



Degree Project in Technology

First cycle, 15 credits

# **A New Approach for Positioning Human Body Models Utilising the 3D-Graphics Program Blender**

**JESPER EIDERBÄCK**

**FELIX JAHNKE**



This project was performed in collaboration with  
KTH Royal Institute of Technology  
Supervisor at KTH Royal Institute of Technology, Division of Neuronic Engineering: Xiaogai  
Li

## A New Approach for Positioning Human Body Models Utilising the 3D-Graphics Program Blender

Ett nytt tillvägagångsätt för att positionera mänskliga  
kroppsmodeller med hjälp av 3D-grafikprogrammet  
Blender

JESPER EIDERBÄCK  
FELIX JAHNKE

Degree project in medical engineering  
First level, 15 credits  
Supervisor at KTH: Tobias Nyberg, Mattias Mårtensson  
Examiner: Mats Nilsson

KTH Royal Institute of Technology  
School of Engineering Sciences in Chemistry, Biotechnology and Health  
Hälsövägen 11 C  
SE-141 57 Huddinge, Sweden  
<http://www.kth.se/cbh>

2023



## Abstract

A finite element human body model (FE HBM) is a detailed virtual model of the human body that, for example, is used for simulating traffic accidents. A problem with HBMs is that there is no simple way to position the HBMs in non-standard positions. As different postures during an impact will affect the body in different ways it is vital to have the ability to position the HBMs. In this project it was investigated if it is possible to position a HBM from THUMS, by first positioning only the skin and skeleton, as control points, in the 3D-graphics program Blender. Thereafter a radial basis function interpolation is utilised to morph the rest of the HBM into the new position. The results indicate that in theory, it is possible to position a HBM using a 3D-graphics software. However, the method developed in this project resulted in a disfigurement of the morphed model. The disfigurement is possibly due to the change in distance between the skin and skeleton when positioning those body parts in Blender.

**Keywords:** *Human Body Model, HBM, positioning, THUMS, Blender, Radial basis function.*



## Sammanfattning

En finit element människokroppsmodell (FE HBM) är en detaljerad virtuell modell av människokroppen som exempelvis används för att simulera trafikolyckor. Ett problem med HBM:er är att det inte finns något enkelt sätt att positionera dem i annat än standardpositioner. Eftersom olika kroppsställningar påverkar kroppen på olika sätt under en kollision är det viktigt att ha möjlighet att kunna positionera en HBM. I detta projekt undersöktes om det är möjligt att positionera en HBM från THUMS, genom att först positionera endast huden och skelettet, som kontrollpunkter, i 3D-grafikprogrammet Blender. Därefter användes en radiell basfunktionsinterpolation för att flytta resten av HBM till den nya positionen. Resultaten indikerar att det är möjligt att positionera en HBM med hjälp av ett 3D-grafikprogram. Metoden som utvecklades i detta projekt resulterade dock i en deformation av den positionerade modellen. Deformationen beror möjligen på att avståndet mellan hud och skelett ändrades vid positioneringen av dessa kroppsdelar i Blender.

**Nyckelord:** *Mänsklig kroppsmodell, HBM, positionering, THUMS, Blender, Radiell Basfunktion*



# Contents

1	Introduction .....	1
1.1	Aim.....	1
2	Background .....	3
2.1	The human body model.....	3
2.2	The HBM positioning challenge .....	3
2.3	THUMS .....	3
2.4	LS-PrePost and LS-DYNA .....	4
2.5	Blender .....	4
2.6	Radial basis function .....	5
2.7	Element quality .....	7
3	Method .....	9
3.1	Pre-processing of THUMS-model in LS-PrePost .....	9
3.2	Rigging of skeleton and skin in Blender .....	9
3.3	Weight-painting in Blender .....	10
3.4	Positioning tests of rigged AF05 V4.02 Pedestrian model .....	11
3.5	Decimation and positioning of model before interpolation. ....	11
3.6	Interpolation and element quality check .....	12
4	Results .....	13
4.1	Pre-processing of THUMS-model in LS-PrePost .....	13
4.2	Positioning tests of the rigged adult model .....	13
4.3	Morphed HBMs by RBF interpolation.....	14
4.4	Element quality check .....	18
5	Discussion .....	21
5.1	Rigging, weight painting and positioning tests in Blender .....	21
5.2	Decimation and positioning in Blender.....	22
5.3	Morphing HBMs by RBF interpolation .....	22
5.4	Final thoughts and future development.....	23
6	Conclusion.....	25
7	References .....	27
Appendices		
Appendix 1: setIKstrech.py		



# 1 Introduction

As of 2022 an estimated number of 1.35 million people die and up to 50 million people are injured in traffic accidents each year [1]. Traffic accidents is the leading cause of death for children and young adults, as well as the eighth leading cause of death for all age groups [1]. Enhancement of vehicle safety, through improvements in both protection during accidents and in accident-avoidance systems, is one way to reduce these numbers.

Traditionally traffic accident simulations have been done with physical tests, involving mechanical crash test dummies. But these physical dummies are both limited in their composition and more costly than a virtual model [2]. In later years virtual simulations of traffic accidents with finite element (FE) human body models (HBMs) have become more common in traffic safety studies, both in accident simulations [3-4] and in accident-avoidance system testing [5]. FE HBMs have an advantage over physical test dummies in that they mimic the human body more precisely [6]. Consequently, the effect an impact has on the human body can be studied in greater detail using an HBM compared to a classical test dummy.

There are several different HBMs, VIVA+ [7], THUMS [8], PIPER [2], SAFER [3][9], to name a few. A problem with HBMs is that it is challenging to positioning the HBMs in nonstandard positions. One approach to solve the positioning problem is to first position certain control points of the model and then interpolate a function to morph the rest of the HBM into the new position [10].

In this project a new approach to the positioning challenge was tested, by positioning the skin and skeleton of two different THUMS V4 HBMs in the 3D-graphics program Blender. After positioning, the skin and skeleton were used as control points in a radial basis function (RBF) interpolation to morph the HBM into the position.

## 1.1 Aim

The aim of this project was to develop a new method for positioning human body models via the 3D-graphics program Blender version 3.4.1 combined with an RBF interpolation.



## 2 Background

When aspiring to improve vehicle safety, it is important to understand the effect a traffic accident will have on the human body. This understanding can be gained through crash tests, wherein a model is exposed to a simulated collision to study potential injuries. Traditionally crash test has been conducted with physical crash test dummies. However, these tests are both expensive and limited in their ability to provide detailed information about the effect an impact has on the human body [11]. These limitations arises from the fact that physical test dummies is limited to predetermined characteristics such as weight, height, age and other compositions such as skin elasticity, muscle mass, fat tissue et cetera [11].

In this chapter, the human body model and its positioning challenge is introduced in sections 2.1 and 2.2. A short introduction to Toyotas human body models, THUMS, is provided in section 2.3, and an introduction to the main software used, LS-PrePost and Blender, in this project are provided in section 2.4 and 2.5 respectively. Lastly, radial basis function interpolation and element quality checks are explained in section 2.6 and 2.7 respectively.

### 2.1 The HBM

An FE HBM refers to a fully functional virtual model of the human body, containing all the essential parts of the body such as skin, bones, organs, ligaments et cetera [6]. An FE HBM is made up of a mesh [12]. The mesh is composed of nodes (vertices), connected by a line, called an edge, and the resulting geometrical shape is called an element (face) [13]. The mesh is used to implement the finite element method (FEM) in the simulations [14].

