

Západočeská univerzita v Plzni  
Fakulta aplikovaných věd  
Katedra informatiky a výpočetní techniky

## **Bakalářská práce**

# **Modelování tvaru lidské postavy s fyziologickými omezeními**

# Prohlášení

Prohlašuji, že jsem bakalářskou práci vypracoval samostatně a výhradně s použitím citovaných pramenů.

V Plzni dne 30. dubna 2018

Lukáš Ščurko

# Poděkování

Chtěl bych poděkovat panu doc. Ing. Josefu Kohoutovi, Ph.D. za odborné vedení práce a cenné rady, které mi pomohly tuto práci zkompletovat.

## **Abstract**

The subject of the thesis is the algorithm for the observation of the physiological limits in modeling the human figure based on statistically processed anthropometric data. This algorithm is implemented in the existing open source application Makehuman. The algorithm to Makehuman tool user presents its outputs via color highlighting of named part of body. The main benefit of the thesis result is a feedback to Makehuman tool user whether the shaped body is realistic or not. During the development the algorithm was tested from several point of view. Positive fact is, that the performance tests of algorithm confirmed minimal influence to Makehuman tool responses speed. An open point for future work stays a fact that in some cases the algorithm shows a real human body as physiologically unrealistic.

## **Abstrakt**

Předmětem práce je algoritmus hlídající fyziologické meze při modelování lidské postavy na základě statisticky zpracovaných antropometrických dat. Tento algoritmus je implementován do již existující open source aplikace Makehuman. O reálnosti tvaru lidské postavy je uživatel informován pomocí barevného označení názvu části těla. Hlavním přínosem této práce je, že uživatel dostane zpětnou vazbu o reálnosti vytvořené lidské postavy. Během vývoje byl algoritmus testován z několika hledisek. Pozitivním faktem je, že daný algoritmus aplikaci výrazně nezpomaluje. Otevřeným bodem pro další práci zůstává skutečnost, že v některých případech algoritmus ukazuje, že reálné lidské tělo je fyziologicky nerealistické.

# Obsah

<b>1</b>	<b>Úvod</b>	<b>7</b>
<b>2</b>	<b>Nástroje</b>	<b>8</b>
2.1	MakeHuman . . . . .	8
2.2	3D Human Body Generator . . . . .	9
2.3	Reshaper and Autorigger . . . . .	9
2.4	Character Creator . . . . .	11
2.5	Body Talk . . . . .	12
2.6	Styku . . . . .	13
2.7	Shrnutí . . . . .	14
<b>3</b>	<b>Metody</b>	<b>15</b>
3.1	Cage-based deformation . . . . .	15
3.1.1	Techniky pro vytvoření ovládací sítě . . . . .	16
3.2	Parametric body deformation . . . . .	17
3.3	Laplacian Surface Editing . . . . .	19
3.3.1	Manipulace se sítěmi . . . . .	20
3.3.2	Deformace . . . . .	21
3.4	Srovnání . . . . .	21
<b>4</b>	<b>Návrh aplikace</b>	<b>23</b>
4.1	Vstupní model . . . . .	23
4.2	Grafické uživatelské rozhraní . . . . .	23
4.3	Deformační prostředek . . . . .	24
4.4	Deformační metoda . . . . .	25
4.5	Fyziologické omezení . . . . .	25
4.5.1	Korelace . . . . .	26
4.5.2	Percentil . . . . .	26
<b>5</b>	<b>Implementace</b>	<b>32</b>
5.1	Fyziologické omezení . . . . .	32
5.1.1	Data . . . . .	33
5.1.2	Testování statistických hypotéz . . . . .	34
5.1.3	Zpracování dat . . . . .	36
5.1.4	Programovací část . . . . .	42
5.2	Používané technologie . . . . .	43

<b>6 Testování</b>	<b>44</b>
6.1 Časové a paměťové nároky . . . . .	44
6.2 Intuitivnost a realističnost výsledků . . . . .	45
6.2.1 První testování . . . . .	45
6.2.2 Druhé testování . . . . .	48
6.3 Prostor pro zlepšení . . . . .	49
<b>7 Závěr</b>	<b>50</b>
<b>Literatura</b>	<b>51</b>

# 1 Úvod

V poslední době roste popularita 3D virtuálního světa a tím vzniká motivace k vytvoření 3D avatara konkrétního člověka. Modelování lidského těla může vést k realistické animaci virtuálních lidí ve filmech, videohrách, lékařských simulačních prostředích apod.

Existuje nespočet nástrojů, které umožňují úpravu rozměrů 3D avatara. Ovšem na světě je jen pár nástrojů, které se navíc pokoušejí o hlídání fyziologických mezí (např. Body Talk). Ve většině případů uživatel dostane k dispozici model a k němu nějaký deformační prostředek, pomocí kterého model upravuje. Po provedení několika změn na postavě ale dost často již dostane nereálné rozměry. To ale většina nástrojů neřeší, protože jsou určeny pro modeláře, kteří vytvářejí postavy do filmů, počítačových her apod. V těchto případech je mnohdy nereálnost žádoucí a stačí, když postavy dobře vypadají. Pro běžného uživatele, který chce vidět, jak by vypadal, kdyby zhubl nebo případně přibral na váze, je tento nástroj nedostačující. To mě motivovalo k tomu, abych se pokusil tento problém oslovit. Cílem je informovat uživatele o reálnosti vytvořené postavy. Pokud se dostane mimo reálné rozměry, měl by být o tom informován. Jak se k tomu daný uživatel postaví, už je na něm, ale bude vědět, že daná postava se na světě jen ztěží pohybuje.

V následující kapitole si popíšeme existující nástroje pro modelování lidské postavy a srovnáme je. V kapitole 3 prozkoumáme metody používané pro modelování tvaru postavy nerigidní deformací výchozího modelu. Dále navrhneme metodu umožňující intuitivně měnit tvar dané 3D postavy v rámci fyziologických mezí.

Na základě návrhu vytvoříme algoritmus, který bude v intuitivním editoru, ve kterém uživatel bude snadno měnit vzhled dané 3D postavy ve smyslu přiber nebo zhubni, hlídat fyziologické meze, tj. např. není možné, aby dotyčný měl malý zadek, když má velkou pánev apod.

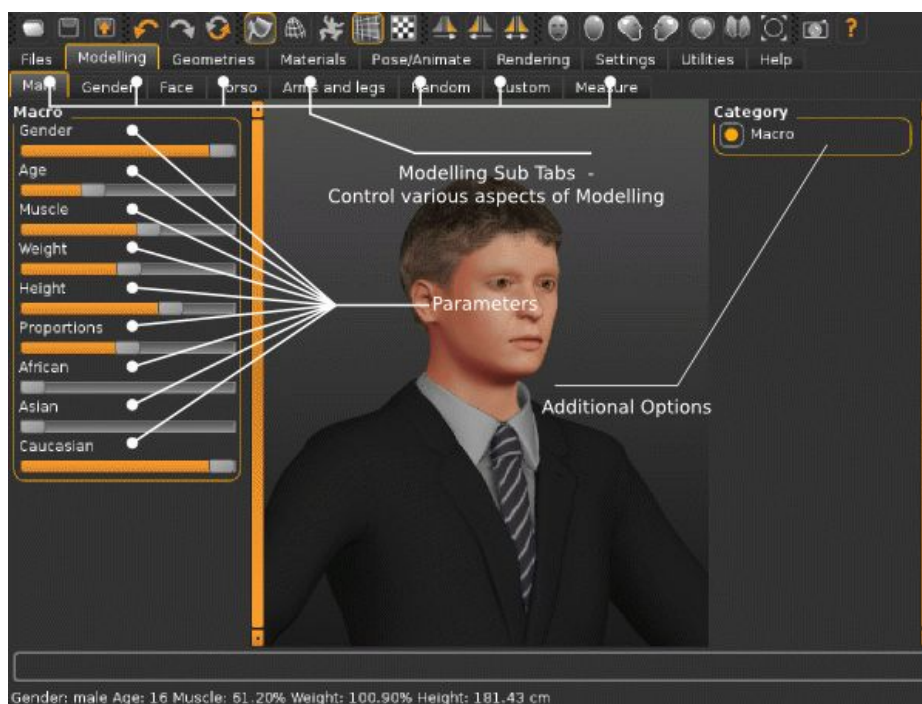
## 2 Nástroje

Existují nástroje, které se zabývají modelováním lidské postavy. Ty se liší jak v ceně, tak i v možnostech jednotlivých úprav. Některé se zabývají převážně změnou parametrů postavy, jiné spíše pohyby člověka.

### 2.1 MakeHuman

MakeHuman je nástroj, který je volně stažitelný i se zdrojovým kódem. Uživatel má možnost snadno měnit parametry postavy pomocí táhel v levém panelu (viz. obrázek 2.1). Lze měnit nejen pohlaví, věk, svalnatost, váhu a výšku, ale také se například dá měnit délka prstů, výška trupu nebo tvar obličeje. V nadsázce by se dalo říct, že lze z modelu 3D postavy vytvořit také mimozemskou bytost.

Může sloužit pro představu, jak by člověk vypadal v oblečení, které nabízí aplikace. Touto problematikou se více zabývá například velmi populární komerční nástroj Poser, který poskytuje tisíce druhů oblečení a doplňků.



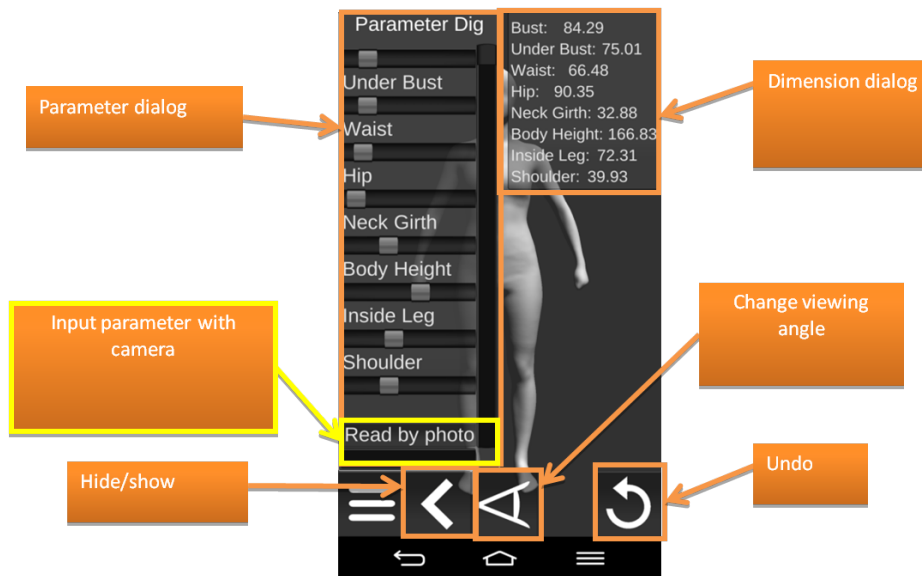
Obrázek 2.1: Screenshot aplikace MakeHuman<sup>1</sup>

<sup>1</sup><https://img.lawyerment.com/dl/screenshots/30/14930.jpg>



## 2.2 3D Human Body Generator<sup>2</sup>

3D Human Body Generator je Android aplikace, která je parametrickým nástrojem 3D modelu lidského těla. Ovládacími prvky uživatelského rozhraní jsou posuvníky v dialogovém okně parametrů, které zahrnují poprsí, bustu, pas, kyčle, obvod krku, výšku těla, vnitřní stranu nohy a rameno. Při přetažení posuvníku se model deformuje podle hodnoty posuvníku a rozměrové dialogové okno okamžitě zobrazí novou hodnotu (viz. obrázek 2.2).



Obrázek 2.2: Android aplikace 3D Human Body Generator<sup>2</sup>

## 2.3 Reshaper and Autorigger<sup>3</sup>

Reshaper je volně dostupný nástroj od společnosti SmartBody. Používá se pro zpracování a manipulaci s 3D naskenovaným lidským tělem. K naskenování uživatel může použít např. nástroj Styku (viz. podkapitola 2.6), též Microsoft Kinect nebo kterýkoliv skener, který naskenuje jeho postavu. Po naskenování se nahraje sken do aplikace. Ta využívá databázi tvarů těl, které porovnává se skenem a vybere nejvíce odpovídající model, který ve skutečnosti poté uživatel deformuje.

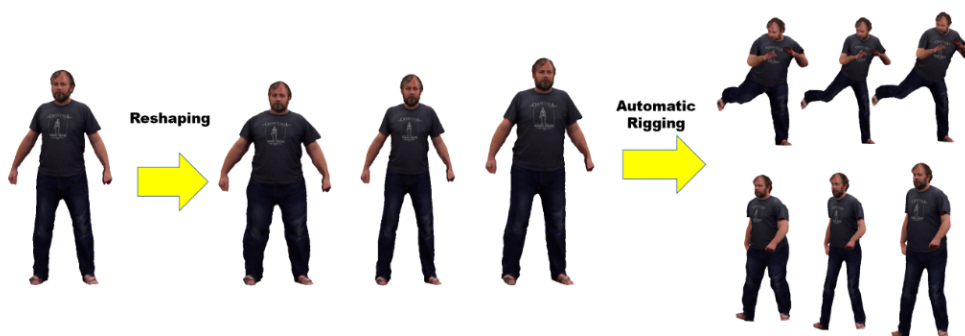
Autorigger je taktéž volně dostupný nástroj od SmartBody, který se zaměřuje na reálné pohyby modelů. Ať už se jedná o překonávání překážek,

<sup>2</sup><http://homepage.tudelft.nl/h05k3/Projects/proj3DHumanBodyGenAPP.html>

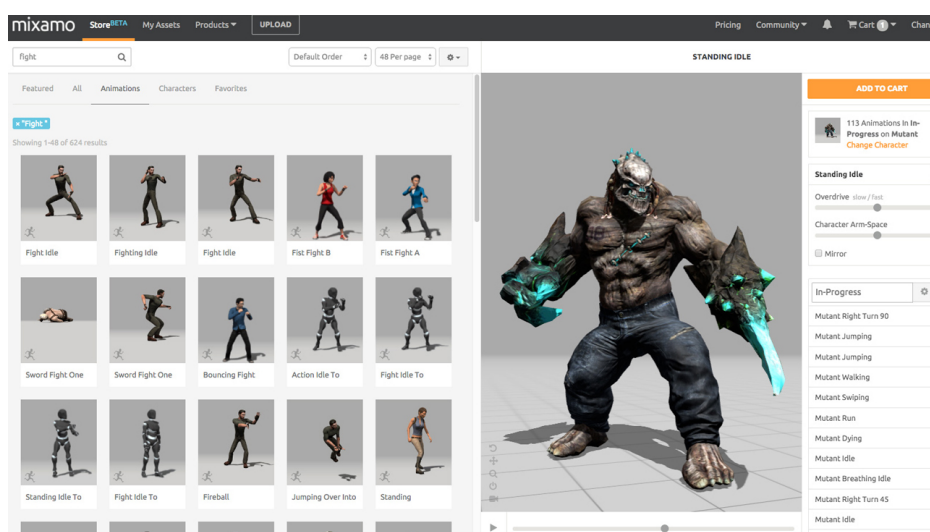
<sup>3</sup><http://smartbody.ict.usc.edu/autoriggerandreshaper>

rozpohybování rtů při mluvení či běh. Autorigger dokáže vzít výstup z Reshaperu a přidat modelu pohyby (viz. obrázek 2.3).

