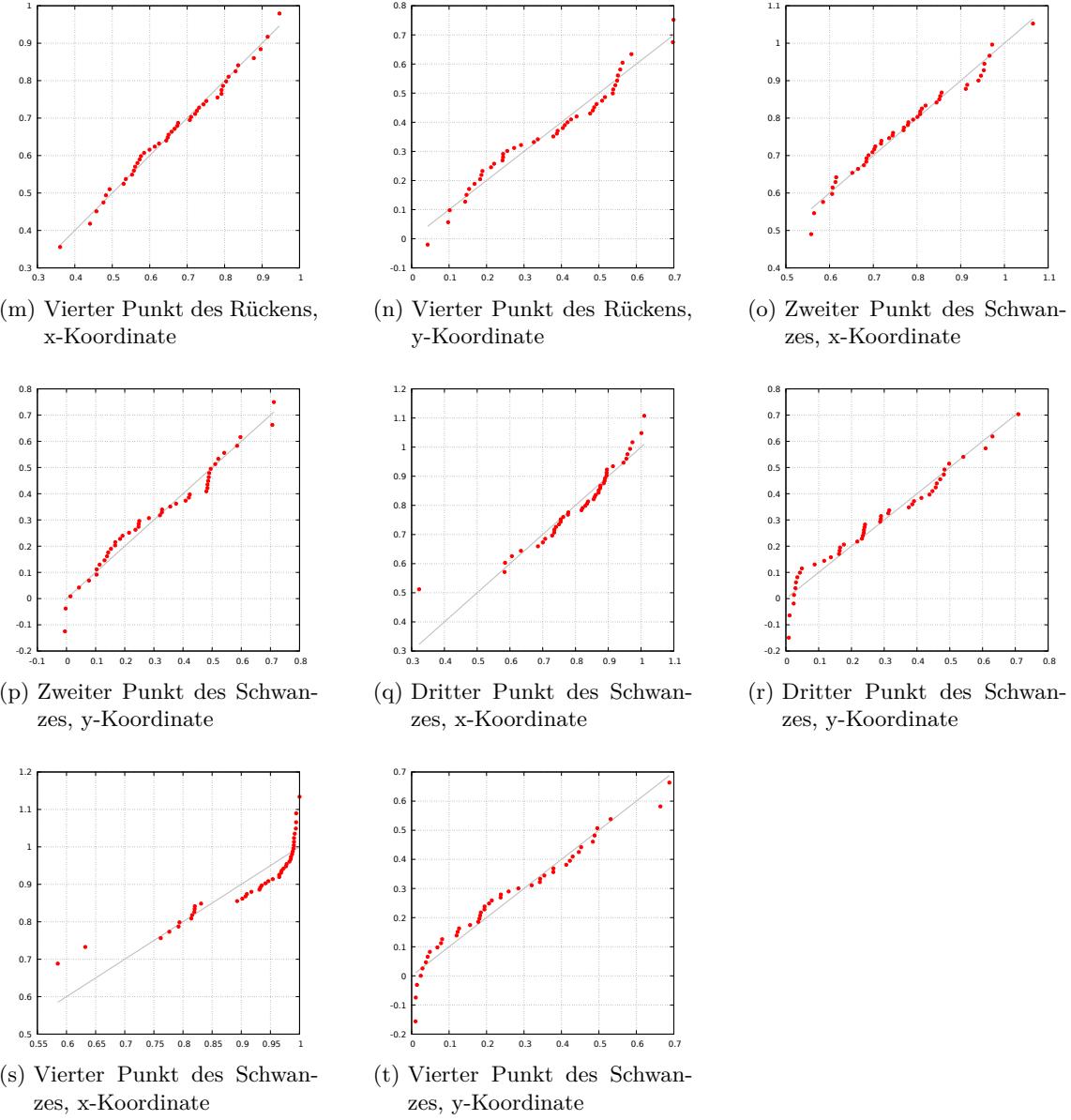


Abbildung C.1.: Quantil-Quantil-Diagramme für alle Dimensionen der Wirbelsäule

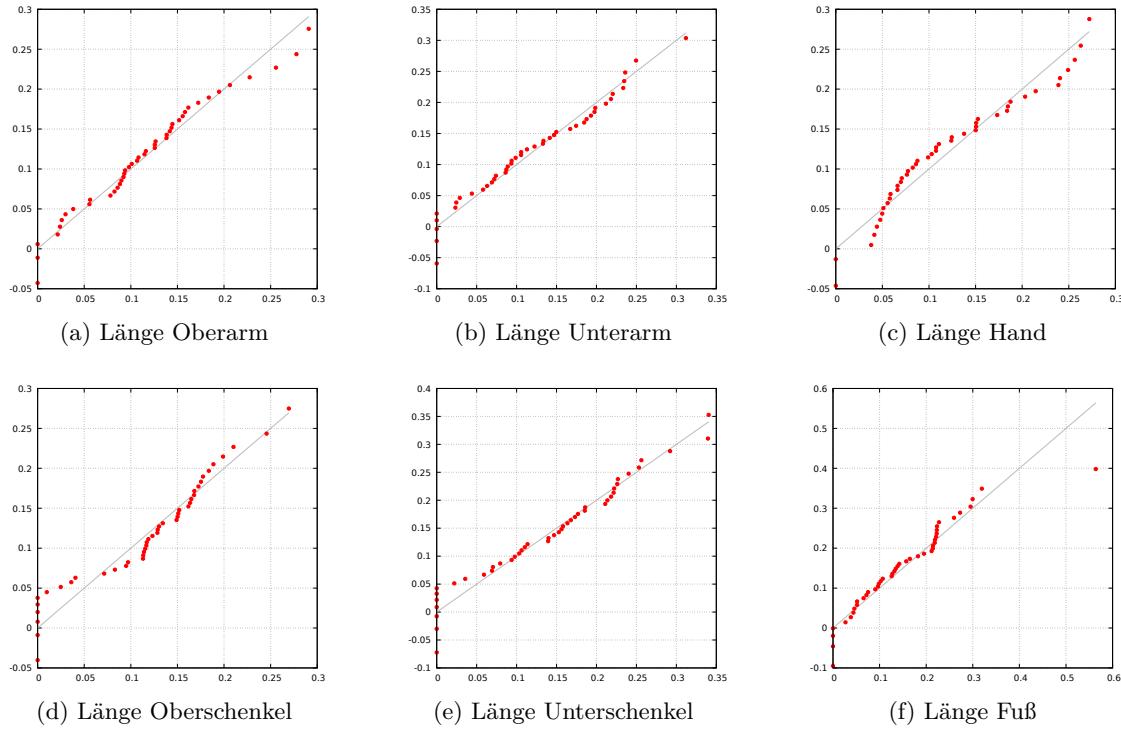


Abbildung C.2.: Quantil-Quantil-Diagramme für die Dimensionen der Eingabedaten, die zusätzlich zur Wirbelsäule erhoben wurden. Nicht dargestellt ist das binäre Attribut *Flügel* und die Anzahl der Beine mit Bodenkontakt.

- 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/Panzerspitzmaus>
- 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/Gnus>
- Schwein 100kg, <https://de.wikipedia.org/wiki/Hausschwein>
- Känguru 2-90kg, <https://de.wikipedia.org/wiki/K%C3%A4ngurus>
- Tyrannosaurus 9 Tonnen, <https://de.wikipedia.org/wiki/Tyrannosaurus>
- Dromedar 300-700kg, <https://de.wikipedia.org/wiki/Dromedar>
- Afrikanischer Strauß bis 135kg, [https://de.wikipedia.org/wiki/Afrikanischer\\_Strau%C3%9F](https://de.wikipedia.org/wiki/Afrikanischer_Strau%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 7. April 2020

(Nina Zimbel)