### 2.2 The HBM positioning challenge

A human's posture during a traffic accident can affect the consequences of the impact [15]. To gain more insight in how different positions affect the human body in traffic accidents, simulations in different positions must be done. However, the positioning of a FE HBM is a challenging task, since unlike a physical dummy, the positioning of an HBM is not straightforward, due to the deformation and interactions of the different parts of the finite element model.

A solution to the positioning problem has been developed in the PIPER project (Position and Personalize Advanced Human Body Models for Injury Prediction), during which a graphical user interface for the personalization and positioning of HBMs was created [16]. However, the PIPER tool solution is not as simple as importing an HBM into a positioning tool and starting the positioning process. It requires additional files with metadata that contain landmarks and control points [17].

### 2.3 THUMS

In this project HBMs are downloaded from THUMS (Total HUMAN Model for Safety). THUMS are HBMs developed by Toyota which was made freely available in January 2021 [18]. THUMS provides multiple HBMs consisting of bones, ligaments, muscles and organs [8]. Different parts of the THUMS model *AF05 V4.02 Pedestrian* are shown in Figure 1, below, where it also can be seen that the HBM is built up with different colours to make it easier to differentiate between the different parts of the model.

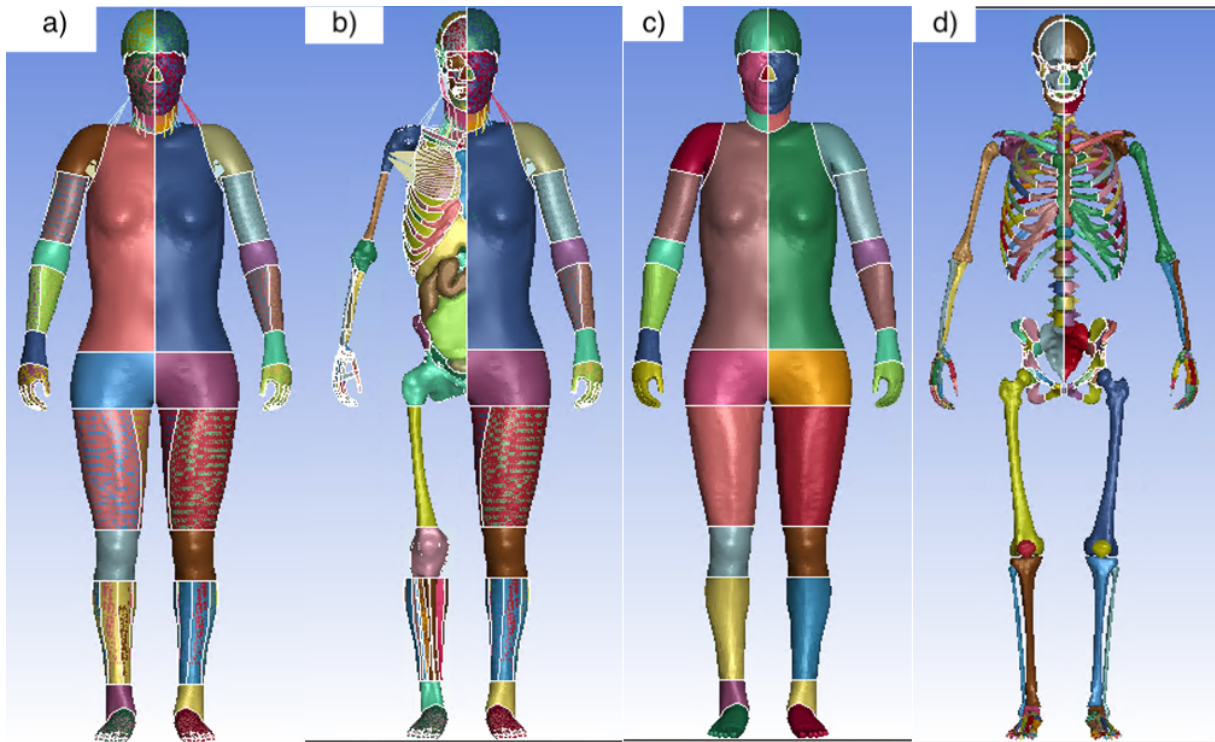


Figure 1. The THUMS AF05 V4.02 Pedestrian model: a) Complete model b) Organs and skeleton on one side visible c) Only skin d) Only skeleton

## 2.4 LS-PrePost and LS-DYNA

LS-PrePost and LS-DYNA is software provided by ANSYS. LS-PrePost is used for the creation and editing of FE models and visualization of results of simulations in LS-DYNA which is a software where physical simulation can be performed with the finite-element method (FEM) [19]. In this project LS-PrePost is used for the extraction of the skin and skeleton from an HBM, and for visualization of HBMs morphed to new positions.

## 2.5 Blender

Blender is a free 3D-graphics software. In this project Blender is used to rig (see *Rigging* below), and then position, the skeleton and skin from an HBM, to make them move together. Below are some Blender terms.

**Scene:** The scene in Blender is the graphical user interface where the objects being worked with can be displayed and manipulated.

**Modes:** There are multiple modes in Blender where different tasks can be performed. The most common modes are Object-mode, Edit-mode, Weight Paint-mode and Pose-mode.

**Rigging:** Rigging is a term which means that controls are added to an object so that the object can be manipulated [20].

**Rigify:** Rigify is an add-on that, if activated, provides access to premade bone models and rigging functions [20]. The premade bone-model used in this project is called “Basic Human (Meta-Rig)” and is referred to as the “meta-rig”. Rigify has a built-in function, “Generate Rig”, which generates steering controls for the rig [20].

**Weight painting:** How much a part (node) of a rigged model will move with a specific control is determined by how much weight (0 to 1) that is associated with the specific node [20]. In weight painting, the weights of the nodes are represented by a colour scale from blue

to red, where blue means 0 (minimum) weight and red means 1 (maximum) weight [20]. The weights of the nodes can, through weight painting, be changed by painting them in the colour that represents the desired weight.

**Decimate:** Blender has a built-in modifier for decimation of the model called “decimate”. The modifier decreases the number of faces in the model by a chosen ratio, for example a ratio of 0.1 decimates the number of faces to 10% of the original number of faces.

**Parenting:** An object can be set to another objects parent. The object who gets a parent is called a child, of the parent. After parenting, the child object will follow the movements of the parent object.

### IK and FK

Inverse kinematic (IK) and forward kinematic (FK) are two different ways that a rigged object can be moved in Blender. With FK, each part of the rigged object is moved individually. With IK, parts higher up in a hierarchy will move automatically when one part of the object is moved. For example, if the hand of a rigged skeleton is moved with IK controls, then the whole arm will follow, automatically. With FK, the hand can be rotated in relation to the arm, but the arm must be moved for the hand to move in space.

### IK stretch

IK stretch is a property that can be set between 0 and 1 for an object that has been rigged with the rigify function “Generate Rig”. With IK stretch set to greater than 0, the rigged object will stretch when moved with the IK controls. The object will stretch the most when IK stretch is set to 1. For example, for a rigged skeleton, IK stretch will make the bones longer when the skeleton is moved with IK controls.

## 2.6 Radial basis function

After rigging and positioning the skin and skeleton in Blender, the nodes of the skin and skeleton are utilised to morph the full HBM into the new position. In this project, a premade executable file (krbf.exe 2023 v1.9.4, provided by project supervisor Xiaogai Li) are used for the morphing by a radial basis function interpolation.