Rozpohybováním modelů se taktéž zabývá Mixamo<sup>4</sup>. V tomto nástroji může uživatel použít již předem vytvořené postavy společností Mixamo nebo si nahraje vlastní vytvořenou postavu. Po zvolení postavy má na výběr velkou škálu pohybů, které daná postava bude provádět. U pohybů lze měnit jejich rychlost, délku skoku apod. (viz. obrázek 2.4)



Obrázek 2.3: Znázornění, jak SmartBody mění vstupní model.<sup>3</sup>



Obrázek 2.4: Screenshot z aplikace Mixamo<sup>4</sup>

<sup>4</sup><https://www.mixamo.com/>

## 2.4 Character Creator<sup>5</sup>

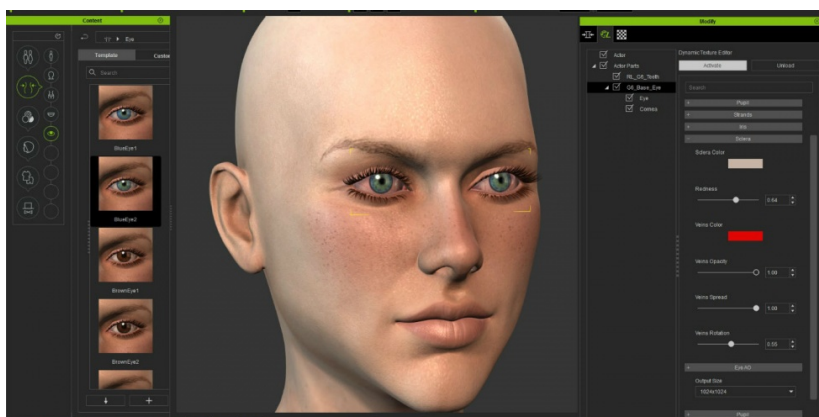
Character Creator je volně stažitelný s tím, že si lze připlatit a získat tak další technologie. Zde jsou uvedeny nabízené balíčky:

- **Character Creator**, cena: zadarmo
  - Character Creator se pyšní realistickými pohyby modelů a nepočtem možností úprav jednotlivých částí postavy. Za zmínku určitě stojí obličej, který lze upravit do podoby konkrétního člověka (viz. obrázek 2.5). Dokonce umožňuje simulovat stárnutí postavy.
- **iClone** - obsahuje Character Creator + iClone7, cena: US\$199.00
  - iClone7 zjednodušuje svět 3D animace v uživatelsky přívětivém výrobním prostředí, které kombinuje tvorbu postav, animaci, scénický design a filmové vyprávění. GPU vykreslovač poskytuje jedinečnou produkční rychlost a uměleckou vizuální kvalitu. IClone Animation Pipeline bezproblémově propojuje běžné 3D nástroje a herní stroje s průmyslovými standardy pro interaktivní aplikace, filmovou a virtuální produkci. iClone je široce využíván nezávislými tvůrci, spisovateli, režiséry, animátory nebo kýmkoliv, kdo si přeje rychle změnit svou vizi na realitu.
- **Character Animation Pipeline** - obsahuje Character Creator + iClone7 + 3DXchange Pipeline, cena: US\$599.00
  - Pomocí technologie 3DXchange Pipeline lze importovat a exportovat všechny typy 3D prostředků, včetně statických objektů, animovaných rekvizit, kódů. Lze je také upravit s výstupními možnostmi přizpůsobenými pro Unity, Unreal, Maya, Blender, Cinema 4D nebo Daz Studio. Díky tomu je iClone univerzálním a základním nástrojem pro návrh hry.

Obdobnou funkcionalitou disponují nástroje DAZ a Poser.

---

<sup>5</sup><https://www.reallusion.com/character-creator/>



Obrázek 2.5: screenshot z aplikace Character Creator<sup>5</sup>

## 2.5 Body Talk

Vývojáři Body Talku se rozhodli navrhnout jednoduché řešení pro realistické, metricky přesné, 3D lidské avatary a ukázat, že je překvapivě přesné [12]. Všichni mají představu o tom, jak vypadá štíhlé, či podsadité tělo. Podobně jako u Reshaper (viz. podkapitola 2.3) tuto znalost autoři Body Talku použili k vytvoření aplikace. Oproti Reshaperu má Body Talk navíc výpočetní model, který vyjadřuje tvar těla podle slov (viz. obrázek 2.6). Může generovat 3D těla ze slov a nebo naopak slova z těl. Kromě toho jako jeden z mála nástrojů hlídá fyziologické meze.



Obrázek 2.6: Screenshot z aplikace BodyTalk. V levé části jsou slova, která popisují danou osobu [12].

V první řadě byl vytvořen statistický model, který zachycuje tvarovou variaci lidí z přibližně 4000 laserových skenů mužů a žen. Vybralo se 128

mužských a 128 ženských šablon, které se použily jako tréninkový set. Ten se stará o omezení na reálný 3D tvar těla.

Body Talk vytváří tvary, které jsou na první pohled nerozlišitelné od těl vytvořených pomocí skenů s vysokým rozlišením a přesnost metrických údajů je dostatečná pro mnoho úkolů.

## 2.6 Styku<sup>6</sup>

Na závěr kapitoly uvedu velice zajímavý nástroj Styku, který sice neumožňuje úpravu lidské postavy, ale lze jej využít k naskenování člověka a poté výsledný sken vložit například do Reshaperu, Autoriggeru nebo Mixama. Uvedené nástroje jsou popsány výše.

K nástroji Styku je potřeba speciální podstavec. Po postavení se na tento rotující podstavec, začne skener snímat člověka (viz. obrázek 2.7). Díky svým ostrým infračerveným snímkům s vysokým rozlišením dokáže technologie Styku zachytit miliony datových bodů za několik vteřin rychlým, neinvazivním procesem.



Obrázek 2.7: 3D skener člověka pro aplikaci Styku<sup>6</sup>

Styku sestavuje 3D model z více než 600 infračervených snímků za sekundu. Výsledný 3D model lze otáčet a zvětšit tak, aby bylo možné podrobně vyhodnotit jednotlivé části těla. Jakmile je model vytvořen, technologie Styku pro pokročilé rozpoznávání funkcí automaticky nalézá detailní body na těle (např. kyčle), které se používají k získávání obvodů pro různé oblasti těla.

<sup>6</sup><http://www.styku.com/bodyscanner>

## 2.7 Shrnutí

Rozdíl mezi jednotlivými nástroji není jen v zaměření se na parametry či pohyby, ale také v ovládání. V aplikacích jako jsou např. MakeHuman a 3D Human Body Generator se parametry postavy mění posuvníky, které jsou obvykle v levé či pravé části obrazovky. Zato např. v aplikaci Character Creator uživatel mění postavu tak, že klikne myší na část těla, kterou chce změnit. Při stisknutí myši táhne na stranu, kam chce danou část posunout. Po uvolnění myši aplikace posune i okolní body tak, aby byla zachována hladkost deformované postavy.

# 3 Metody

Výše popsané nástroje používají nejrůznější deformační metody, které se starají o rozměry 3D postavy. Existuje několik základních způsobů, jak deformaci provést, některé si v této kapitole popíšeme.

## 3.1 Cage-based deformation

V této metodě jsou uvedeny dva tvary: model a ovládací síť. Model reprezentuje objekt, který chceme deformovat. Pro vytvoření ovládací sítě byla navržena řada automatických nebo poloautomatických technik. Tyto metody lze zhruba kategorizovat do několika typů, některé jsou popsány níže v 3.1.1[10]. Pro každý vrchol  $v$  v modelu nejprve vypočítáme jeho průměrné hodnoty váhových funkcí  $w_j$  s ohledem na každý vrchol  $p_j$  v nedeformované ovládací síti. Pro provedení deformace přesouváme vrcholy řídicí sítě pro vyvolání deformace na původním povrchu. Nechť  $\hat{p}_j$  je poloha vrcholů z deformované řídicí sítě, pak se nová pozice bodu  $v$  v deformovaném modelu vypočte jako

$$\hat{v} = \frac{\sum_j w_j \hat{p}_j}{\sum_j w_j}. \quad (3.1)$$

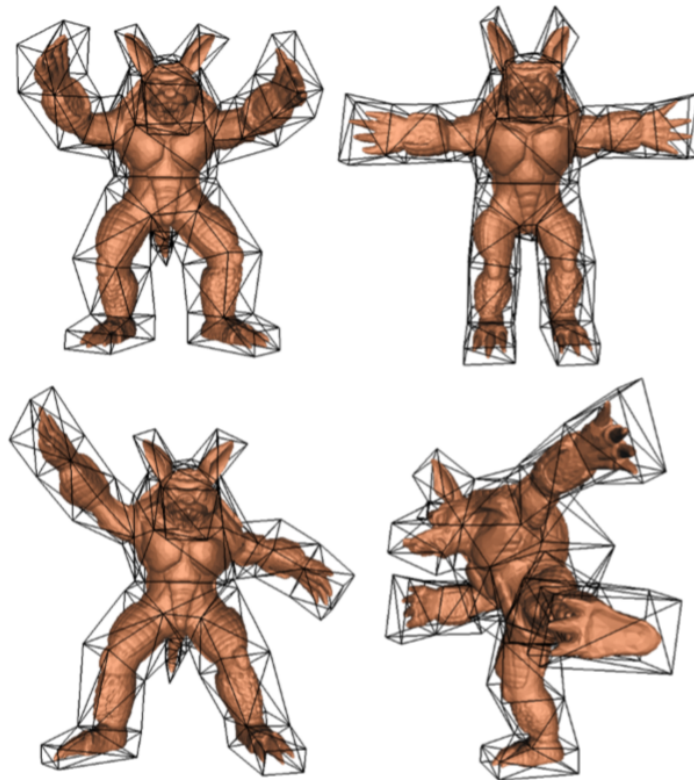
Kvůli lineární přesnosti, jestliže  $\hat{p}_j = p_j$ , pak  $\hat{v} = v$ .

Existuje více způsobů, jak volit váhové funkce. Jednou z funkcí je konstrukční funkce, která interpoluje sadu hodnot definovaných na vrcholech sítě, je základní operací v počítačové grafice [8]. Interpolant má využití v aplikacích, jako je stínování, parametrizace a deformace. Souřadnice středních hodnot se u uzavřených polygonů osvědčily jako vynikající metoda pro konstrukci interpolantu. Tato metoda generalizuje souřadnice středních hodnot z uzavřených 2D polygonů na uzavřené trojúhelníkové sítě. Vzhledem k takové síti  $P$  jsou tyto souřadnice spojitě všude a hladké na vnitřním povrchu  $P$ . Souřadnice jsou lineární na trojúhelnících  $P$  a mohou reprodukovat lineární funkce na vnitřním povrchu  $P$ . Pro ilustraci jejich užitečnosti, s ohledem na několik zajímavých aplikací včetně konstrukce volumetrických textur a povrchové deformace [8].

Povrchová deformace je jednou aplikací souřadnic střední hodnoty, která závisí na vlastnosti lineární přesnosti.

Obrázek 3.2 a obrázek 3.1 ukazují několik příkladů deformací vytvořených tímto procesem. Obrázek 1 (a) znázorňuje koně před deformací a ob-

klopující ovládací síť je zobrazena černě. Přesunutí vrcholů řídicí sítě vytváří hladké deformace koně zobrazené v (b, c, d). Deformační technika umožňuje umělci specifikovat libovolný uzavřený trojúhelníkový povrch jako kontrolní síť a nevyžaduje volumetrické buňky k pokrytí vnitřku. Tato technika také vytváří hladké, realisticky vypadající deformace i s malým počtem kontrolních bodů a je poměrně rychlá. Generování souřadnic střední hodnoty pro obrázek 3.2 trvalo dle autorů 3.3s a 1.9s pro obrázek 3.1. Nicméně každá z deformací trvala pouze 0.09s a 0.03s, což je dostatečně rychlé na to, aby se tyto deformace použily v reálném čase. Vzhledem k tomu, že tyto hodnoty byly naměřeny před 12 lety, tak na současných počítačích bude pravděpodobně deformace ještě rychlejší.

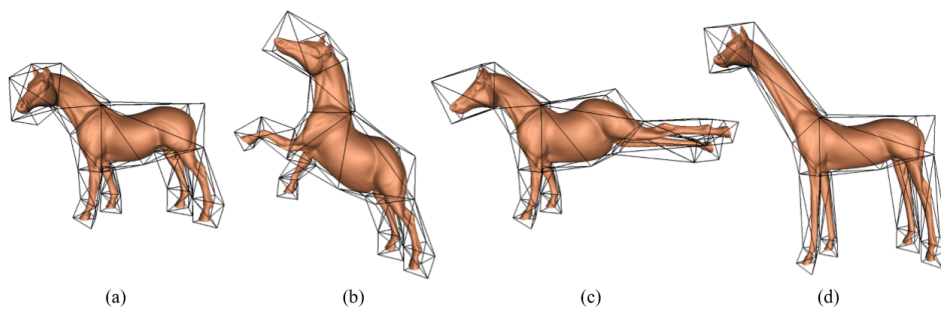


Obrázek 3.1: Původní model a ovládací síť v okolí, zobrazená černě (vlevo nahoře). Deformace řídicí sítě vytváří hladké deformace základního modelu [8].

### 3.1.1 Techniky pro vytvoření ovládací sítě

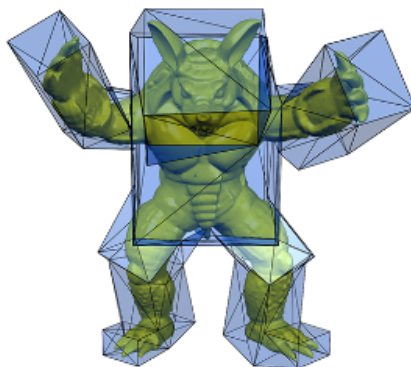
Technik pro vytvoření ovládací sítě je několik, ty jsou popsány v článku [10].





Obrázek 3.2: (a) Původní model koně s uzavřenou, ovládací trojúhelníkovou sítí vyzobrazenou černě. (b, c, d) Několik deformací generovaných pomocí souřadnic 3D středních hodnot aplikovaných na modifikovanou řídicí síť [8].

- **Template based cage generation methods** sestaví sadu klecových šablon s předem definovanými topologiemi pro vytvoření těsně ohraničené klece pro daný 3D model.
- **Bounding shape based approaches** vytváří klec pro vstupní síť tak, že nejprve vytvoří orientovaný ohraničovací kvádr pro každou síťovou část a poté je spojuje.



Obrázek 3.3: Klec pomocí techniky Bounding shape [10].

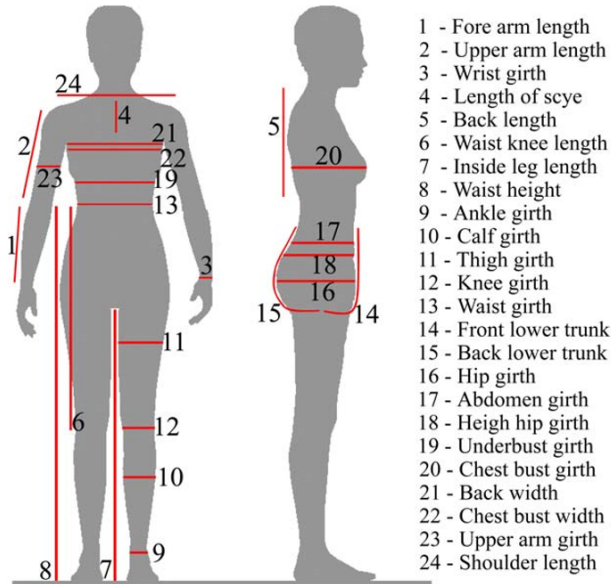
- **The skeleton-based approach** využívá skelet vytvořený uživatelem, který vede k vytvoření mnoha dílčích klecí a následně je spojuje.

## 3.2 Parametric body deformation

Velmi rozšířenou metodou je parametrická deformace těla, která je popsána v [9]. Na základě šablonového modelu se podle antropometrických měřítek

vytvářejí různé velikosti těla. Tyto standardy se používají během konstrukčního stádia modelu tělesa šablony tak, aby se segmentovaly do oblastí, kde je každá oblast deformována odpovídajícími parametry měření. V antropometrickém přístupu je třeba určit více než 20 specifických částí těla. Tento proces je ručně zpracováván během fáze návrhu modelu s tím, že označuje konkrétní oblasti a označuje je jedinečnými identifikačními čísly. Takže každý identifikátor odpovídá soustavě vrcholů, nebo přesněji specifické oblasti hlavního tělesa. Určení oblastí je jakýmsi heuristickým procesem založeným na hlavních měřících bodech získaných z norem ISO-7250 a ISO-8559, které jsou znázorněny na obrázku 3.4. Tyto oblasti rozdělujeme do tří skupin:

- Oblasti, kde se deformace aplikuje vertikálně, konkrétněji měření výšky, tj. výška pasu, délka vnitřní nohy.
- Oblasti, kde je obvodová deformace aplikována na klouby, tj. obvod kolen, zápěstí.
- Oblasti, kde je obvodová deformace aplikována mezi dva klouby nebo oblasti, tj. obvod stehna, lýtka, pasu.

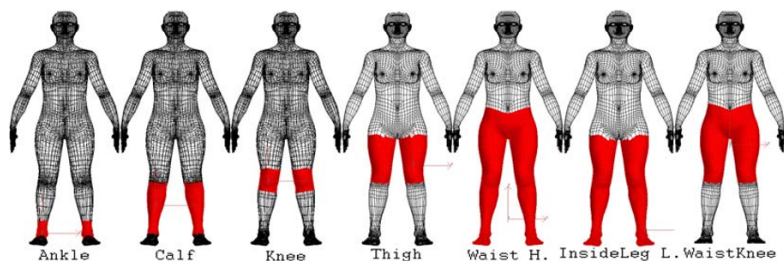


Obrázek 3.4: Některé z hlavních bodů měření těla [9].