A radial basis function (RBF) is a mathematical algorithm that can be used to approximate a function  $s(\mathbf{x})$  given the datapoints  $(\mathbf{x}_j, f(\mathbf{x}_j))$ . The main part of an RBF is the radial function  $\varphi(r)$  which is a function depending on the Euclidian distance,  $r = \|\mathbf{x} - \mathbf{y}\|$ , between the input vectors. See Table 1 for some examples of radial functions [21].

Table 1. Four examples of Radial functions,  $\varepsilon$  is a constant.

Name	$\varphi(r)$
Linear spline	$r$
Cubic spline	$r^3$
Gaussian	$e^{-(\varepsilon r^2)}$
Thin plate spline	$r^2 \ln(r)$

An RBF in its simplest form is given by Equation ( 1 ) and ( 2 ) [21].

$$f(\mathbf{x}) \approx s(\mathbf{x}) = \sum_{i=1}^n w_i \varphi(\|\mathbf{x} - \mathbf{x}_i\|) \quad (1)$$

$$y_j \approx s(\mathbf{x}_j) = \sum_{i=1}^n w_i \varphi(\|\mathbf{x}_j - \mathbf{x}_i\|), \quad j = 1, 2, \dots, n \quad (2)$$

where  $w_i$  are weights.

$\varphi(\|\mathbf{x}_j - \mathbf{x}_i\|) = \varphi_i(\mathbf{x}_j)$  is the radial function.

If we define  $\mathbf{w}$ ,  $\mathbf{y}$  and  $\mathbf{H}$  as:

$$\mathbf{w} = [w_1, \dots, w_n]^T \quad \mathbf{y} = [y_1, \dots, y_n]^T \quad \mathbf{H} = \begin{bmatrix} \varphi_1(\mathbf{x}_1) & \dots & \varphi_n(\mathbf{x}_1) \\ \vdots & \ddots & \vdots \\ \varphi_1(\mathbf{x}_n) & \dots & \varphi_n(\mathbf{x}_n) \end{bmatrix}$$

then Equation ( 2 ) can then be written as:

$$\mathbf{y} = \mathbf{H}\mathbf{w} \quad (3)$$

With  $\mathbf{y}$  and  $\mathbf{H}$  known, the weight coefficients can be solved for from Equation ( 3 ).

In this project an RBF is used to interpolate a function that maps nodes from a baseline position to a target position, which are represented by vectors in  $R^3$ . The nodes of the baseline position are the nodes in the original HBM, and the target nodes are the same nodes as in the baseline position but moved to a new position. Only the skin and skeleton nodes are used to build the function. The final function are then used to move the rest of the HBM to the new position according to Equation ( 4 ).

$$\mathbf{x}_{target} = \mathbf{x}_{base} + s(\mathbf{x}_{base}) \quad (4)$$

which can be rearranged to:

$$s(\mathbf{x}_{base}) = \mathbf{x}_{target} - \mathbf{x}_{base} \quad (5)$$

where  $s(\mathbf{x})$  is an displacement vector with an RBF approximation in each direction separately, given by Equation ( 6 ) [21].

$$s(\mathbf{x}) = \begin{bmatrix} d_x(\mathbf{x}) \\ d_y(\mathbf{x}) \\ d_z(\mathbf{x}) \end{bmatrix} \quad (6)$$

where  $d_j(\mathbf{x}) = \sum_{i=1}^n w_i^j \varphi(\|\mathbf{x} - \mathbf{x}_i\|)$ ,  $j = x, y, z$

With the known baseline and target vectors of the skin and skeleton, the weights in  $s(\mathbf{x})$  are solved for, in each direction separately, according to Equation ( 8 ).

$$d_j(\mathbf{x}_{base}) = (\mathbf{x}_{target} - \mathbf{x}_{base})_j \quad (7)$$

$$\sum_{i=1}^n w_i^j \varphi(\|\mathbf{x}_{base} - \mathbf{x}_i\|) = (\mathbf{x}_{target} - \mathbf{x}_{base})_j \quad (8)$$

where  $\mathbf{x}_i$  is the i-th baseline vector. Setting up ( 8 ) for all  $n$  pairs of known  $\mathbf{x}_{base}$  and

$\mathbf{x}_{target}$ , yields  $n$  equations and  $n$  unknown  $w_i^j$  in each separate direction  $j = x, y, z$ .

### Regularization

If the RBF interpolation is done as described above, the resulting function will map all the data points that were used to generate the function, exactly. This can become a problem if the data is noisy, because then the noise will affect the resulting function. This problem can be resolved by regularization. With regularization, Equation ( 3 ) is replaced by the following minimization problem [22], to be solved for  $\mathbf{w}$ :

$$\min_{\mathbf{w}} \{ \|\mathbf{H}\mathbf{w} - \mathbf{y}\|^2 + \lambda \|\mathbf{w}\|^2 \} \quad (9)$$

where  $\lambda$  is a chosen constant. Equation ( 9 ) has a penalty term for  $\mathbf{w}$ , hence, the weight vector  $\mathbf{w}$  should fit the training data well but the length of the vector should be kept short. With regularization the resulting function will not map all the training data exactly. Regularization was not used in this project, as it increases computing time, and the data was not especially noisy.

## 2.7 Element quality

To control if the morphed HBM is suitable for a simulation, the element quality should be checked. The element quality of a FE model can be checked with the Jacobian [23]. The Jacobian indicates how much the shape of the element differs from an ideal shape, with straight edges [23]. The ideal shape would give a Jacobian value of 1.0 [23]. A negative value of the Jacobian indicates a collapsed element, where the edges cross over each other, which would cause problems in a simulation [23].



### 3 Method

Two HBMs, AF05 V4.02 Pedestrian (adult model) and 3YO V4 Occupant (child model), were downloaded from THUMS [24]. For each model, the following method was implemented, which included the extraction of the skin and skeleton from the HBM, rigging and positioning the skin and skeleton in Blender, and morphing the full HBM into the new position, using RBF interpolation. The RBF interpolations were performed on a computer equipped with an 11th Gen Intel Core i7-11700 2.50 GHz processor and 32 GB of random-access memory. Any variations in the method between the two models are specifically indicated.

#### 3.1 Pre-processing of THUMS-model in LS-PrePost

The model was imported into LS-PrePost V4.10.1 (ANSYS, Canonsburg, Pennsylvania, USA). The skin and the skeleton components were separated from the model and were saved as two separate stl files.

#### 3.2 Rigging of skeleton and skin in Blender

The addon “Rigify” was activated in Blender v 3.4.1 (Blender Foundation, Amsterdam, Netherlands) to access ready-made bone models and rigging functions. The stl files with the skeleton and skin were imported in Blender. The skeleton and skin were scaled down, together, to a size where it was possible to rotate the view without having the skin and skeleton disappearing from the scene. The skeleton was separated into its loose parts. Duplicates in the skeleton were deleted and the skeleton was rejoined.