V závislosti na anatomickém tvaru oblastí jsou deformační funkce se specifickými řídicími parametry smíchány za účelem změny velikosti těla. Provedení

deformačních funkcí je založeno na spline a radiální vzdálenosti povrchových vrcholů. Před aplikací jakékoliv deformace se posunutí vrcholů filtruje, aby se zachovala kontinuita mezi sousedními a překrývajícími se oblastmi.

Deformování tělesného modelu podle antropometrického parametru vyžaduje přesnou specifikaci oblasti, která bude deformována. Zachování hladkosti a kontinuity sousedních nebo překrývajících se regionů přísně závisí na specifikaci. Obrázek 3.4 představuje některé z orientačních bodů, které se používají pro deformaci modelu těla. Pro dosažení deformace kolem určitého orientačního bodu je ručně definována odpovídající skupina indexů kolem orientačního bodu. Obrázek 3.5 znázorňuje některé oblasti odpovídající orientačním bodům znázorněným na obrázku 3.4.

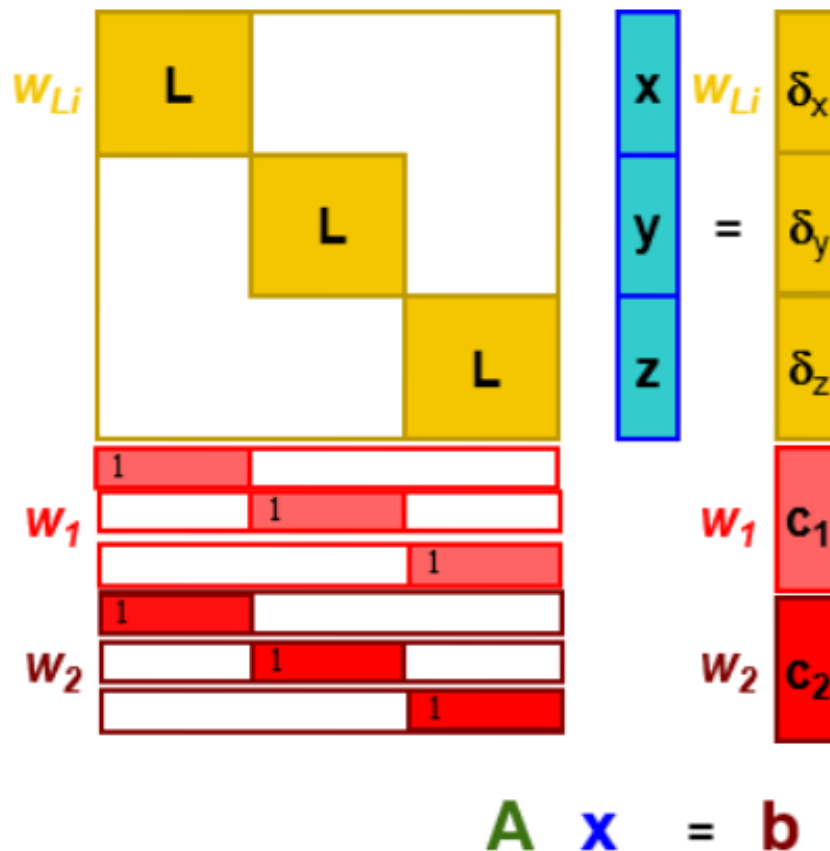


Obrázek 3.5: Některé z oblastí definovaných v modelu [9].

### 3.3 Laplacian Surface Editing

Třetím přístupem k deformaci je editace povrchu, která je popsána v článku [11]. Operace editace povrchu obvykle vyžadují co nejvyšší zachování geometrických detailů povrchu. Tvrdíme, že geometrický detail je vnitřní rotací povrchu a že následně je povrchová editace nejlépe vykonávána působením přes vnitřní zobrazení povrchu. Zajišťujeme reprezentaci povrchu založenou na Laplacianově síti, kódováním každého vrcholu ve vztahu k jeho okolí. Laplacianova síť je vylepšena tak, aby byla invariantní k lokálně linearizovaným tuhým přeměnám a změnám velikosti.

Laplacianova matice  $L$  (viz. obrázek 3.6) obsahuje jednotlivé váhy  $w$  mezi vrcholy. Lze si představit pružinu, která je natažena mezi dvěma vrcholy, kde  $w$  určuje její tuhost. Je možné nějaký vrchol zafixovat tak, že s ním nelze nijak pohnout a ani nereaguje na změny sousedních vrcholů. Velkou výhodou této metody je, že matice se napočítá na začátku a od té doby se nemění. Laplacianské souřadnice  $\delta$  jsou lineární funkcí globální geo-



Obrázek 3.6: Schématické maticové znázornění soustavy lineárních rovnic, kde  $L$  je Laplacianovská matice,  $\delta$  představuje rotaci a  $C$  je lineární omezení (pevně daná poloha vrcholu) [11].

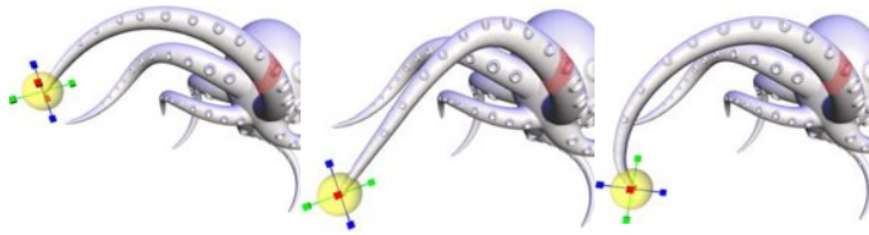
metrie sítě, která umožňuje efektivní přeměnu mezi absolutními a vnitřními reprezentacemi řídkého lineárního systému. Laplacianské souřadnice jsou invariantní při přesunu (absolutní geometrie), ale nejsou invariantní k měřítku a rotaci, což představuje hlavní praktický problém.

Tento postup sice obsahuje problém, ale je zde popsán, protože je jednoduchý na pochopení. Vzniklý problém řeší několik technik, které danou invarianci zajistí. Tyto techniky jsou důkladně popsány v článku Laplacian Surface Editing [11].

### 3.3.1 Manipulace se sítěmi

Snad nejjednodušší forma manipulací s existující sítí spočívá v manipulaci s rukojetí (handle), což je sada vrcholů, se kterými uživatel může posouvat a otáčet. Manipulace s rukojetí (handle) se podobá výsledku manipulaci s

předmětem vyrobeným z fyzicky měkkého materiálu (viz. obrázek 3.7).



Obrázek 3.7: Na prvním obrázku je zobrazen výběr umístění rukojeti. Druhý obrázek je výsledek po posunutí rukojeti a na třetím je vidět, jak se změní obrázek při otočení rukojeti. Můžeme vidět, že ani posunutí ani otočení rukojeti nezmění kvalitu obrázku [11].

### 3.3.2 Deformace

Máme vstupní síť, tedy známe souřadnice vrcholů sítě, díky tomu lze určit Laplacianovu matici  $L$  a  $\delta$ .

$L$  a  $\delta$  popisují vztahy mezi vrcholy sítě, tj. definují tvar sítě včetně všech detailů. Po stanovení sítě je zapotřebí znalost souřadnic vrcholů v původní síti. Změna tvaru či nějakého detailu lze provést pomocí změny  $\delta$ . Po změně  $\delta$  nebo po přidání lineárního omezení (např. zafixování nějakého vrcholu v prostoru, apod.) je potřeba vypočítat přeurčenou soustavu rovnic (obecně nelineární), která se nejčastěji vyřeší metodou nejmenších čtverců pro stanovení nových souřadnic vrcholů.

## 3.4 Srovnání

Uvedené metody se dají porovnat z několika pohledů. Z hlediska vstupu je nejjednodušší Laplacianská metoda, jelikož potřebuje pouze 3D vstupní model, zatímco Cage-based má na vstupu ještě klec a u parametrické metody je zapotřebí mít parametry vstupního modelu.

Časově na tom bude nejlépe Cage-based, který obsahuje  $m$  vrcholů klece a  $n$  vrcholů jemné sítě, kterou deformujeme. Tím dostáváme čas  $m*n$ . Paměť téměř žádnou nepotřebuje, ta je  $m+n$ . U parametrické metody je to daleko složitější, jelikož je model rozřezán na několik částí. Laplacian potřebuje matici, která je sice řídká, ale za to velká. Paměťově je to  $n^2$ . Řešíme zde soustavu lineárních rovnic, tedy složitost je  $n^3$ , i když ta bude o něco menší, jelikož máme řídkou matici.

Cage-based nemá příliš intuitivní ovládání, protože změna jednoho bodu klece ovlivňuje celý model. Laplacian je v tomto ohledu o něco lepší, ale nejvíce intuitivní ovládání má parametrická metoda, u které uživatel přesně ví, kterou část modelu mění.

Každá metoda má své výhody, proto žádnou z nich nemůžeme zavrhnout. Všechny přístupy jsou na sebe převoditelné, to znamená, že uživatel může mít například táhlítka, která mění obvod pasu, šířku ramen apod., ale tím se jen určí poloha klíčových bodů sítě a zbytek se Laplacianovskou deformací zdeformuje.

## 4 Návrh aplikace

Naším cílem je vytvořit aplikaci, umožňující uživateli intuitivním způsobem modelovat postavu a do aplikace implementovat algoritmus, hlídající fyziologické meze na základě statisticky zpracovaných antropometrických dat.

Vytvoření této aplikace vyžaduje provedení několika rozhodnutí, a to: jaký model bude mít uživatel na vstupu, jaké jsou nezbytné prvky pro prohlížení postavy, jakým způsobem uživatel bude řídit deformaci modelu, co za deformační metodu se použije na pozadí, a jak se bude ověřovat, že výsledek je realistický či nikoliv z pohledu fyziologie člověka.

### 4.1 Vstupní model

Jednou z nezbytných částí je vstupní model. Vstupním modelem je myšlena 3D postava člověka, na kterou se bude aplikovat deformační metoda. Tento model lze získat zpravidla dvěma způsoby a to:

- Stažením z internetových stránek, které tyto modely nabízejí za menší poplatek. Cena se pohybuje v řádech několika dolarů či lze sehnat i avatary, které jsou zcela zdarma.
- Naskenováním vlastního avatara použitím 3D skeneru.

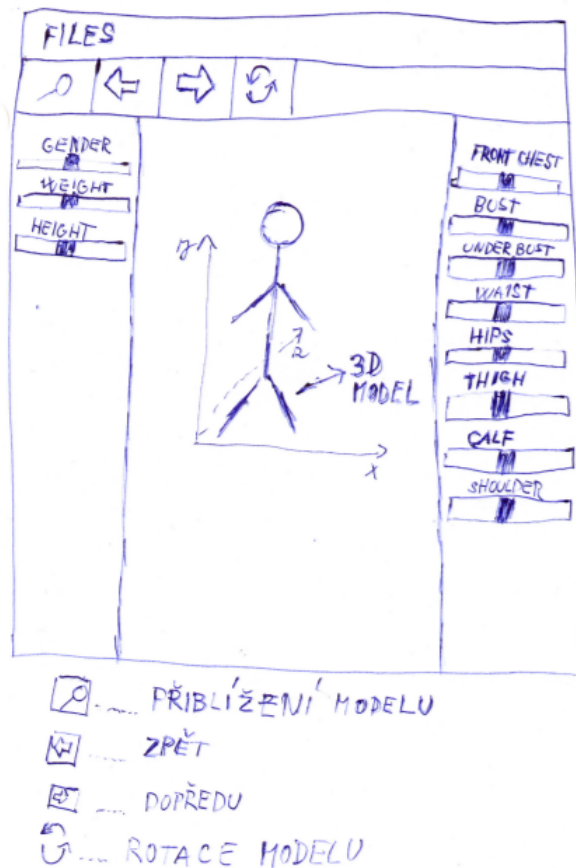
### 4.2 Grafické uživatelské rozhraní

Vstupní model je nutné vložit do grafického prostředí, které je ovšem potřeba nejprve naprogramovat. Jak bude aplikace graficky vypadat, je samozřejmě na programátorovi nebo na člověku, který ji chce naprogramovat.

Ať už bude grafické rozhraní vypadat jakkoliv, mělo by mít několik důležitých vlastností, jako je:

- Prostor pro vstupní model, se kterým bude uživatel pracovat.
- Otáčení, přibližování/oddalování vstupního modelu, aby bylo vidět, jak daná postava vypadá ze všech stran, z dálky a nebo naopak z detailu.
- Deformační prostředek, který umožní měnit parametry modelu (viz. podkapitola 4.3)

Na obrázku 4.1 je vidět, jak by takové grafické uživatelské rozhraní mohlo vypadat.



Obrázek 4.1: Návrh grafického uživatelského rozhraní.

### 4.3 Deformační prostředek

Aby uživatel mohl měnit tvar postavy, potřebuje mít k dispozici nějaký deformační prostředek. Mně se nejvíce zamlouvají táhlítka, která jsou velmi populární. Využívá je mnoho nástrojů, jako například MakeHuman, Body Talk nebo androidní aplikace 3D Human Body Generator. Všechny uvedené nástroje jsou více popsány v kapitole 2. Posunutím táhlítka se zdeformuje část postavy, se kterou je dané táhlítko propojeno. Například může měnit obvod pasu, hrudníku nebo i obecně váhu či výšku.

Kromě táhlítek lze jako deformační prostředek použít "handle" neboli rukojeť (viz. podkapitola 3.3.1) či ovládací síť, kterou si lze představit jako



klec okolo postavy. Posunutím některým z vrcholů ovládací sítě (klece) se deformuje i model. (viz. kapitola 3.1).

Z hlediska následného hlídání fyziologických mezí je dle mého názoru ideální využít jako deformační prostředek táhlítka. Každé táhlítko bude měnit jeden rozměr, což by mělo pomoci při získávání rozměrů jednotlivých částí těla a následnému hlídání fyziologických mezí.

## 4.4 Deformační metoda

Po vytvoření grafického prostředí, načtení vstupního modelu a vybrání deformačního prostředku, přichází již na řadu implementace deformační metody. Tu bude uživatel využívat na deformaci vstupního modelu skrze deformační prostředek. Více o deformačních metodách je v kapitole 3.

## 4.5 Fyziologické omezení

K tomu, abychom deformací dostali reálné postavy a ne žádné mimozemské bytosti, je vhodné implementovat do deformační metody algoritmus, který bude hlídat fyziologické meze. To znamená, že uživatel dostane zpětnou vazbu ohledně reálnosti postavy. Tedy pokud nějaká část těla nebude odpovídat v souvislosti s ostatními rozměry těla, bude o tom uživatel vhodným způsobem informován.

K hlídání fyziologických mezí je zapotřebí získat antropometrická data, která obsahují rozměry lidí. Ideální je mít data od určité skupiny lidí, na které bude aplikace cílena. Například, pokud bychom se rozhodli vytvořit aplikaci pro profesionální sportovce, určitě by se nebrali v úvahu děti a senioři. Ale naopak by se použily rozměry aktivních sportovců napříč všemi sporty s věkem od 18 do 35 let. Samozřejmě na nejvyšší úrovni můžou sportovat i lidi mladších 18 let nebo starších 35 let, jako například Jaromír Jágr, který i ve svých 45 letech pořád předvádí neskutečné výkony. Ale my hledáme největší skupinu lidí, kteří spadají do této kategorie, proto jsem uvedl toto věkové rozmezí. Důležité je také určit pohlaví, na které se zaměříme. Obě pohlaví mají odlišnou stavbu těla, z čehož vyplývá, že i poměry mezi jednotlivými částmi těla se budou lišit. Muži mají zpravidla širší ramena, ženy zase větší obvod hrudníku atp. Pokud se rozhodneme zacílit jak na muže, tak na ženy, je vhodné poté tyto dvě skupiny odlišit. Také si musíme dát pozor na to, abychom měli dostatek potřebných rozměrů a dostatečný počet lidí, kteří nám poskytnou své rozměry. Je samozřejmé, že ze dvou lidí nedostaneme statistické hodnoty, o které bychom se mohli opřít. Čím více

dat získáme, tím přesnější hodnoty dostaneme.

Po získání dostatečného množství dat pro vybranou skupinu lidí musíme určit, které části těla jsou na sobě závislé, tedy navzájem se ovlivňující. To si můžeme určit podle našeho odhadu. Vidím, že čím větší mají lidé obvod hýždí, tím větší mají i obvod stehen. Tento přístup je ale celkem nešťastný. Ano, můžeme mít ve většině odhadů pravdu, ale také nemusíme. Je to subjektivní pohled daného člověka. Lepším postupem bude vzájemnou závislost získat pomocí statistiky. Na to lze použít korelace.

### 4.5.1 Korelace

Korelace [1] znamená vzájemný vztah mezi dvěma procesy nebo veličinami. Pokud se jedna z nich mění, mění se korelativně i druhá a naopak. Pokud se mezi dvěma procesy ukáže korelace, je pravděpodobné, že na sobě závisí, nelze z toho však ještě usoudit, že by jeden z nich musel být příčinou a druhý následkem. To samotná korelace nedovoluje rozhodnout, protože korelace neimplikuje kauzalitu.

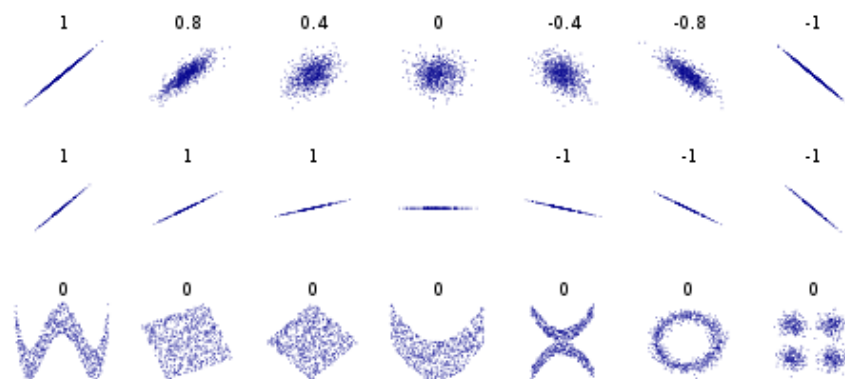
V určitějším slova smyslu se pojem korelace užívá ve statistice, kde znamená vzájemný lineární vztah mezi veličinami  $x$  a  $y$ . Míru korelace pak vyjadřuje korelační koeficient, který může nabývat hodnot od  $-1$  až po  $+1$ .

Vztah mezi znaky či veličinami  $x$  a  $y$  může být kladný, pokud (přibližně) platí  $y = kx$ , nebo záporný ( $y = -kx$ ). Hodnota korelačního koeficientu  $-1$  značí zcela nepřímou závislost (antikorelaci), tedy čím více se zvětší hodnoty v první skupině znaků, tím více se zmenší hodnoty v druhé skupině znaků, např. vztah mezi uplynulým a zbývajícím časem. Hodnota korelačního koeficientu  $+1$  značí zcela přímou závislost, např. vztah mezi rychlostí bicyklu a frekvencí otáček kola bicyklu. Pokud je korelační koeficient roven  $0$  (nekorelovanost), pak mezi znaky není žádná statisticky zjiřitelná lineární závislost. Je dobré si uvědomit, že i při nulovém korelačním koeficientu na sobě veličiny mohou záviset, pouze tento vztah nelze vyjádřit lineární funkcí, a to ani přibližně.

Korelace nám sice řekne, jak moc jsou na sobě jednotlivé veličiny závislé, ale už se z ní nedozvíme, od jaké hodnoty budeme brát data jako závislá. Jestli například od korelačního koeficientu  $0.6$  nebo  $0.7$ , to už záleží na našem úsudku.

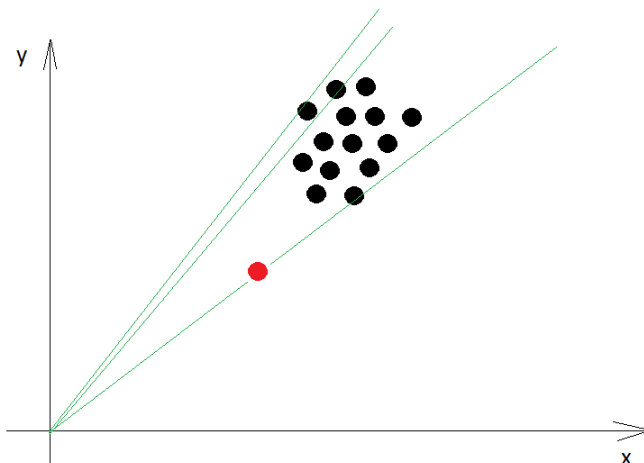
### 4.5.2 Percentil

Když provedeme korelaci a určíme si, od kterého korelačního koeficientu budeme brát dva rozměry člověka jako spolu související, můžeme provést



Obrázek 4.2: Na obrázku jsou vidět grafy se vztahy mezi dvěma veličinami. Čísla nad grafy reprezentují korelační koeficient[1].

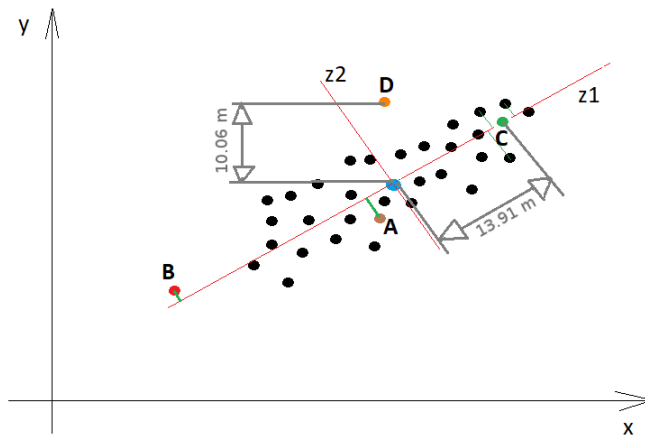
poměr mezi těmito dvěma rozměry. Poměr provedeme jednoduše tak, že u každé osoby vydělíme jeden rozměr tím druhým. Dostaneme tedy desetinné číslo, které bude vyjadřovat poměr mezi dvěma souvisejícími veličinami (například obvod hýždí s obvodem pasu). Na obrázku 4.3 vidíme shluk černých bodů, které představují dva rozměry (např.obvod hýždí, pasu) lidí. Poměr je popsán přímkou začínající v počátku a procházející daným bodem. Tím nastává problém u tohoto postupu, jelikož bod, který je sice mimo shluk, ale leží na dané přímce, bude brán jako dobrý, i když už by měl být mimo mez (viz. červený bod na zelené přímce).



Obrázek 4.3: Na obrázku se nachází graf, který znázorňuje, jak se chová poměr.

Dalším přístupem může být PCA (principal component analysis), který

je vidět na obrázku 4.4. Pomocí PCA získáme bod ve středu (modrý centroid), tj. nový počátek a dva ortogonální vektory, jeden ve směru maximální distribuce, druhý ve směru minimální distribuce (červená osa souřadnic). Sestrojíme přímku procházející novým počátkem ve směru maximální distribuce (přímka  $z_1$ ) a pro každý bod (obvod hýždí, pasu) stanovíme vzdálenost od této přímky (zelené úsečky vedoucí od bodu k přímce), přičemž toto číslo pak bude vstupem do percentilu. Ovšem i toto řešení má své problémy. Červený bod B je tímto postupem vyhodnocen jako lepší než hnědý bod A, ačkoliv vizuálně bod B je zcela mimo. To by se mohlo vyřešit použitím Euklidovské vzdálenosti (od bodu k centroidu), ale to také problém neřeší, protože shluk bodů není kruhem, ale elipsou (v optimálním případě úsečkou). Oranžový bod D by byl hodnocen jako lepší než zelený bod C.



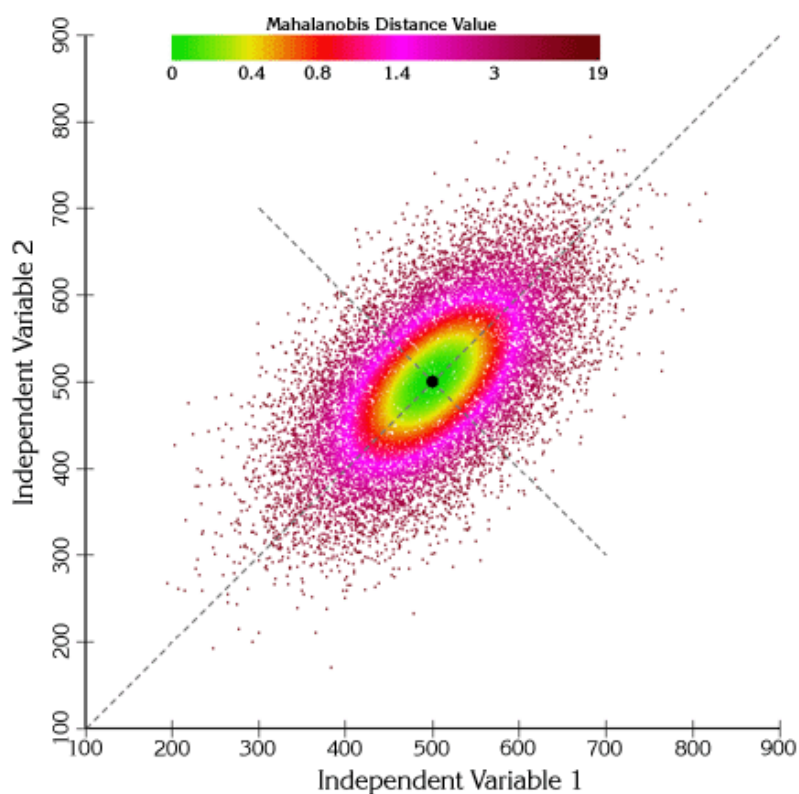
Obrázek 4.4: Na obrázku se nachází graf, který znázorňuje PCA.

Problém vzniklý u Euklidovské vzdálenosti řeší Mahalanobisova vzdálenost, která je definována vztahem

$$D_m(x_i, x_p) = \sqrt{(x_i - x_p)^T S^{-1} (x_i - x_p)}, \quad (4.1)$$

kde  $D_m$  je vzdálenost  $i$ -tého bodu  $x_i$  od průměru  $x_p$  vzhledem ke konfidenčnímu elipsoidu, určenému kovarianční maticí  $S$  [4]. Chování Mahalanobisovy vzdálenosti je vidět na obrázku 4.5. Vzdálenost odpovídá rozptylu každé proměnné a kovariaci mezi proměnnými. Geometricky to dělá to, že se data transformují na nekorelované údaje a poté se počítají běžné euklidovské vzdálenosti pro transformované údaje. Mahalanobisova vzdálenost tedy poskytuje řešení, jak měřit vzdálenosti, které berou v úvahu distribuci dat.

<sup>2</sup>[http://www.jennessent.com/images/graph\\_illustration\\_small\\_3.gif](http://www.jennessent.com/images/graph_illustration_small_3.gif)



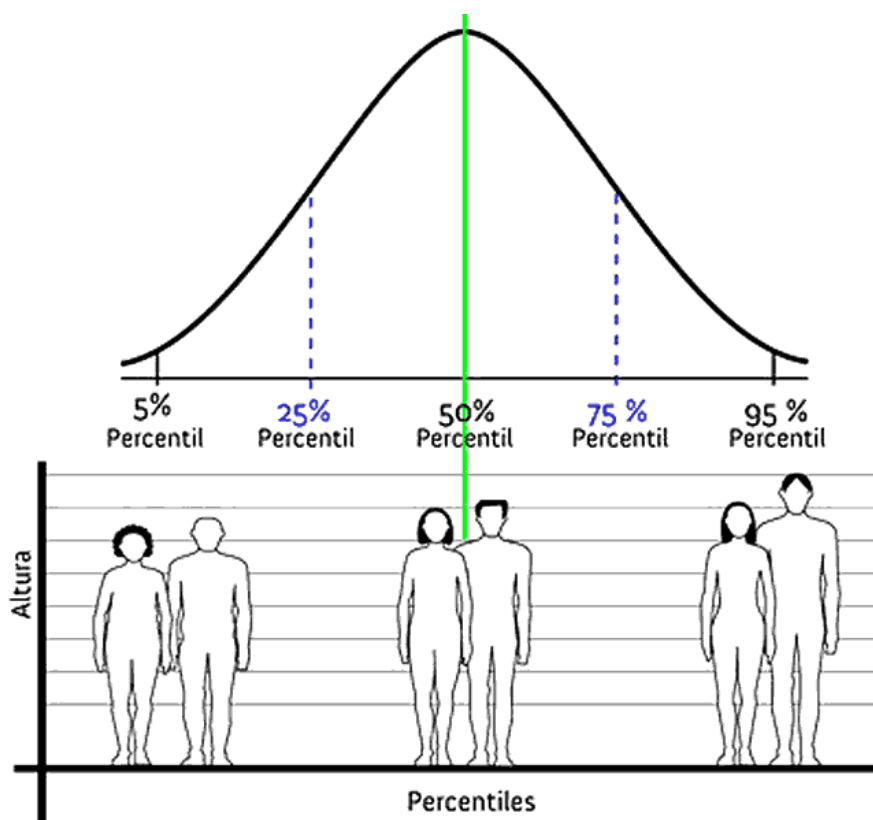
Obrázek 4.5: Na obrázku se nachází graf, který znázorňuje, jak se chová Mahalanobisova vzdálenost<sup>2</sup>.

Na získané hodnoty jednou z metod použijeme percentil [2], který je statistickým parametrem, určujícím, kde se nachází poměr mezi naměřenými hodnotami, které jsou na sobě závislé (např. obvod pasu/obvod přes hýždě, obvod pod prsy/obvod přes prsa, apod.) v rámci celé populace stejného věku a pohlaví. Vyjadřuje tedy relativní postavení vůči svým vrstevníkům. Každý jeden percentil vyjadřuje jednu setinu frekvence výskytu v rámci celé populace.

Například, když zjistíme padesátiprocentní percentil u výšky mužů. To nám řekne, že padesát procent mužů je menších a padesát procent vyšších. Jinými slovy pokud by se muži seřadili do řady podle výšky, tento muž by stál uprostřed. Kdybychom hledali sedmdesátipětiprocentní percentil, tak bychom našli muže, který je vyšší než sedmdesát pět procent mužů a menší, než dvacet pět procent mužů. Viz. obrázek 4.6.

Když tento příklad převedeme na naši situaci, tedy poměry mezi na sobě závislými rozměry těla, tak percentilem určíme, kolik procent lidí má menší

<sup>4</sup>[https://curiosoando.com/wp-content/uploads/2013/09/ejemplo\\_gr%C3%A1fico\\_percentil.png](https://curiosoando.com/wp-content/uploads/2013/09/ejemplo_gr%C3%A1fico_percentil.png)



Obrázek 4.6: Horní polovina obrázku znázorňuje graf s percentily (grafem je Gaussova křivka). Dolní polovina ukazuje, do jaké části grafu spadají lidé s danou výškou.<sup>4</sup>

či větší poměr než námi navržená postava. Tuhle znalost můžeme využít při rozhodování, zda-li daná postava je reálná či naopak reálná není.

Například pokud nám vyjde, že poměr mezi obvodem hýždí a obvodem stehén je jednaprocentní percentil, bude to znamenat, že devadesátdevět procent lidí má větší poměr než naše postava. Z čehož plyne, že tato postava se statisticky vymyká naprosté většině populace. Pro naše účely můžeme předpokládat, že tato postava není reálná.

Naopak, když poměr vyjde jako padesátiprocentní percentil, bude tato postava na vrcholu Gaussovy křivky, tudíž statisticky má takovou postavu nejvíce lidí. My tedy budeme předpokládat, že je reálná.

V uvedených příkladech bylo jasné, kdy postava je reálná a kdy ne. Tyto situace ale nastávají jen zřídka kdy a my se musíme rozhodnout, kde udělat hranici mezi postavou reálnou a naopak nereálnou.

Variant je několik. Můžeme říci, že nereálná postava bude ta, kterou má například maximálně 5 % lidí. Pokud by nám nestačilo procentuální vyjádření hranice a například i 1 % lidí by nám přišlo ještě jako příliš velká

skupina lidí na to, abychom ji označili jako nereálnou, můžeme hranici posunout ještě dále a to na hodnotu, která by se považovala za outlier. Outliers [5] jsou dány vztahem navrženým panem Tukey.

$$O1 \geq Q3 + (Q3 - Q1) * 1.5 \quad (4.2)$$

$$O2 \leq Q1 - (Q3 - Q1) * 1.5, \quad (4.3)$$

kde  $O1$  a  $O2$  jsou outliers,  $Q3$  je třetí kvartil (75% percentil) a  $Q1$  označuje první kvartil (25% percentil). Rozdíl těchto dvou hodnot je vynásoben v nerovnici konstantou a poté přičten k třetímu kvartilu či odečten od prvního kvartilu, což záleží na tom, jestli hledáme horní nebo dolní outlier. Dle mého uvážení je dobré zvolit několik hranic. Určíme si, že okolo padesáti-procentního percentilu (vrchol Gaussovy křivky) se nachází reálné postavy. Postupně, jak budeme klesat po Gaussově křivce jak na jednu, tak i na druhou stranu, bude se postava postupně měnit v nereálnou. Tudíž, uživatel bude vědět, že se dostal za hranici průměrné postavy. Pokud bude pokračovat po Gaussově křivce dál, zjistí, že už je na hraně reálnosti. Po posunutí ještě o kousek dále již následuje oznámení, že je postava nereálná.