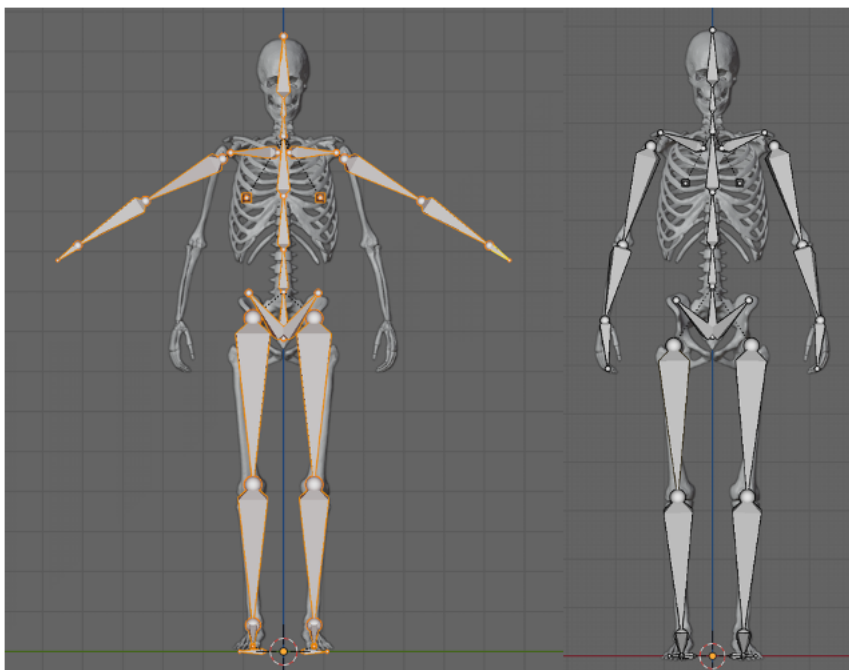


Figure 2. Frontal plane of the skeleton of the THUMS *AF05 V4.02 Pedestrian* model and the meta-rig, in Blender. Before (left) and after (right) positioning of the meta-rig in the skeleton.

Rigify's pre-made bone-model "Basic Human (Meta-Rig)" was added to the scene. The meta-

rig was positioned in the imported skeleton, see Figure 2. The models were then rigged in two different ways. The *3YO V4 Occupant* model was rigged through the rigify-function "Generate rig". After the rig was generated, all IK stretches were set to 0 by running a script (see Appendix 1) in Blender. The skin was attached by parenting the generated rig to the skin, with automatic weights. The *AF05 V4.02 Pedestrian* model was rigged by parenting the meta-rig to the skeleton, with automatic weight. The skin was attached by parenting the meta rig to the skin, with automatic weights.

### 3.3 Weight-painting in Blender

The skeleton was separated into its loose parts. The weight distribution of each individual part were checked. The parts that should not affect the selected part were removed. In Figure 3 the weight distribution for the left thigh is shown. The influence of the left hand was removed by clicking the cross followed by clicking Normalize and then Copy. The same process was done for all loose parts of the skeleton. The skeleton was then rejoined.

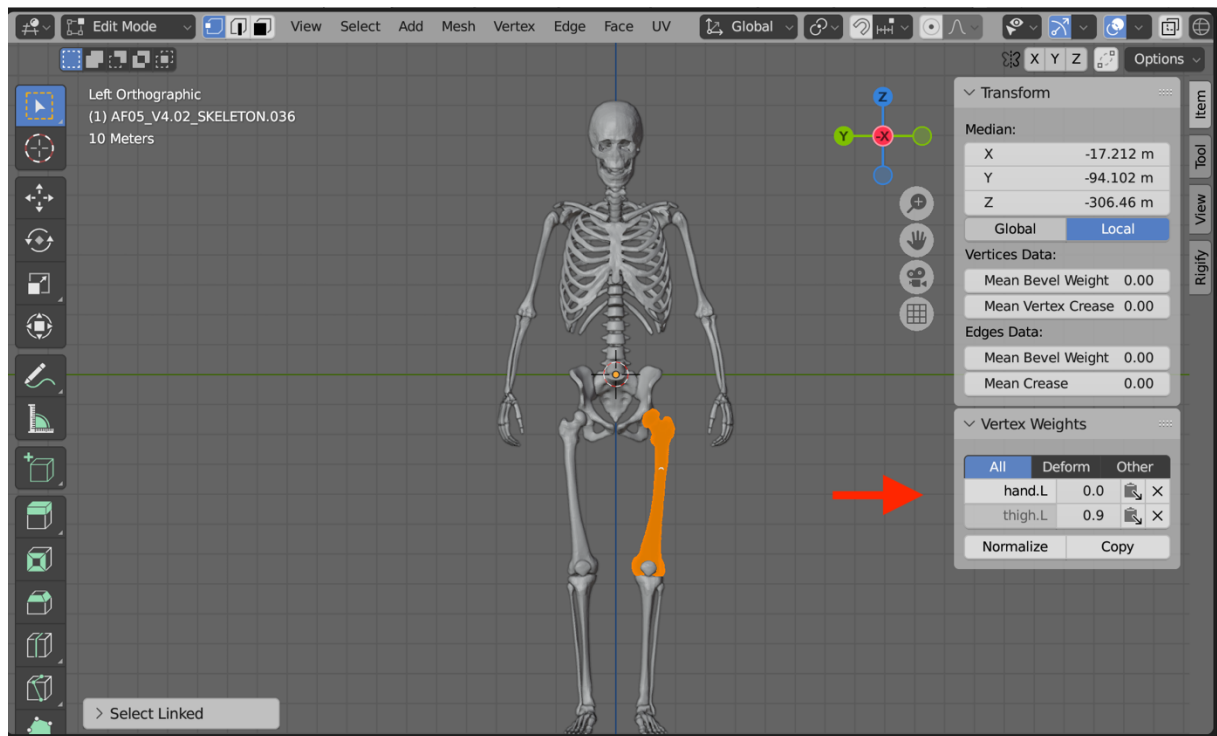


Figure 3. The skeleton of the THUMS *AF05 V4.02 Pedestrian* model in Blender. The red arrow points at the left thighs weight distribution, with the Normalize and Copy buttons below.

The following process was then followed for weight painting. The skin and skeleton were weight painted separately until they moved naturally by visual inspection. Skeleton parts that were challenging to weight paint correctly, when the whole skeleton was selected, were instead weight painted with the skeleton separated into its individual parts, as described above. See Figure 4 for an illustration of the difference between weight painting with and without the skeleton separated.