# 5 Implementace

V předchozí kapitole popsaný návrh řešení jsem se rozhodl realizovat na již existujícím nástroji. Po prostudování dostupných nástrojů, jsem si vybral Makehuman, který již má implementovaný vstupní model s deformační metodou. Jedním z důvodů, proč jsem se rozhodl pro tento nástroj je ten, že Makehuman je OpenSource. Tedy volně stažitelný i se všemi zdrojovými kódy a je umožněno do něho zasahovat.

Makehuman jako deformační prostředek používá táhlítka. Při spuštění tohoto nástroje jsou v levé části táhlítka, která upravují celkový vzhled postavy. Tím je myšlena například výška, váha, pohlaví a další základní parametry člověka. To je druhý důvod, díky kterému jsem se rozhodl pro využití tohoto nástroje, protože si těmito táhlítky uživatel vytvoří svůj typ postavy. Poté se přepne do záložky Measure, která se nachází v horní liště, kde má k dispozici táhlítka, která mění již jednotlivé části postavy jako je například obvod pasu, hýždí nebo šířka ramen.

## 5.1 Fyziologické omezení

Jelikož Makehuman má již naimplementovanou deformační metodu a deformace funguje za použití táhlítek, mohl jsem začít s implementováním algoritmu, který bude hlídat fyziologické meze.

Zde nastaly první problémy. Makehuman je psaný v jazyce Python, který jsem do té doby znal jen okrajově, tudíž jsem se musel pro mě novým programovacím jazykem seznámit.

Když jsem pochopil syntaxi a pomalu jsem psanému kódu rozuměl, začal jsem studovat, kde se co provádí, abych mohl do vhodného místa kódu přidat později svůj algoritmus na hlídání fyziologických mezí.

Po detailnějším zkoumání jsem zjistil, že hodnota táhlítek není v centimetrech, ale pohybuje se na uzavřeném intervalu 0 až 1. Původně jsem chtěl k hlídání fyziologických mezí použít hodnoty přímo z táhlítek, které bych následně kontroloval s rozměry naměřených lidí, ale za těchto okolností tento způsob nelze použít.

Musel jsem tedy najít jiný způsob, jak rozměry modelu získat. To se mi povedlo díky funkci, která u každého táhlítka hodnotu přepočítává na centimetry. Jelikož tyto hodnoty budu používat i v jiné třídě a navíc potřebuji mít uložené hodnoty všech rozměrů modelu, vytvořil jsem HashMapu, což je v podstatě slovník. Pro definici slovníku se používá dvojice klíč-hodnota.



V mém případě klíčem je název části těla a hodnotou je rozměr části těla v centimetrech. HashMapa je ukládána globálně, aby k ní byl přístup odkudkoli. Při každé změně nějakého z rozměrů postavy, se tato HashMapa přepisuje a díky tomu mám neustále k dispozici aktuální rozměry modelu, se kterými budu následně moci pracovat. K tomu ovšem potřebuji antropometrická data.

### 5.1.1 Data

Jak již je uvedeno v kapitole 4, k hlídání fyziologických mezí jsou potřeba naměřená data určité skupiny lidí.

K dispozici jsem měl dvoje antropometrická data, která obsahují naměřené rozměry lidí. Oboje data jsou veřejně dostupná. Tato data se liší jak v počtu měřených hodnot, tak hlavně typem lidí. Zatímco první data jsou vojenská, která zahrnují populaci hlásící se do armády. Ve druhých datech je široká veřejnost. Následují informace o datech:

- **Vojenská data ANSI88<sup>1</sup>** - Průzkum amerického vojenského antropometru z roku 1988 je jednou z nejpoužívanějších antropometrických databází kvůli velkému množství opatření a přísné metodice.
- **Data se širokou veřejností NHANES<sup>1</sup>** - Vláda USA provádí pravidelné studie výživy od počátku 60. let. V rámci těchto studií byly získány základní antropometrické údaje.

Vojenská data mají sice o každém člověku mnoho rozměrů, ale mohla by zkusovat výsledky, poněvadž předpokládám, že do armády se hlásí převážně lidi s atletickou postavou. Proto jsem chtěl ve své práci použít data obsahující širokou veřejnost. Ovšem problémem mohlo být nedostatek informací o každém člověku.

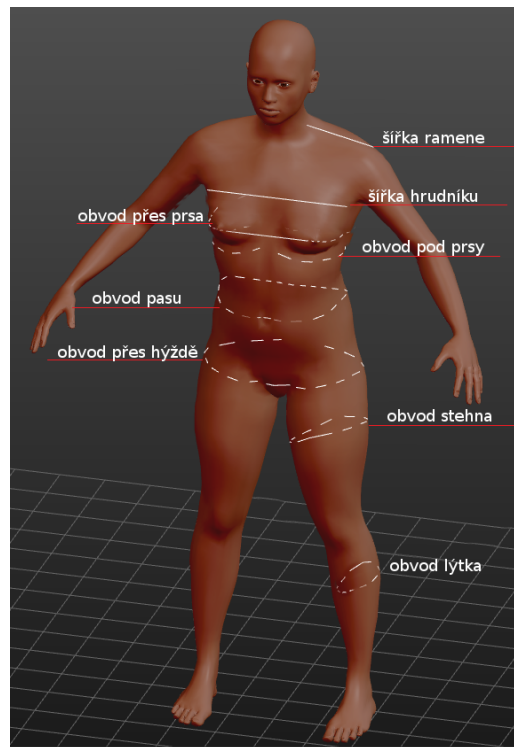
Před tím, než jsem začal zjišťovat, jaké rozměry o každém člověku data obsahují, musel jsem se zamyslet, jaké rozměry mě vůbec budou zajímat. Předpokládám, že dospělý člověk nebude měnit svoji výšku, ani poměr mezi výškou trupu a nohou. Tudíž jsem se rozhodl použít rozměry (viz. obrázek 5.1), které se změní při zhubnutí či ztloustnutí každé osoby.

Problém nastal při zjištění, že v datech se širokovou veřejností úplně chybí rozměry hrudníku, tedy obvod přes prsa a obvod pod prsy. Přesto jsem pořád uvažoval, že tato data alespoň částečně použiji na ostatní parametry postavy. To bych zrealizoval tak, že bych oboje data(ANSI88 a NHANES) nakombinoval. K tomu, abych tuhle akci mohl provést, je zapotřebí, aby

---

<sup>1</sup><http://mreed.umtri.umich.edu/mreed/downloads.html>

se data příliš nelišila. Na tento problém jsem použil testování statistických hypotéz, konkrétně t-studentův test.



Obrázek 5.1: Naměřené hodnoty člověka, které budou použity pro hlídání fyziologických mezí.

### 5.1.2 Testování statistických hypotéz

Testování hypotéz [3] je statistická metoda, která určuje, jak pravděpodobná jsou naměřená data v případě platnosti hypotézy, kterou testujeme.

Pro testování vždy vytváříme dvě hypotézy. Testovanou, kterou označíme jako nulovou a značíme ji  $H_0$  a alternativní, kterou značíme  $H_1$ . Nulová hypotéza  $H_0$  je formulovaná negativně („to, co chceme prokázat, neplatí“) a alternativní hypotéza  $H_1$  formulovaná jako neplatnost  $H_0$ .

Nulová hypotéza  $H_0$  bývá formulovaná pomocí rozdělení, nebo pravděpodobnosti. Cílem testování je  $H_0$  zamítnout a přijmout  $H_1$ .

K samotnému testování se využívá tzv. testová statistika T. Jedná se o vzorec, funkci dat, která udává, jak pravděpodobná jsou naměřená data, pokud platí nulová hypotéza.

- **p-hodnota testu** (z anglického p value) je pravděpodobnost, že při  $H_0$

by testová statistika  $T$  nabyla hodnoty, jaká vyšla z dat, nebo hodnoty ještě extrémnější (mimo interval  $\langle -T, T \rangle$ ).

- **Hladina testu** se označuje  $\alpha$ . (Nejčastěji  $\alpha = 0.05 = 5\%$ ). Je to zvolené číslo z intervalu od 0 do 1, resp. 100 % (čím menší, tím lepší). Pokud je  $p < \alpha$ , tak platnost  $H_0$  je velmi málo pravděpodobné a potom: Zamítáme  $H_0$  na hladině  $\alpha$  a přijímáme  $H_A$ .

Buď  $H_0$  platí, ale nastala data, která se objevují s pravděpodobností menší než  $\alpha$  (nastalo něco velmi nepravděpodobného), nebo vskutku platí  $H_A$ , k čemuž se kloníme.

Výsledek testu je pak statisticky významný na hladině  $\alpha$ .

- **Kritická hodnota testu pro hladinu  $\alpha$**  je hranice mezi statisticky významnou a nevýznamnou hodnotou testové statistiky  $T$ .

Neporovnává se  $p$  přímo s  $\alpha$ , ale pro  $\alpha$  se vypočítá kritická hodnota, s níž se porovná samotné  $T$ . Jedná se o ekvivalentní porovnání. (Dříve numericky přístupnější - nebylo nutné počítat kvantily.)

## T-studentův test

Jedním z často používaných testů statistické hypotézy je T-studentův test [6]. V závislosti na situaci, kdy se používá, se rozlišuje:

- **jednovýběrový t-test**, který slouží k porovnání střední hodnoty  $\mu$  s konstantou ( $H_0: \mu = \mu_0$ );
- **dvouvýběrový (nepárový) t-test**, který slouží k porovnání střední hodnoty  $\mu_1$  jedné skupiny se střední hodnotou  $\mu_2$  jiné skupiny ( $H_0: \mu_1 - \mu_2 = \text{konst}$ ).

např. střední hodnota systolického tlaku u kuřáků a nekuřáků; nebo střední hodnota systolického tlaku u skupiny, která bere placebo, a skupiny, která bere  $\beta$ -blokátory.

- **párový t-test**, který slouží k porovnání středních hodnot mezi prvními a druhými prvky uspořádaných dvojic ( $H_0: \mu_1 - \mu_2 = \text{konst}$ ).

např. střední hodnota systolického tlaku u kuřáků před ukončením kouření a po ukončení kouření; nebo střední hodnota hladiny oxytocinu v krvi u matek a u jejich dětí.

V mém případě, tedy při porovnávání dvou na sobě nezávislých měření lidí, jsem použil dvouvýběrový (nepárový) t-test. Nejprve jsem si vytvořil dvě hypotézy. Testovanou hypotézu  $H_0$  označující, že obě skupiny nevykazují žádné zvláštní odchylky a jsou tak snadno zaměnitelné. Alternativní hypotézou je hypotéza  $H_1$ , která říká, že existují odlišnosti mezi zástupci obou skupin. Tento test jsem provedl pomocí online kalkulátoru<sup>2</sup>. U obou dat jsem si vyfiltroval stejnou skupinu lidí. Tou skupinou byly ženy od 20 do 30 let. Otestoval jsem dva parametry, kterými jsou obvod pasu a obvod přes hýždě. Oba testy dopadly téměř stejně. Uvedu tedy výsledek jen jednoho testu a to obvodu pasu.

Jak je vidět z obrázku 5.2, p-hodnota testu, tedy pravděpodobnost, že jsou data stejná, je 0.1392 (13.92 %). I když podle konvenčních kritérií se tento rozdíl nepovažuje za statisticky významný, jak je uvedeno u výsledku tohoto testu, mně 13.92 % shody přišlo velmi málo na to, abych tato data mohl bez obav sloučit. Tedy zamítám hypotézu  $H_0$  a přijímám hypotézu  $H_1$ .

Na základě t-studentova testu jsem se rozhodl, že použiji ve své práci pouze jedna data, kterými jsou data vojenská. A to i přesto, že bych raději použil data se širokou veřejností. Ty jsem ovšem nemohl využít kvůli chybějícím údajům o obvodu přes prsa a obvodu pod prsy.

### 5.1.3 Zpracování dat

Vybral jsem tedy vojenská data, která obsahují lidi od 18 do 50 let. Já se rozhodl zaměřit pouze na ženy, protože ty řeší svůj vzhled více než muži a tento nástroj by jim mohl poskytnout zpětnou vazbu, jak by mohly vypadat. Data jsem si načel do tabulkového softwaru Microsoft Excel.

Rozmezí od 18 do 50 let je příliš velké na to, aby se tyto ženy daly dát do jedné skupiny. S člověkem stárne celé jeho tělo, každý tělesný orgán prodělává své proměny. Kůže postupně ztrácí na pružnosti, mizí podkožní tuk a kůže se tak svrašťuje a propadá, také se zmenšují svaly a jsou nahrazovány vazivem [7]. Všechny tyto změny ovlivňují i jednotlivé rozměry člověka, které jsem si ještě ověřil t-studentovým testem. Vzal jsem z vojenských dat rozměry obvodu pasu od lidí od 20 do 30 let a porovnal je s lidmi od 40 do 50 let. Výsledek je vidět na obrázku 5.3. Z obrázku lze vyčíst, že p-hodnota testu je 0.0001. Z čehož vyplývá, že tato data nejsou vůbec podobná. Na druhou stranu ve vojenských datech je velmi málo žen od 40 do 50 let. Což je pochopitelné, jelikož do armády se většina lidí hlásí v mladším věku. Každopádně malá množina žen, přesněji 59 žen mohla značně tento test ovlivnit.

---

<sup>2</sup><https://www.graphpad.com/quickcalcs/ttest1.cfm>

Pro zjištění přesnějšího rozdílu v datech by byla potřeba větší množina žen v této věkové skupině.

Jednak z důvodu stárnutí, ale také díky malé množině žen ve věkové skupině 40-50 let jsem se rozhodl, že fyziologické omezení udělám na základě žen z vojenských dat ve věku od 20 do 35 let, jelikož v tomto věku všichni umí používat digitální technologie. Také se domnívám, že je to věková skupina, které nejvíce záleží na svém vzhledu. Důkazem toho je neustále se vyskytující nové případy, kdy se v této věkové skupině najde jedinec, který trpí psychickými problémy, protože není spokojen se svým tělem.

## QuickCalcs

[1. Select category](#)

[2. Choose calculator](#)

[3. Enter data](#)

**4. View results**

### Unpaired *t* test results

#### P value and statistical significance:

The two-tailed P value equals 0.1392

By conventional criteria, this difference is considered to be not statistically significant.

#### Confidence interval:

The mean of ANSUR-women-20-30-pas minus NHANES-women-20-30-pas equals -110.6142

95% confidence interval of this difference: From -257.5403 to 36.3119

#### Intermediate values used in calculations:

$t = 1.4792$

$df = 3367$

standard error of difference = 74.782

#### Learn more:

GraphPad's web site includes portions of the manual for GraphPad Prism that can help you learn statistics. First, review the meaning of [P values](#) and [confidence intervals](#). Then learn how to interpret results from an [unpaired](#) or [paired](#) *t* test. These links include GraphPad's popular *analysis checklists*.

#### Review your data:

Group	ANSUR-women-20-30-pas	NHANES-women-20-30-pas
Mean	71.9080	182.5222
SD	5.7915	2942.2111
SEM	0.1472	68.9476
N	1548	1821

Obrázek 5.2: Výsledek dvouvýběrového (nepárového) testu obvodu pasu ze dvou různých zdrojů dat

Na to, abych získal množinu žen od 20 do 35 let jsem použil v Excelu filtr, který mi umožnil odstranit řádky s ženami mimo tento věkový interval. Dále jsem potřeboval najít sloupce (tedy rozměry ženy), které budou hlídat fyziologické meze. Tyto rozměry jsou vidět na obrázku 5.1.

# QuickCalcs

[1. Select category](#)

[2. Choose calculator](#)

[3. Enter data](#)

**4. View results**

## Unpaired *t* test results

### P value and statistical significance:

The two-tailed P value is less than 0.0001

By conventional criteria, this difference is considered to be extremely statistically significant.

### Confidence interval:

The mean of ANSUR-women-20-30-pas minus ANSUR-women-40-50-pas equals -15.226

95% confidence interval of this difference: From -16.782 to -13.670

### Intermediate values used in calculations:

$t = 19.2242$

$df = 1605$

standard error of difference = 0.792

### Learn more:

GraphPad's web site includes portions of the manual for GraphPad Prism that can help you learn statistics. First, review the meaning of [P values](#) and [confidence intervals](#). Then learn how to interpret results from an [unpaired](#) or [paired](#) *t* test. These links include GraphPad's popular *analysis checklists*.

### Review your data:

Group	ANSUR-women-20-30-pas	ANSUR-women-40-50-pas
Mean	71.908	87.134
SD	5.792	9.587
SEM	0.147	1.248
N	1548	59

Obrázek 5.3: Výsledek dvouvýběrového (nepárového) testu obvodu pasu dvou odlišných věkových skupin z vojenských dat

Zde nebylo u všech rozměrů jasné, v jaké části těla byly přesně měřeny. Například pas je v datech uveden dvakrát. První údaj je měřen klasicky v pase a druhý udává obvod přes pupík. Při přiblížení si postavy v Makehuman jsem zjistil, že čára, která znázorňuje pas, vede přes pupík. Problém s určením přesného místa nebyl jen u pasu, ale například i u obvodu stehna, které se může měřit dokonce na třech místech. Abych tedy vybral z vojenských dat stejné rozměry, které jsou v Makehuman, využil jsem manuál<sup>3</sup> k datům. V manuálu je podrobně popsáno, jak se daný rozměr měřil.

## Korelace

Jak již je uvedeno v kapitole 4, po získání potřebné množiny dat, je nutné provést korelaci (o korelaci je více v podkapitole 4.5.1). Ta mi určí, které

<sup>3</sup>[http://mreed.umtri.umich.edu/mreed/downloads/anthro/ansur/ADAS-Dimension\\_Definitions.pdf](http://mreed.umtri.umich.edu/mreed/downloads/anthro/ansur/ADAS-Dimension_Definitions.pdf)

rozměry se sebou navzájem souvisí. Koeficient korelace byl počítán přes vzorec v Excelu, kterým je  $CORREL(matice1;matice2)$ . Matice1 reprezentuje sloupec jednoho rozměru a matice2 sloupec druhého rozměru. Výsledkem je korelační koeficient, který určuje, jak moc jsou na sobě data závislá.

Jaké jsou výsledky je možné vidět na obrázku 5.4. Hranici, kdy jsou rozměry na sobě závislé a kdy naopak nezávislé, jsem určil jako hodnotu koeficientu korelace 0.7. Tedy všechny výsledky nad 0.7 jsou brány jako závislé. Pro tuto hodnotu koeficientu korelace jsem se rozhodl na základě vypočítané tabulky na obrázku 5.4. Přemýšlel jsem nad hodnotami 0.6, 0.7 a 0.8, ale dle tabulky mi přišlo, že při zvolení 0.6 by na sobě bylo závislých příliš mnoho rozměrů a při hodnotě 0.8 naopak příliš málo. Z tohoto důvodu jsem vybral hodnotu 0.7.

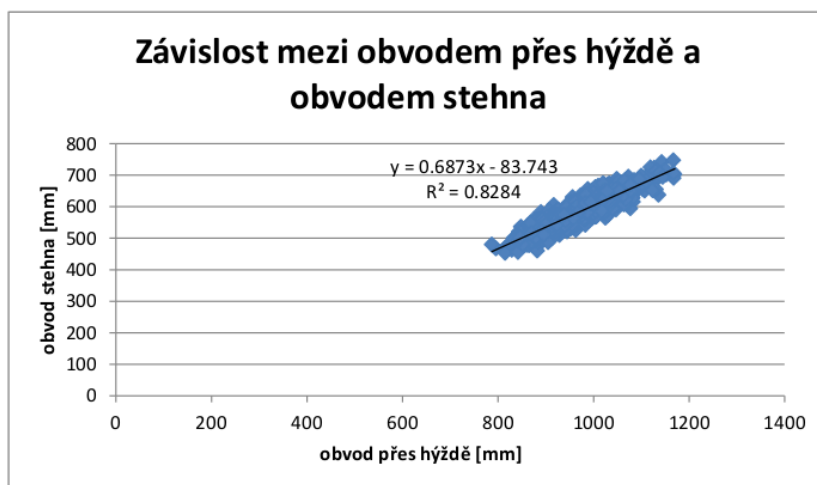
Tabulka je v podstatě symetrickou maticí, která má na diagonále (označeno žlutě) samé jedničky. Což znamená, že závislost jednoho rozměru na tom samém je 100%, což je logické. Červeně označené rozměry nejsou na žádném jiném rozměru závislé. Nejvíce nás zajímají zelená políčka, která znázorňují závislost mezi dvěma rozměry.

Korelace	obvod pasu	obvod přes hýždě	obvod přes prsa	obvod pod prsy	šířka hrudníku	šířka ramene	obvod lýtka	obvod stehna	obvod krku
obvod pasu	1	0.724722952	0.766210099	0.735156403	0.713223141	0.039389923	0.4507909	0.682207584	0.556686051
obvod přes hýždě	0.724722952	1	0.695310163	0.666056209	0.636328407	0.121699907	0.677378443	0.910139129	0.557129213
obvod přes prsa	0.766210099	0.695310163	1	0.862693048	0.824943571	0.081692857	0.52371782	0.674254048	0.625750893
obvod pod prsy	0.735156403	0.666056209	0.862693048	1	0.854975674	0.085718949	0.536758815	0.632616832	0.624101264
šířka hrudníku	0.713223141	0.636328407	0.824943571	0.854975674	1	0.107265739	0.494823878	0.590360497	0.589734269
šířka ramene	0.039389923	0.121699907	0.081692857	0.085718949	0.107265739	1	0.142914813	0.076713162	0.093013004
obvod lýtka	0.4507909	0.677378443	0.52371782	0.536758815	0.494823878	0.142914813	1	0.726171765	0.485057028
obvod stehna	0.682207584	0.910139129	0.674254048	0.632616832	0.590360497	0.076713162	0.726171765	1	0.561858686
obvod krku	0.556686051	0.557129213	0.625750893	0.624101264	0.589734269	0.093013004	0.485057028	0.561858686	1

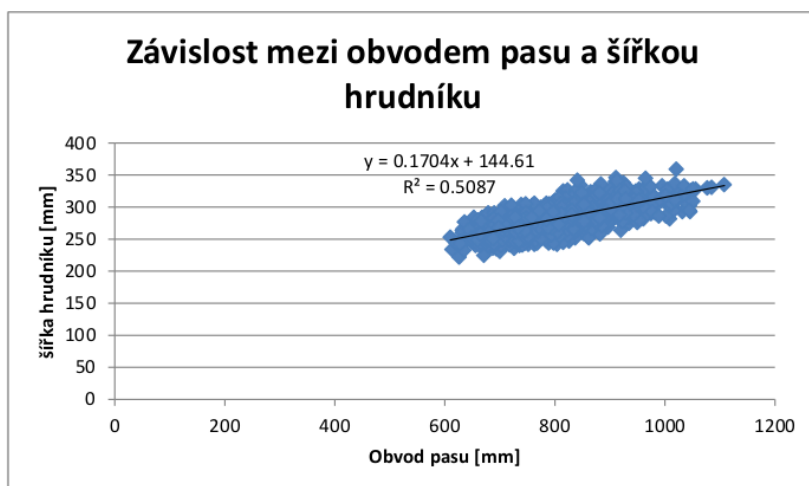
Obrázek 5.4: Naměřené hodnoty člověka, které budou použity pro hlídání fyziologických mezí.

Kromě tabulky lze závislost vidět i na grafech. Na obrázku 5.5 je bodový graf, který má na ose x obvod přes hýždě a na ose y obvod stehna. Čím více jsou body u sebe, tím je větší závislost mezi daty na ose x a na ose y. Graf je proložen lineární přímkou trendu. Rovnice přímky je vypsána vedle shluku bodů.  $R^2$ , který je uveden pod rovnicí přímky, značí hodnotu spolehlivosti.

Stejné typy grafu jsou i na obrázku 5.6 a obrázku 5.7, ovšem liší se v tom, jak jsou data na ose x a ose y na sobě závislá. Je vidět, že na obrázku 5.6 je shluk větší, tedy závislost je menší než u obrázku 5.5. A na posledním obrázku 5.7 je závislost dokonce téměř nulová.

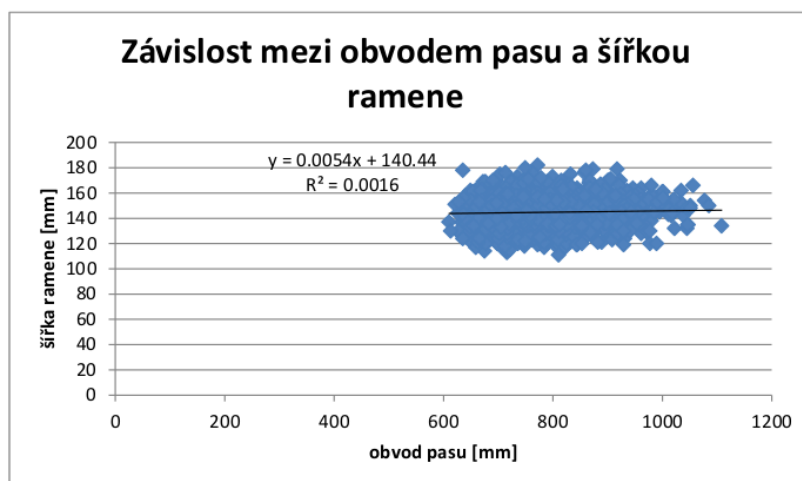


Obrázek 5.5: Bodový graf, který znázorňuje závislost mezi hodnotami na ose x a ose y.



Obrázek 5.6: Bodový graf, který znázorňuje závislost mezi hodnotami na ose x a ose y.





Obrázek 5.7: Bodový graf, který znázorňuje závislost mezi hodnotami na ose x a ose y.

## Percentil

Jak je vidět na obrázku 5.4, korelací jsem získal 9 poměrů, které budou hlídat fyziologické meze. Nyní bylo nutné se rozmyslet, zda-li použít metodu, ve které se počítají poměry mezi jednotlivými rozměry nebo si vybrat metodu PCA. Metoda přes poměry je na první pohled jednodušší, ale nebude nejspíše tak přesná jako metoda PCA. Je otázkou, jestli se vyplatí pouštět do složitější metody, jelikož Makehuman už sám o sobě má určité omezení. Tedy nedovolí uživateli jít do žádných extrémů, ve kterých by tato metoda mohla být neefektivní. Na základě tohoto faktu jsem se rozhodl, že naprogramuji metodu s poměry.

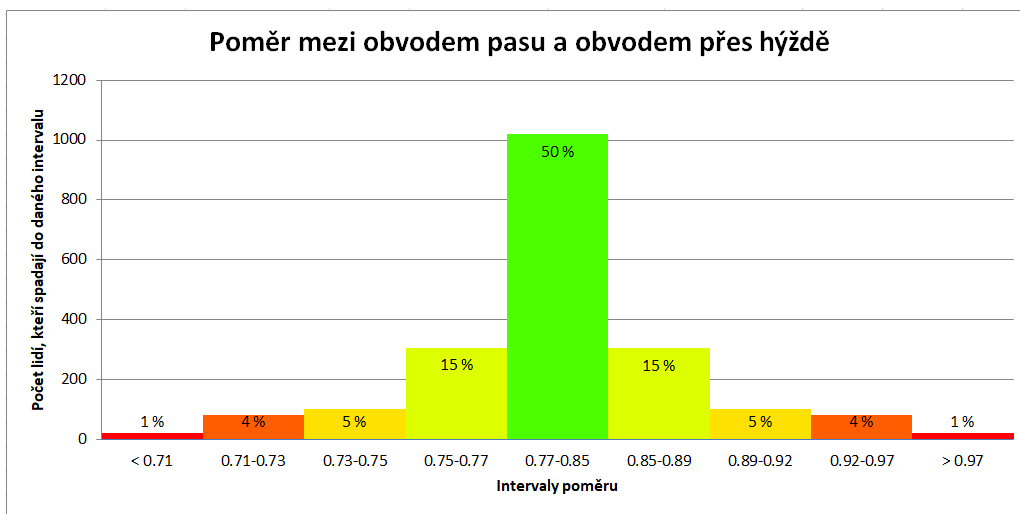
Nejprve musím získat hodnoty poměru, které budou určovat hranici mezi reálným a naopak nereálným poměrem daných rozměrů. K rozdělení hodnot jsem použil percentil, popsany v podkapitole 4.5.2.

Před počítáním percentilu je nutné nejprve určit procentní hranice reálnosti. Vybral jsem statisticky významná procenta a to: 1 %, 5 %, 25 %, 75 %, 90 %, 95 % a 99 %. K výpočtu byl použit Excel, konkrétně funkce PERCENTIL.INC(matice;k), kde matice je oblast dat (sloupec), kde se nachází hodnoty poměru a  $k$  definuje hodnotu percentilu z uzavřeného intervalu od 0 do 1. Tedy za  $k$  se dosadily procenta jako desetinné číslo (např. 1 % = 0.01). Výsledné hodnoty každého poměru označují hranice reálnosti.

Jak výpočet percentilu dopadl je možné vidět na obrázku 5.8, kde je sloupcový graf s poměrem mezi obvodem pasu a obvodem přes hýždě. Na ose x jsou intervaly poměru vypočítány pomocí percentilu. Na ose y je poté četnost lidí, kteří spadají do daného intervalu. Každý sloupec je označen

barvou, která se bude zobrazovat uživateli v samotné aplikaci Makehuman. Zelená barva značí, že daná postava patří mezi 50 % populace, tedy její rozměry jsou naprosto v souladu se statistickými hodnotami. Pokud se budeme dostávat mimo těchto 50 %, bude se barva měnit postupně ze zelené ve světle zelenou, žlutou, oranžovou až nakonec červenou. Ta definuje nereálnou postavu dle statistických měření.

Výsledky prvního testování (viz. podkapitola 6.2.1) ukázaly, že takto stanovené hranice nejsou vhodné, proto přišla z mé strany úprava hranic. Jak již je psáno v podkapitole 4.5.2, pokud nám nestačí procentuální vyjádření hranice, přichází v úvahu outliers. Jelikož se 1 % ukázalo vzhledem k dostupným datům jako příliš velká skupina, nahradil jsem ji hodnotami outliers.



Obrázek 5.8: Sloupcový graf, který vyjadřuje počet lidí, které spadají do daného intervalu poměru.

### 5.1.4 Programovací část

Vypočítané percentily z poměrů jednotlivých rozměrů jsem vložil do excelovského souboru. Vždy všechny percentily dvou závislých rozměrů do jednoho sloupce. Získané statistické údaje byly již připraveny k implementaci do aplikace Makehuman. Tudíž jsem přemýšlel, jak data načíst do kódu tak, abych s nimi mohl jednoduše pracovat.

Jelikož většina externích dat se do Makehuman vkládá z datového formátu JSON (JavaScript Object Notation), měl jsem tendenci vydat se touto cestou a převést data z Excelu do JSONu. Ale při hledání, jak data převést, jsem narazil na knihovnu Openpyxl, která umožňuje v Pythonu přímo načítat soubory ve formátu xls čixlsx, což jsou formáty Excelu.

Po nastudování této knihovny jsem ji importoval do aplikace Makehuman a otevřel pomocí ní soubor s připravenými daty. V této chvíli přišla na řadu otázka, jak data ze souboru číst. Chtěl jsem udělat algoritmus takový, aby fungoval pro všechna táhlítka najednou.

Jelikož každé táhlítko obsahuje název části těla, kterou deformuje, využil jsem této znalosti. V excelovském souboru s daty jsem do první buňky každého sloupce napsal název obou rozměrů, které sloupec obsahuje a oddělil je středníkem. V Pythonu jsem napsal kód, který daný soubor čte po sloupcích a vždy první buňku rozdělí podle středníku na dva řetězce. Každý řetězec znázorňuje jednu část těla, kterou poměr v tomto sloupci obsahuje. Dále se tedy testuje, zda jeden ze dvou řetězců se shoduje s názvem, které vlastní táhlítko. Pokud ne, přeskočí se na další sloupec a provádí se to samé. V opačném případě se provede poměr hodnot v aplikaci a výsledek tohoto poměru se porovná se statisticky získanými daty. Podle intervalu, do kterého vypočítaný poměr spadá, se daná táhlítka obarví. Když nějaká část těla je ovlivněna několika jinými rozměry, bere se vždy v úvahu ta nejhorší varianta. Například pokud poměr mezi obvodem pasu a obvodem přes hýždě bude patřit do červených hodnot (tedy nereálných), budou obě táhlítka červená a to i přes to, že například poměr mezi obvodem pasu a obvodem přes prsa bude v zeleném intervalu (tedy reálný).

## 5.2 Používané technologie

Kód byl psán v jazyce Python ve verzi 2.7, neboť nástroj MakeHuman byl vyvíjen v témže programovacím jazyce. Python je vyvíjen jako open-source, což znamená, že si každý může přečíst zdrojový kód. Python je dynamicky interpretovaný jazyk, tudíž se kód překládá až za běhu, čímž se případné chyby programu projeví teprve při spuštění. Výhodou je, že je jazyk díky tomuto přístupu jednodušší.