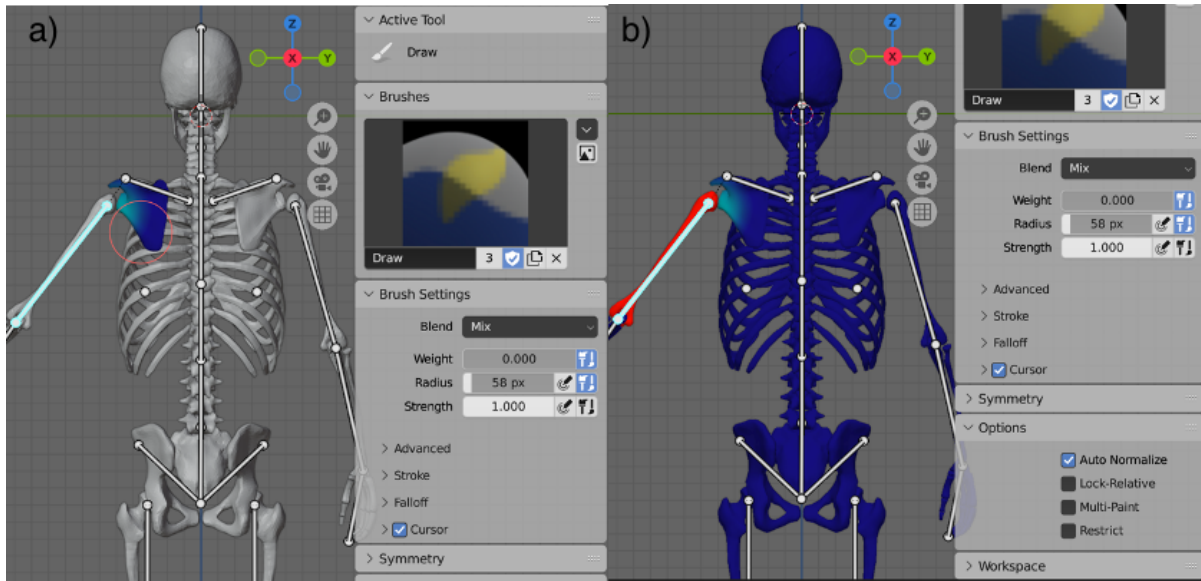


Figure 4. The skeleton of the THUMS *AF05 V4.02 Pedestrian* model being weight painted in Blender. a) Weight painting upper left arms movement effect on the left shoulder blade separately. b) Weight painting upper left arms movement effect on the whole skeleton.

### 3.4 Positioning tests of rigged *AF05 V4.02 Pedestrian* model

The movement limits for the arms and legs, with the skin and skeleton together, of the rigged *AF05 V4.02 Pedestrian* model were tested. The tests were done by rotating the models' hip, knee, elbow, and shoulder joints one at a time, through normal human motion paths, until a problem was found (bones protruding through the skin, unnatural folding et cetera).

### 3.5 Decimation and positioning of model before interpolation.

The skull, hands, and feet were removed from the skeleton of the *3YO V4 Occupant* model. The Decimate modifier in modifier properties was used to decimate the number of faces in the models to different ratios, see Table 2, with symmetry along the sagittal plane.

Table 2. Decimation ratios used for the different models.

Model	Decimation ratio
<i>AF05 V4.02 Pedestrian</i>	Skin: 0.125 Skeleton: 0.125
<i>3YO V4 Occupant</i> (Without the skeletons skull, hands and feet)	Skin: 0.2 Skeleton: 0.15

The model was scaled up to its original size and the baseline model (the not positioned skeleton and skin) was exported as a stl file. The model was then positioned in a desired position in Pose mode, using the rigged controls. *AF05 V4.02 Pedestrian* was positioned in 2 different positions, Position 1 and Position 2. Position 1 was positioned according to the values in Table 3. Position 2 was positioned according to the values in Table 4. *3YO V4 Occupant* was positioned, in position 3, according to the values in Table 5. The body parts not mentioned in Table 3-5, were not manipulated. All positions were exported as stl files as

target models (positioned skeleton and skin) to their corresponding base model.

Table 3. Coordinates of start and end positions for AF05 V4.02 Pedestrian from base to position 1. Coordinate are given in the “Axis angle”-coordinate system, in Blender.

<i>AF05 V4.02 Pedestrian</i> Position 1 Part	Start position (W, X, Y, Z)	End position (W, X, Y, Z)
Right forearm	(0°, 0, 1, 0)	(42.9°, 0.960, .192, -.204)
Left forearm	(0°, 0, 1, 0)	(30.9°, 0.960, -.192, .204)
Right thigh	(0°, 0, 1, 0)	(34.1°, -1, 0.001, 0.003)
Left thigh	(0°, 0, 1, 0)	(12.7°, 0.003, 0.003, -1)
Right shin	(0°, 0, 1, 0)	(33.2°, 0.999, 0.040, 0.000135)

Table 4. Coordinates of start and end positions for *AF05 V4.02 Pedestrian* from base to position 2. Coordinates are given in the “Axis angle”-coordinate system, in Blender.

<i>AF05 V4.02 Pedestrian</i> Position 2 Part	Start position (W, X, Y, Z)	End position (W, X, Y, Z)
Right upper arm	(0°, 0, 1, 0)	(40.5°, 0.167, .036, -.985)
Left upper arm	(0°, 0, 1, 0)	(40.5°, 0.167, -.036, .985)
Right thigh	(0°, 0, 1, 0)	(48.6°, -1, 0.001, 0.003)
Left thigh	(0°, 0, 1, 0)	(48.6°, -1, 0.001, -0.003)
Right shin	(0°, 0, 1, 0)	(35.9°, 0.999, 0.040, -0.000136)
Left shin	(0°, 0, 1, 0)	(35.9°, 0.999, -0.040, -0.000136)

Table 5. Coordinates of start and end positions for *3YO V4 Occupant* from base to positioned. Coordinate are given in the “Axis angle”-coordinate system, in Blender.

<i>3YO V4 Occupant</i> Position 3 Part	Start position (W, X, Y, Z)	End position (W, X, Y, Z)
Right upper arm	(0°, 0, 1, 0)	(80.3°, 0.022, -0.038, -0.941)
Right hand	(0°, 0, 1, 0)	60°, 0.476, 0.878, 0.055
Left thigh	(0°, 0, 1, 0)	(0°, -0.1, 0.35)
Left foot	(0°, 0, 1, 0)	(64.8°, -1, 0, 0)

### 3.6 Interpolation and element quality check

An RBF algorithm executable file was used for the interpolation (krbf.exe 2023 v1.9.4, provided by project supervisor Xiaogai Li). The baseline and target models were converted to separate k files and were used as inputs in the interpolation program together with their corresponding original HBM. Both models, in all positions, were interpolated with linear radial basis function, without regularization. After the interpolation the shell elements of the morphed models were quality checked in LS-PrePost, using the Jacobian with minimum allowable Jacobian value of 0.0, 0.1 and 0.7. The element qualities of the corresponding base models were also checked.

## 4 Results

In this chapter, the results from executing the described method are presented. The results include how many nodes and elements the models had, the results of the positioning tests of the *AF05 V4.02 Pedestrian* model, the results of morphing both models by RBF interpolation, and the results of the element quality checks of the morphed models. Furthermore, the *AF05 V4.02 Pedestrian* model will also be referred to as the adult model and the *3YO V4 Occupant* model will also be referred to as the child model.

### 4.1 Pre-processing of THUMS-model in LS-PrePost

The *AF05 V4.02 Pedestrian* model consists of 878461 nodes and 2514872 elements. The extracted skeleton consisted of 115039 nodes and 230122 elements. The extracted skin consisted of 114993 nodes and 240208 elements.

The *3YO V4 Occupant* model consists of 828608 nodes. The extracted skeleton consisted of 93480 nodes and 191487 elements. The extracted skin consisted of 81585 nodes and 163166 elements.

### 4.2 Positioning tests of the rigged adult model

The adult model was tested in multiple positions. Problems that arose are shown in Figure 5-7. Specific problems that are shown in the figures include, the forearm penetrated the upper arm, (Figure 5a), the shoulders caved in when rotating the shoulder joint upwards, (Figure 7b) and the skeleton protruded through the skin in multiple positions (Figure 5b, 6b and 7a). In the figures in this section, the rotation angle, W, is the rotation from the baseline position of the model.

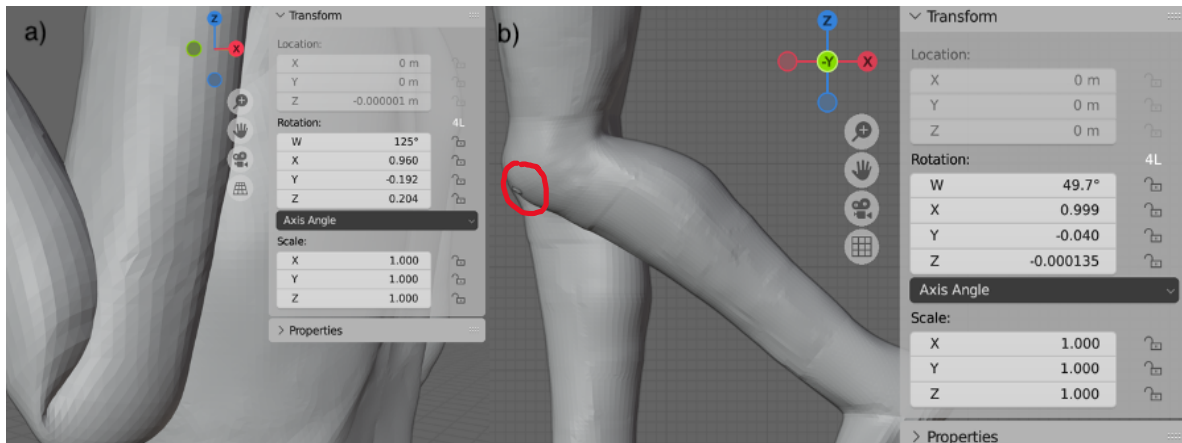


Figure 5. The skeleton and skin of the THUMS *AF05 V4.02 Pedestrian* model, rigged in Blender. W is the rotation angle from the baseline position of the joint. a) Left forearm rotated 125° from base position, forearm penetrates the upper arm. b) Left knee joint rotated 49.7° backwards from the base position, bone protrudes through the skin by the knee (seen inside the red circle).

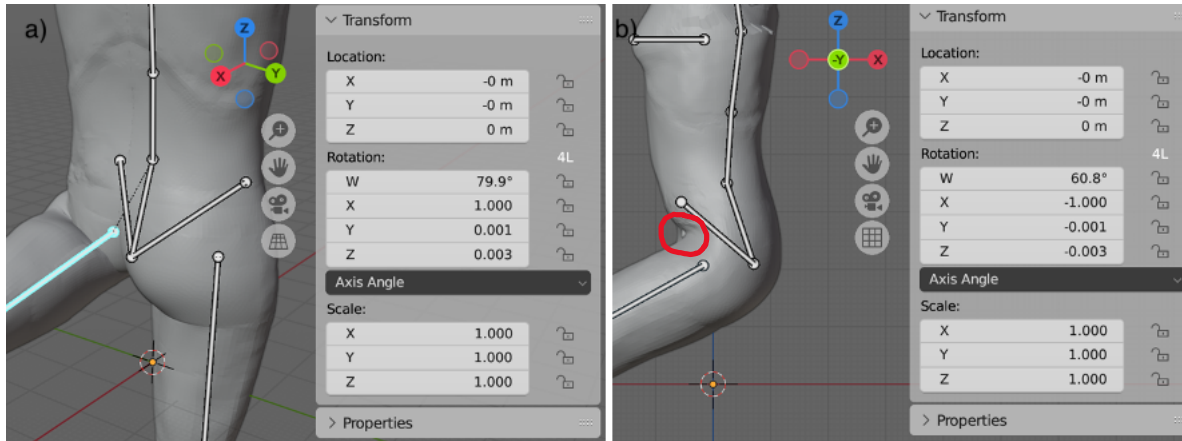


Figure 6. The skeleton and skin of the THUMS *AF05 V4.02 Pedestrian* model, rigged in Blender. W is the rotation angle from the baseline position of the joint. a) Hip joint rotated backwards  $79.9^\circ$  from the base position, backside of the hip area caves in. b) Hip joints rotated forward  $60.8^\circ$  from the base position, bone protrudes through the skin by the hip (seen inside the red circle).

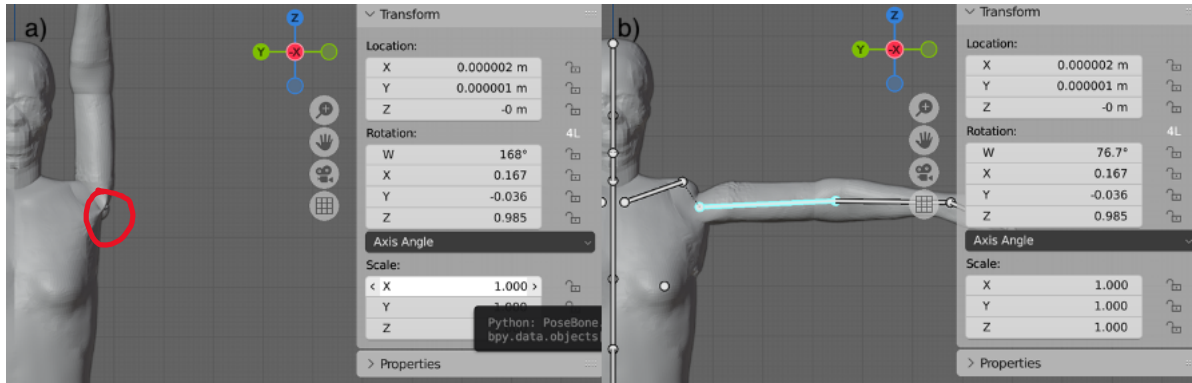


Figure 7. The skeleton and skin of the THUMS *AF05 V4.02 Pedestrian* model, rigged in Blender. W is the rotation angle from the baseline position of the joint. a) Shoulder joint rotated to the side  $168^\circ$  from the base position, bone protrudes through the skin by the armpit (seen inside the red circle). b) Shoulder joint rotated to the side  $76.7^\circ$  from the base position, shoulder caves in.

### 4.3 Morphed HBMs by RBF interpolation

After decimation in Blender, the *AF05 V4.02 Pedestrian* model had 24128 nodes and 52965 elements in Blender and, 23974 nodes and 52965 elements in LS-PrePost. After decimation in Blender, the *3YO V4 Occupant* model had 25157 nodes and 52792 elements in Blender and the same in LS-PrePost. The interpolation results from running the RBF algorithm executable file are presented below.

#### Adult model interpolations

The baseline (red) and target (blue) positions of the adult model in position 1 are shown in Figure 8. The baseline and target positions are overlapping each other to indicate where the difference between the positions lies. The base model is shown in Figure 9a and the model morphed to position 1 are shown in Figure 9b-d, from different angles. Close ups of the rib area, that are shown in Figure 10a-b, indicates no visual problems. The right forearm of the morphed model, shown in Figure 10c, compared to the right forearm of the base model, shown in Figure 10d, indicates a possible problem with the area of the forearm, close to the elbow joint.