Ke zpracování dostupných antropometrických dat byl použit Excel, který patří do skupiny programů, kterým se říká tabulkové procesory nebo spreadsheety. Tabulkové procesory představují komplexní program pro řešení obecných vědecko-inženýrských výpočtů. Nabízejí nejen širokou škálu funkcí matematických, ale i statistických a finančních. Nezanedbatelná je rovněž schopnost grafické prezentace dat i možnost provádět složitější odborné analýzy či rozborů.

K testování, které je popsáno v následující kapitole, byl kromě Pythonu použit i formulář od Googlu, díky kterému jsem dostal zpětnou vazbu od respondentů a vygenerované grafy mi pomohly k vyhodnocení testování.

# 6 Testování

Po naprogramování algoritmu, který hlídá fyziologické meze, jsem jej otestoval. A to ze dvou hledisek:

- Časové a paměťové nároky
- Posouzení intuitivnosti a realističnosti dosažených výsledků

## 6.1 Časové a paměťové nároky

Jelikož jsem rozšiřoval již existující kód a načítal jeden excelovský soubor, který má 10kB, paměťové nároky se výrazně oproti již existujícímu kódu nezměnily.

Časové nároky byly testovány na notebooku Asus N551JQ-CN050H s parametry:

- Procesor: Intel Core i5-4200H (2.8 GHz)
- Paměť (RAM): 8 GB
- Grafická karta: NVIDIA GeForce with CUDA 845M (N15S-GT1R)
- Operační systém: Windows 10 (64bit)

Čas jsem otestoval pomocí timeru v Pythonu. Abych jej mohl použít, musel jsem nejdříve importovat knihovnu `time`. Poté jsem na začátek algoritmu vložil funkci `time()` z knihovny. Tato funkce vrátí čas v sekundách od epochy jako číslo s desetinnou čárkou. Z tohoto důvodu jsem na konci algoritmu použil tu samou funkci a následně oba časy od sebe odečetl. Odečtení těchto dvou časů mi dalo přesný čas, který aplikaci zabral můj algoritmus.

Průchod všech táhlítek algoritmem včetně jejich obarvení trvá na uvedeném notebooku v rozmezí od 14 do 22 ms.

Stejným způsobem jsem získal i čas potřebný k otevření excelovského souboru, ve kterém jsou statistické údaje, které hlídají fyziologické meze. Zde jsem se dostal na 9 až 10 ms. Aplikace nebyla algoritmem příliš zpomalena a tak umožňuje modelování lidské postavy v reálném čase.

## 6.2 Intuitivnost a realističnost výsledků

Narozdíl od paměťových a časových nároků, intuitivnost a realističnost výsledků nebyla řešena pomocí programovacího kódu. Jelikož je tato aplikace cílena pro veřejnost, byla testována přímo na lidech, kteří byli ochotni věnovat část svého času na hraní si s aplikací.

### 6.2.1 První testování

Při prvním testování bylo respondentů celkem 8. Byly to převážně ženy ve věku od 20 do 35 let, protože jsem chtěl provádět testování hlavně na cílové skupině.

Na základě mnou vytvořeného dotazníku s návodem k použití aplikace, měli tito lidé za úkol hrát si s aplikací. Dotazník obsahoval několik otázek s instrukcemi, které měl uživatel v aplikaci provést před tím, než odpověděl na otázku. Jako odpověď uživatel vybral jednu z předem připravených možností. Možnosti byly zpravidla 4.

Prvním úkolem pro uživatele bylo vytvořit pomocí táhlítek, které jsou v aplikaci, vizuálně reálnou postavu. Při vypracovávání této otázky bylo důležité, aby uživatel nesledoval barvy táhlítek a opravdu se soustředil pouze na postavu modelu.

Výsledek první otázky je vidět na obrázku 6.1. Z odpovědí lze vyčíst, že ve většině případů byla postava vyhodnocena algoritmem spíše jako nereálná. Tuto skutečnost si vysvětluji tak, že zpracovaná statistická data zkreslují hranice reálnosti. Tedy, že málokterá žena má postavu, která se podobá postavě ženy hlásící se do armády. Dalším problémem může být použitá metoda, tedy poměr rozměrů lidí (viz. podkapitola 4.5.2). Problém může být částečně i ve zvolených hranicích. Hranice nereálnosti nastavena na 1 % může být s ohledem na data velmi velkou skupinou lidí. Tuto hranici můžeme posunout na outliers, která jsou dána vztahem navrženým panem Tukey, který je popsán v podkapitole 4.5.2.

Druhá otázka byla zaměřena na ověření, že algoritmus označí postavu jako reálnou jen v případě, že je opravdu reálná. Podle grafu na obrázku 6.2 lze říci, že se uživatel může spolehnout na reálnost postavy, pokud jsou všechna táhlítka zelená.

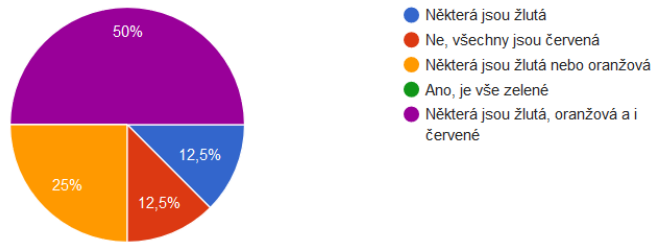
To už ovšem nelze říci o nereálnosti postavy. Pokud uživatel nastavil všechna táhlítka tak, aby byla červená, neznamenalo to vždy, že postava vypadala nereálně. Což je vidět na obrázku 6.3. Stejně jako u první otázky tento výsledek dávám za vinu nevhodným datům, použité metodě a nastavené hranici nereálnosti.

Ze čtvrté otázky (viz. obrázek 6.4) vyplývá, že uživatelům stačily použité barvy k přechodu od postavy reálné k nereálné.

Pátá otázka byla pro kontrolu, že přidáný návod byl pochopitelný a uživatel věděl, jak s aplikací zacházet (viz. obrázek 6.5).

Vytvořte prosím postavu pomocí táhlítek, která je ve skutečnosti reálná. Je v aplikaci také reálná? (Nemá červené táhlítko?)

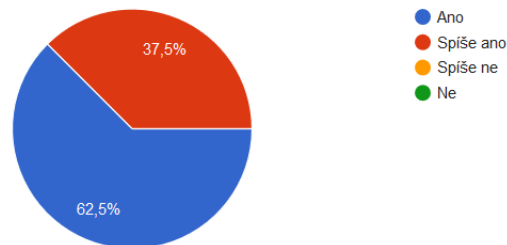
8 odpovědí



Obrázek 6.1: Koláčový graf první otázky z dotazníku.

Nastavte prosím táhlítko tak, aby byla všechna zelená. Je tato postava vizuálně opravdu reálná?

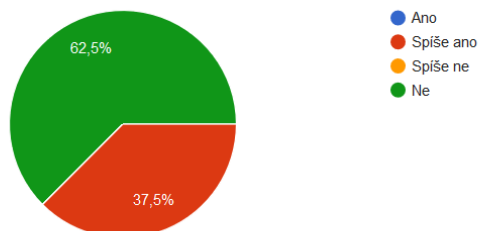
8 odpovědí



Obrázek 6.2: Koláčový graf druhé otázky z dotazníku.

Nastavte prosím táhlítka tak, aby byla všechna červená. Je tato postava vizuálně opravdu nereálná?

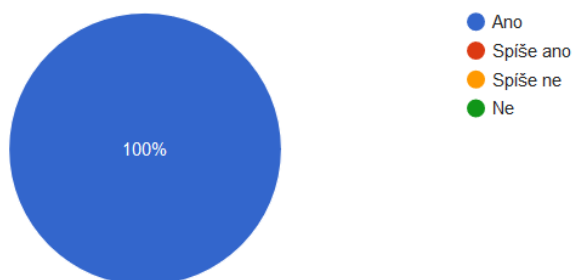
8 odpovědí



Obrázek 6.3: Koláčový graf třetí otázky z dotazníku.

Je počet barev, které definují reálnost postavy dostačující?

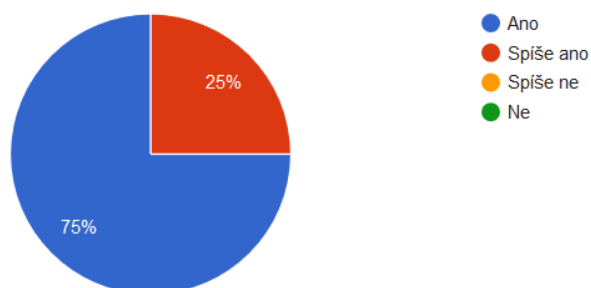
8 odpovědí



Obrázek 6.4: Koláčový graf čtvrté otázky z dotazníku.

Je ovládání aplikace na základě návodu jednoduché?

8 odpovědí



Obrázek 6.5: Koláčový graf páté otázky z dotazníku.

## 6.2.2 Druhé testování

Druhé testování bylo prováděno na stejné metodě a stejných datech jako při prvním testování, ale na upravených hranicích nereálnosti.

Ve druhém testování jsem se nejvíce zaměřil na otázky, které dopadly v prvním případě nepříznivě, ale nechal jsem vyplnit respondenty všechny otázky, abych si ověřil, že jsem při úpravě nepoškodil jinou část algoritmu.

Ostatní otázky dopadly vesměs se stejnými procenty jako v prvním případě, proto je zde uvádět nebudu. V prvním testování nedopadly dobře dvě otázky, které se po změně hranic zlepšily.

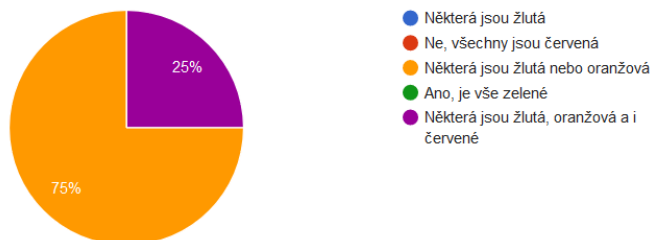
Na obrázku 6.6 je vidět, že když uživatel vytvořil dle svého subjektivního názoru postavu, která je reálná, aplikace již ve většině případů nezobrazovala postavu jako úplně nereálnou. Je jasné, že se změnou hranice 1 % na outliers nemohlo dosáhnout toho, že by byla všechna táhlítka zelená.

Graf na obrázku 6.7 ukazuje, že posunutím hranice nereálnosti na outliers se docílilo toho, že už respondent jednoznačně neuvedl, že postava je nereálná, ale už váhal mezi spíše reálnou a spíše nereálnou postavou. Není to ještě ideální výsledek, ale posun k lepšímu oproti prvnímu testování je.

Negativním faktem druhého testování je, že se mi z časových důvodů podařilo sehnat pouze 4 respondenty, což mohlo částečně testování ovlivnit.

Vytvořte prosím postavu pomocí táhlítek, která je ve skutečnosti reálná.  
Je v aplikaci také reálná? (Nemá červené táhlítka?)

4 odpovědi

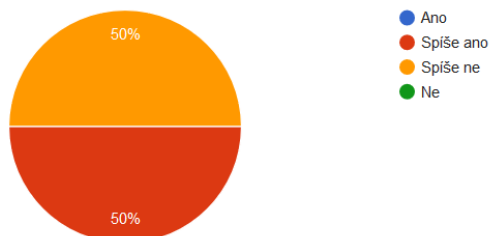


Obrázek 6.6: Koláčový graf první otázky z dotazníku (druhé testování).



Nastavte prosím táhlítka tak, aby byla všechna červená. Je tato postava vizuálně opravdu nereálná?

4 odpovědi



Obrázek 6.7: Koláčový graf třetí otázky z dotazníku (druhé testování).

### 6.3 Prostor pro zlepšení

Jelikož testování ukázalo, že posunutím hranice nereálnosti na outliers se výsledky částečně zlepšily, ale stále nedosahují optimálních hodnot, bylo by proto dobré vyzkoušet metodu PCA, která by mohla algoritmus vylepšit a získat tak přesnější informace o reálnosti postavy. Mimo metodu PCA je možné vyzkoušet Mahalanobisovu vzdálenost.

Kromě metod je zapotřebí se zamyslet i nad antropometrickými daty, se kterými se pracovalo. Již při vypracovávání návrhu metody jsem se obával, že vojenská data zkreslí statistické hodnoty, které poté ovlivní hlídání fyziologických mezí. Dle výsledků testování se mé obavy potvrdily. Proto by bylo do budoucna dobré získat potřebné rozměry lidí od široké veřejnosti.

Dále by bylo možné uživateli zamezit posunout táhlítko mimo reálnou mez. Tedy uživatel by neměl možnost vytvořit nereálnou postavu. To by se ovšem některým uživatelům nemuselo líbit. Existují případy, kdy můžeme z nějakého důvodu chtít takovou postavu vytvořit. To je potom otázkou, pro jakou skupinu lidí se aplikace vyvíjí. Pokud by byla cílena pro fitness centra, tato vlastnost by byla pravděpodobně žádoucí. Naopak u filmů se dost často vytváří nereálné postavy, v těchto případech by zamezení nemělo smysl.

## 7 Závěr

V této práci došlo k rozšíření openSource aplikace Makehuman o algoritmus, který hlídá fyziologické meze, což bylo provedeno na základně statisticky zpracovaných antropometrických dat a uživatel je o tom informován pomocí barev. Kromě mě se o hlídání fyziologických mezí pokusili vývojáři nástroje Body Talk, který je uveden v teoretické části (viz. podkapitola 2.5). Body Talk neinformuje uživatele o reálnosti postavy pomocí barev jako moje metoda, ale nedovolí mu takovou postavu vůbec vytvořit.

Výsledky testování rozšířené aplikace Makehuman ukazují, že paměťové a časové zatížení naprogramovaného algoritmu je malé a neovlivňuje tak výrazně plynulost aplikace. Dále ukazují, že algoritmus nám označí správně reálnou postavu. Před opravou hranic nám mnoho dalších reálných postav je označeno jako nereálné. To mohlo být následkem několika faktorů. Jeden z faktorů byl, že se používají vojenská data, tedy postavou velmi blízké osoby. To mohlo být při označování nereálné postavy při hranici 1 % lidí nevhodné. Možným řešením bylo posunout tuto hranici na takzvaná outliers, která jsou dána vztahem navrženým panem Tukey.

Tuto domněnku jsem se rozhodl zrealizovat. Posunul jsem tedy hranici 1 % na outliers, což přineslo zlepšení, ale stále se nedá říci, že by algoritmus označil vždy všechny postavy správně.

Zůstávají tedy další faktory, které by mohly algoritmus vylepšit. Jedním z faktorů mohou být data, která jsou získána jen od určité skupiny lidí. Hlavním cílem další práce by tedy mělo být získání vhodných dat. Vhodnými daty se rozumí naměřené potřebné rozměry žen ze široké veřejnosti ve věku od 20 do 35. Tato data poté použít v naprogramovaném algoritmu a zjistit tak, zda-li selepší výsledky rozhodování o reálnosti postavy.

Dalším možným faktorem je použitá metoda, tj. poměr mezi rozměry postavy, a proto by bylo vhodné v následující práci vyzkoušet metodu PCA, která by taktéž mohla algoritmus vylepšit a získat tak přesnější informace o reálnosti postavy. Asi nejpřesnějším řešením by bylo použití Mahalanobisovy vzdálenosti.