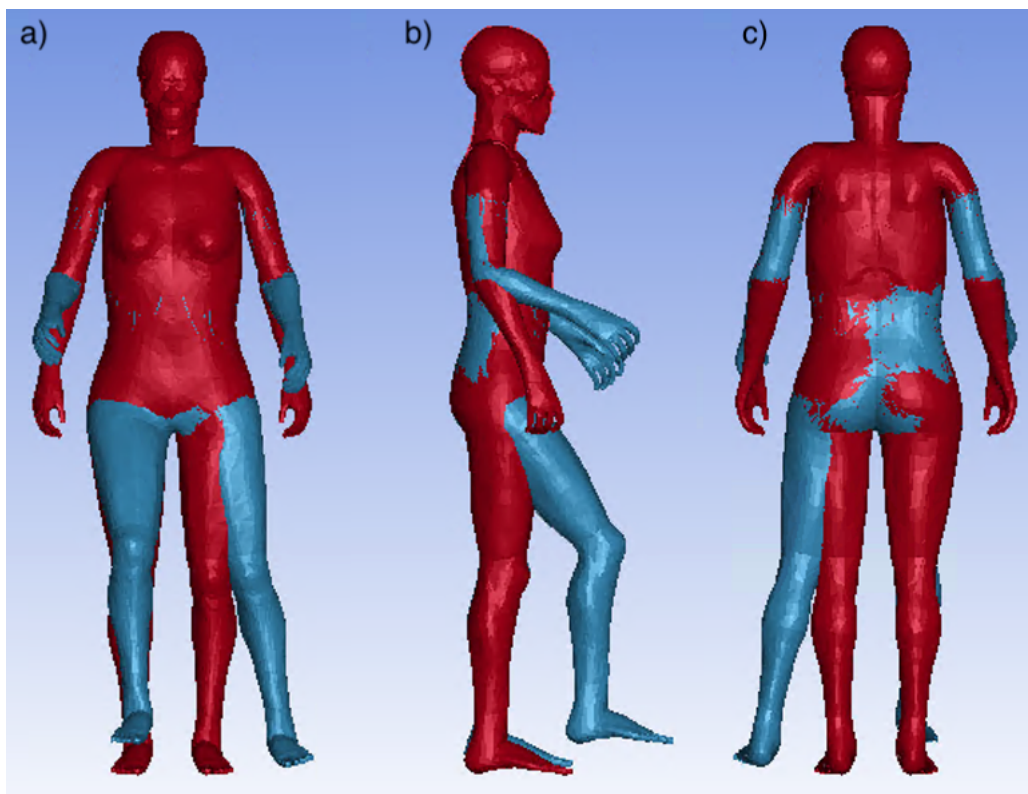


Figure 8. The skin and skeleton of the THUMS model *AF05 V4.02 Pedestrian* in position 1 (blue), overlapping with the baseline position (red), shown in LS-PrePost. a) Front b) Right side c) Back

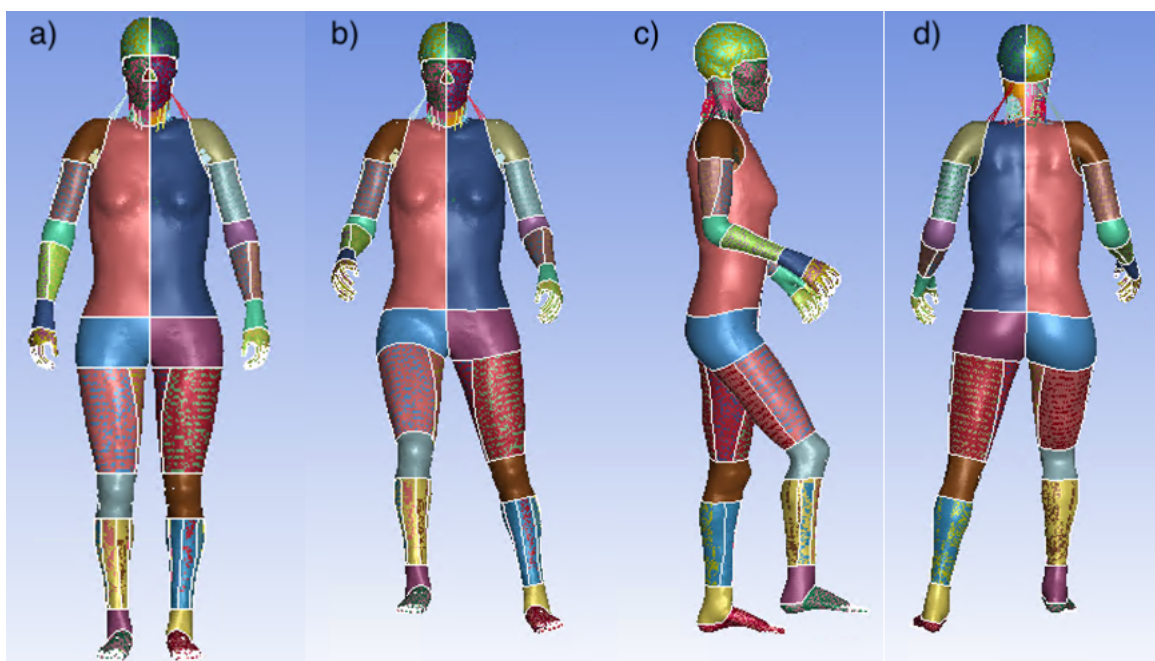


Figure 9. The THUMS *AF05 V4.02 Pedestrian* model. a) Base model. b) Front of model, morphed to position 1. c) Right side of the morphed model in position 1. d) Back of the morphed model in position 1.

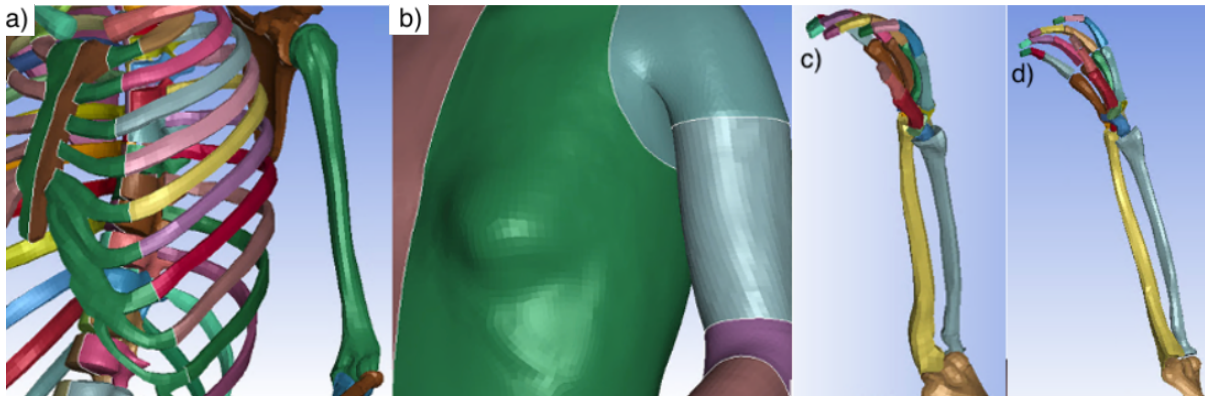


Figure 10. Close up of rib area of the THUMS *AF05 V4.02 Pedestrian* model after morphing to position 1. a) Skeleton of morphed model. b) Skin of morphed model. c) Right forearm bones of morphed model. d) Right forearm bones of original base model.

The baseline (red) and target (blue) positions of the adult model in position 2 are shown in Figure 11. The base model is shown in Figure 9a. The model morphed to position 2 are shown, from different angles, in Figure 12a-c. Problem areas can be seen in the rib areas and on the upper arms, where the skin is forming unnatural wave patterns. Close ups of the rib area are shown in Figure 13, both in Blender (b and d) and in LS-PrePost (a and c). In Figure 13a, a wave-like form of the ribs of the morphed model can be seen. In Figure 13c, the skin in the rib area of the morphed model does not look natural and the skin under the arm is hanging down, this can be compared to the smooth surface in the same area of the model morphed to position 1, displayed in Figure 10b. As seen in Figure 13b and d, the same problems did not occur in Blender before the interpolation.