# Literatura

- [1] *Korelace* [online]. 2018. [cit. 22. 4. 2018]. Dostupné z:  
<https://cs.wikipedia.org/wiki/Korelace>.
- [2] *Co je to percentil* [online]. 2018. [cit. 24.4.2018]. Dostupné z:  
<https://www.childrenbmi.com/cs/faq/co-je-to-percentil/>.
- [3] *Testování statistických hypotéz* [online]. 2016. [cit. 19. 9. 2016]. Dostupné z:  
[https://www.wikiskripta.eu/w/Testov%C3%A1n%C3%AD\\_statistick%C3%BDch\\_hypot%C3%A9z](https://www.wikiskripta.eu/w/Testov%C3%A1n%C3%AD_statistick%C3%BDch_hypot%C3%A9z).
- [4] *Vícerozměrná analýza* [online]. 2018. [cit. 2.3.2018]. Dostupné z:  
[http://www.trilobyte.cz/downloadfree/qcemanual/multivar\\_an.pdf](http://www.trilobyte.cz/downloadfree/qcemanual/multivar_an.pdf).
- [5] *Outliers in SPSS* [online]. 2017. [cit. 12.2.2017]. Dostupné z:  
<http://www.unige.ch/ses/sococ/cl/spss/concepts/outliers.html>.
- [6] *Studentův t-test* [online]. 2016. [cit. 30. 11. 2016]. Dostupné z:  
[https://www.wikiskripta.eu/w/Student%C5%AFv\\_t-test](https://www.wikiskripta.eu/w/Student%C5%AFv_t-test).
- [7] HOŘEJŠÍ, J. *Lidské tělo: Srozumitelný a zevrubný průvodce po strukturách a funkcích lidského organismu*. Gemini (ČR), 1991. ISBN 80-85265-13-3.
- [8] JU, T. – SCHAEFER, S. – WARREN, J. Mean Value Coordinates for Closed Triangular Meshes. *ACM Trans. Graph.* July 2005, 24, 3, s. 561–566. ISSN 0730-0301. doi: 10.1145/1073204.1073229. Dostupné z:  
<http://doi.acm.org/10.1145/1073204.1073229>.
- [9] KASAP, M. – MAGNENAT-THALMANN, N. Parameterized Human Body Model for Real-Time Applications. In *Proceedings of the 2007 International Conference on Cyberworlds, CW '07*, s. 160–167, Washington, DC, USA, 2007. IEEE Computer Society. doi: 10.1109/CW.2007.51. Dostupné z:  
<https://doi.org/10.1109/CW.2007.51>. ISBN 0-7695-3005-2.
- [10] LE, B. H. – DENG, Z. Interactive Cage Generation for Mesh Deformation. In *Proceedings of the 21st ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games, I3D '17*, s. 3:1–3:9, New York, NY, USA, 2017. ACM. doi: 10.1145/3023368.3023369. Dostupné z:  
<http://doi.acm.org/10.1145/3023368.3023369>. ISBN 978-1-4503-4886-7.
- [11] MASUDA, H. – YOSHIOKA, Y. – FURUKAWA, Y. Preserving Form-features in Interactive Mesh Deformation. In *Proceedings of the 4th International*

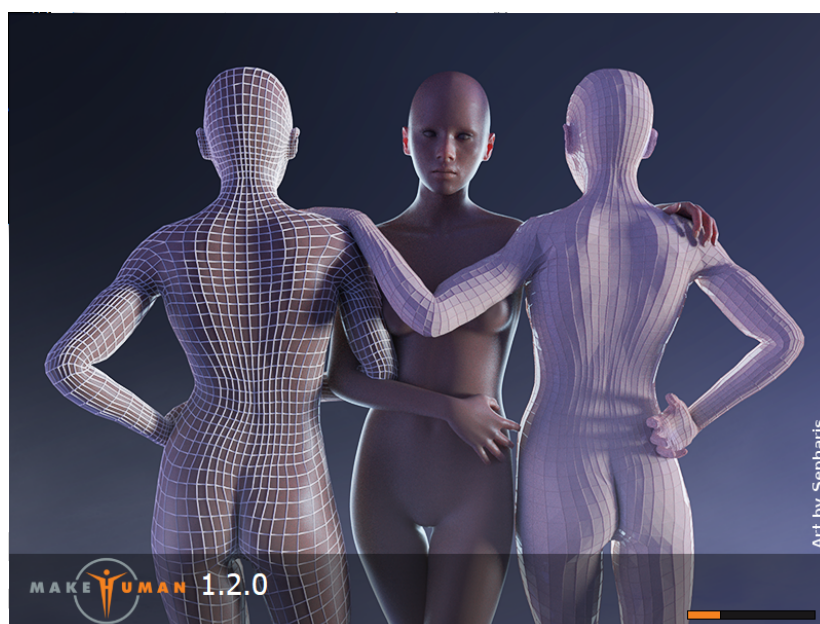
*Conference on Geometric Modeling and Processing, GMP'06*, s. 207–220, Berlin, Heidelberg, 2006. Springer-Verlag. doi: 10.1007/11802914\_15.  
Dostupné z: [http://dx.doi.org/10.1007/11802914\\_15](http://dx.doi.org/10.1007/11802914_15). ISBN 3-540-36711-X, 978-3-540-36711-6.

- [12] STREUBER, S. et al. Body Talk: Crowdshaping Realistic 3D Avatars with Words. *ACM Trans. Graph. (Proc. SIGGRAPH)*. July 2016, 35, 4, s. 54:1–54:14.

# Příloha A

Přílohou bakalářské práce je DVD, na kterém je aplikace Makehuman s přidáním algoritmem, který hlídá fyziologické omezení. Aplikace Makehuman se spustí pomocí souboru run.bat, který se nachází ve složce MakehumanApp/MakehumanApp/makehuman.

Po spuštění se zobrazí načítací obrázek s progress barem.(viz. obrázek 7.1)

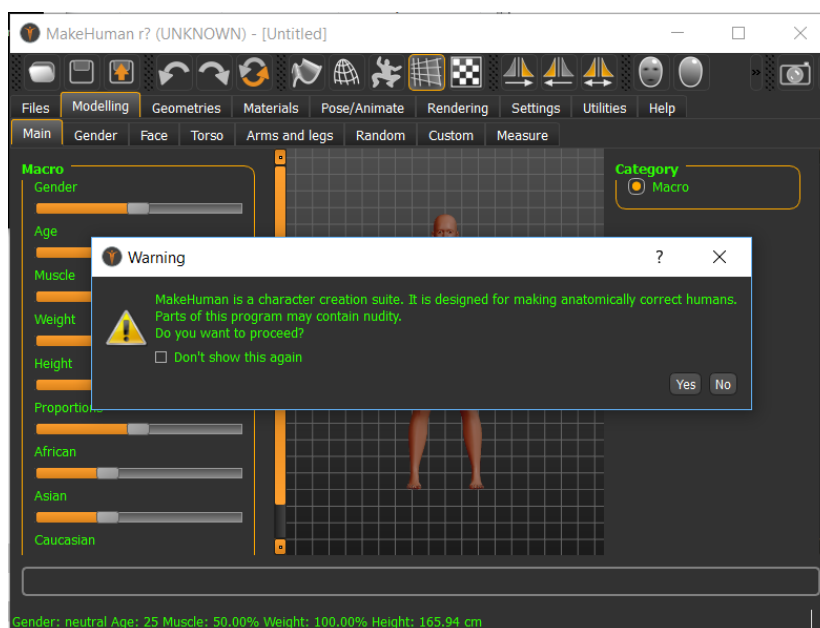


Obrázek 7.1:

Po chvilce čekání se načte aplikace a zobrazí se úvodní obrazovka. Nejprve vyskočí varování (viz. obrázek 7.2), které říká, že aplikace je určena pro vytváření anatomicky správných lidí a že části tohoto programu mohou obsahovat nahotu. Tuto hlášku je potřeba potvrdit kliknutím na tlačítko Yes. V opačném případě by se aplikace vypnula. Po stisknutí tlačítka Yes se zpřístupní aplikace (viz. obrázek 7.3).

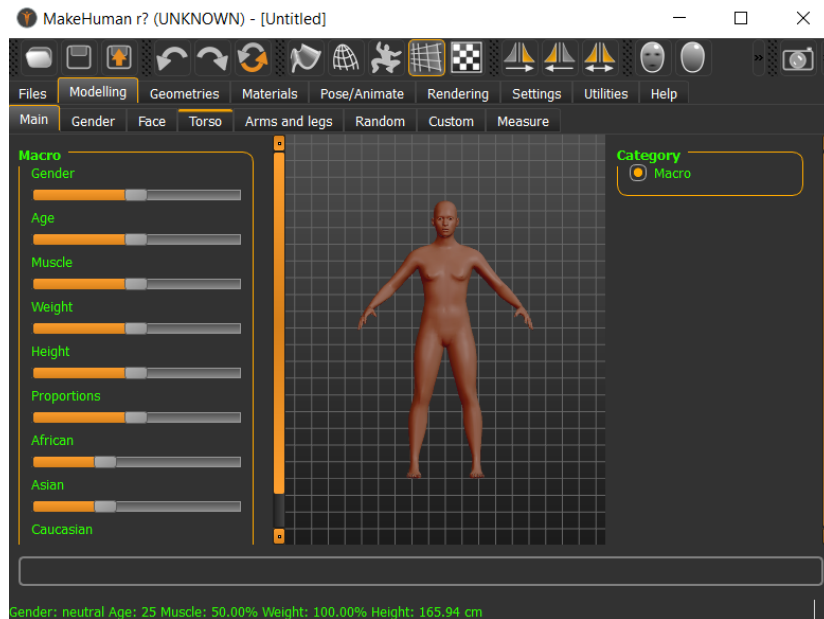
Jelikož jsou fyziologické meze hlídány pouze na ženách, je zapotřebí si zvolit postavu ženy. Ta se nastaví posunutím táhlítka Gender úplně doleva (viz. obrázek 7.4). Další nastavení parametrů záleží na uživateli, pouze věk by bylo dobré udržet v hranici od 20 do 35 let. Tedy neposouvat táhlítko Age příliš na jednu nebo druhou stranu.

Až zde budou nastaveny všechny parametry, přepněte se do záložky Me-

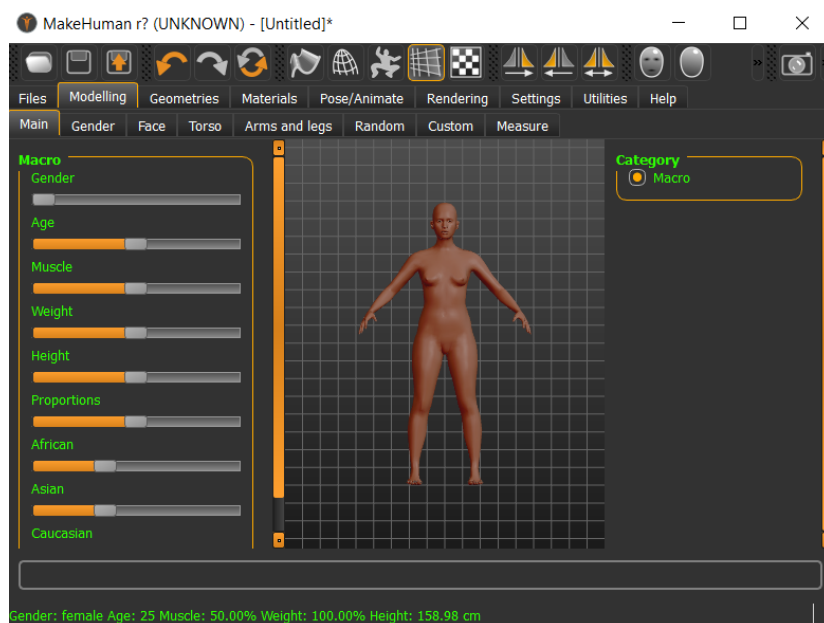


Obrázek 7.2:

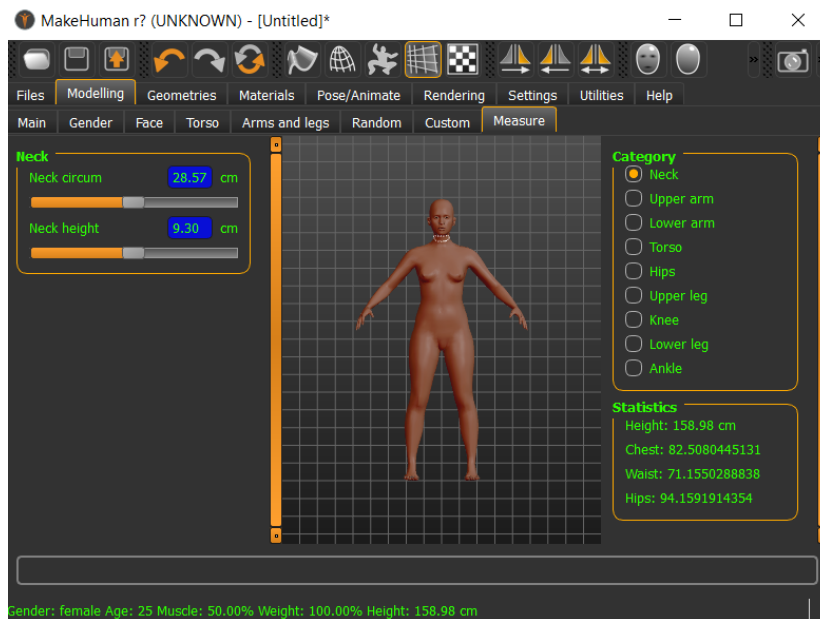
asure (viz. obrázek 7.5), která se nachází v horní části okna. Na této záložce je několik kategorií (oblastí těla), mezi kterými se lze přepínat. Hlídané rozměry těla jsou v kategorii Torso (Front chest dist, Bust circum, Underbust circum, Waist circum) (viz. obrázek 7.6), Hips (Hips circum), Upper leg (Thigh circum.) a Lower leg (Calf circum). Uvedené části těla jsou hlídány a mění barvu podle reálnosti postavy (Zelená = reálná, červená = nereálná). Při změně jiných táhlítek a dosažení tak nereálné postavy jiným způsobem (například nepoměr výšky nohou k tělu), není toto hlídáno, jelikož je předpokládáno, že výšku člověk v dospělosti již nemění, a proto jsou hlídány jen parametry těla, které se mění na dospělé osobě. Tedy obvody a šířky jednotlivých částí těla.



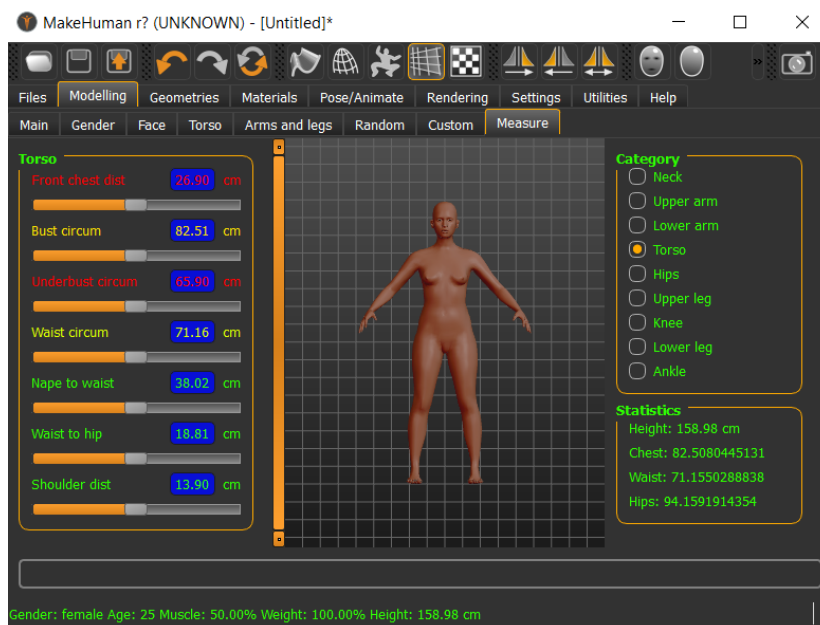
Obrázek 7.3:



Obrázek 7.4:



Obrázek 7.5:



Obrázek 7.6:



# Příloha B

Při programování došlo k úpravě dvou souborů a to:

- **0\_modeling\_a\_measurement.py**, který se nachází ve složce MakehumanApp/MakehumanApp/makehuman/plugins.
- **modifierslider.py**, který se nachází ve složce MakehumanApp/MakehumanApp/makehuman/lib.

## **0\_modeling\_a\_measurement.py**

V tomto souboru byla ve třídě Ruler přidána HashMapa s identifikátorem **my\_measures**. Ve funkci **getMeasure** jsou do HashMapy načítány jednotlivé rozměry.

## **modifierslider.py**

V modifierslider.py je přidáno několik funkcí.

### **controlMeasure**

Funkce **controlMeasure** je volána z funkce **update**, do které postupně vstupují všechna táhlítka při každé změně jakéhokoliv táhlítka a postupně jsou všechna upravována.

Samotná funkce nejprve načte excelovský soubor se statisticky zpracovanými antropometrickými daty, na základě kterých se hlídají fyziologické meze. Tento soubor se načte do globální proměnné.

V excelovském souboru je vždy v první buňce každého sloupce název obou rozměrů, které sloupec obsahuje a rozměry jsou oddělené středníkem. První buňka je rozparsována podle středníku na dva řetězce. Každý řetězec znázorňuje jednu část těla, kterou poměr v tomto sloupci obsahuje. Dále se tedy testuje, zda jeden ze dvou řetězců se shoduje s názvem, které vlastní táhlítko. Pokud ne, přeskočí se na další sloupec a provádí se to samé. V opačném případě se zavolá funkce **getFrequencyBody**.

### **getFrequencyBody**

Zde se provádí poměr mezi danými parametry postavy. Pokud je poměr menší než porovnávaná hodnota z excelovského souboru, je návratová hodnota 1, která říká, že se má změnit barva táhlítka.

### **setColorSliderUp**

Pokud je návratovou hodnotou funkce `getFrequencyBody` 1, funkce `controlMeasure` zavolá funkci `setColorSliderUp`, která mění barvy táhlítek za předpokladu, že již táhlítko nemělo horší výsledek poměru s jinou také na sobě závislou veličinou.