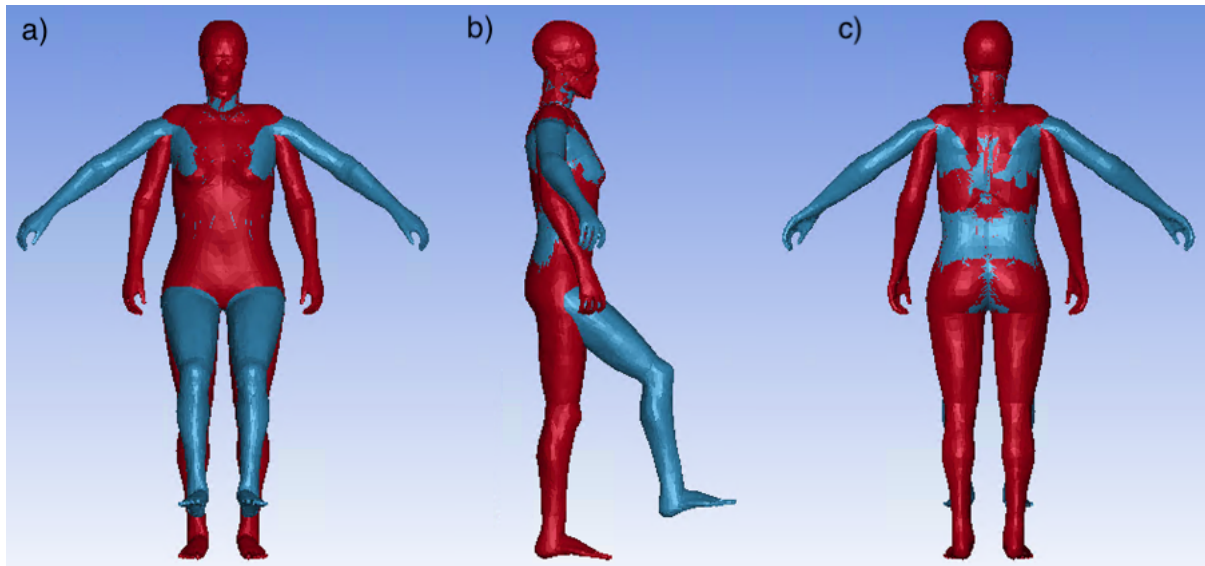


Figure 11. The skin and skeleton of the THUMS *AF05 V4.02 Pedestrian* model in position 2, shown in LS-PrePost. Baseline position (red) and target position (blue). a) Front b) Right side c) Back

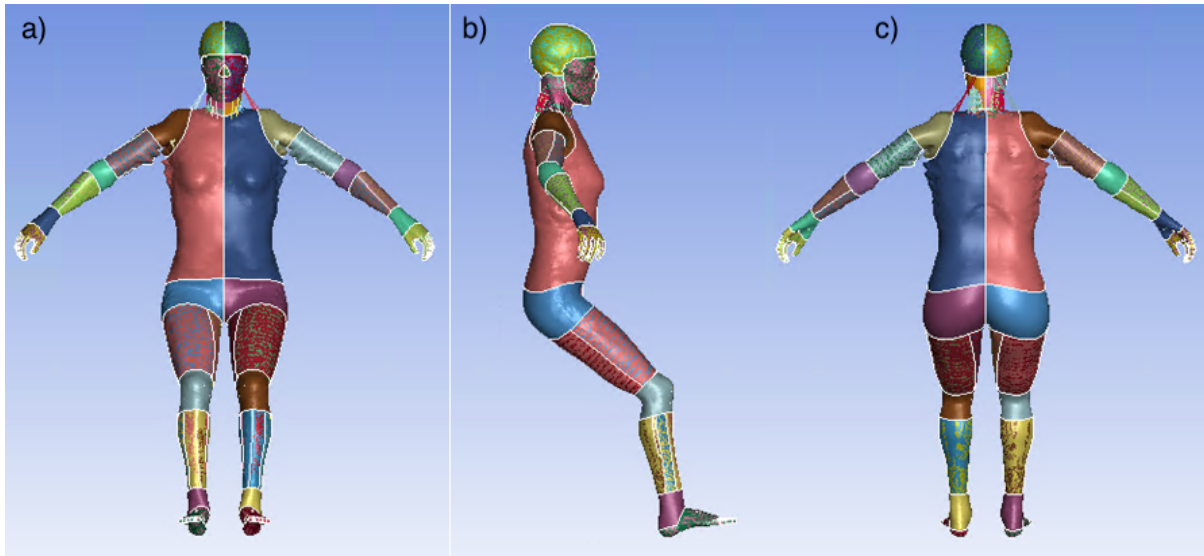


Figure 12. The complete THUMS AF05 V4.02 Pedestrian model, morphed to position 2. a) Front b) Right side c) Back

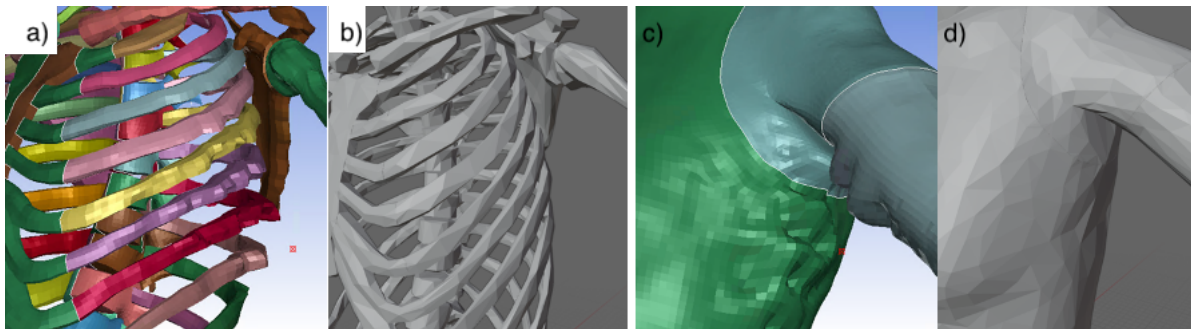


Figure 13. A problem area after the morphing of the THUMS AF05 V4.02 Pedestrian model to position 2. a) Rib area of the skeleton of the morphed model. The ribs have an unnatural wave-like form. b) The skeleton of the model in Blender (before morphing). c) Skin of morphed model. The skin is not smooth by the ribs and the skin is hanging from the upper arm. d) Skin of the model in Blender (before morphing).

### Child model interpolations

Figure 14a shows the child base model. The target position (blue) overlapping with the baseline position (red) are presented in Figure 14b. The child model morphed to the new position is shown in Figure 14c. Figure 14 highlights a problem area of the morphed child model. Figure 14a and 15b shows the morphed model in LS-PrePost. Figure 14a highlights the deformation of the skeleton and Figure 14b shows the deformation of the skin. Figure 14c and d show the corresponding areas before the interpolation, observed in Blender. The issues with this area include distortion in the thorax region, a wave-like deformation of the bones of the ribcage and that the skin is pulled out from the rib area. Similar deformations are occurring on the upper arm, that is a wave-like deformation and hanging skin.

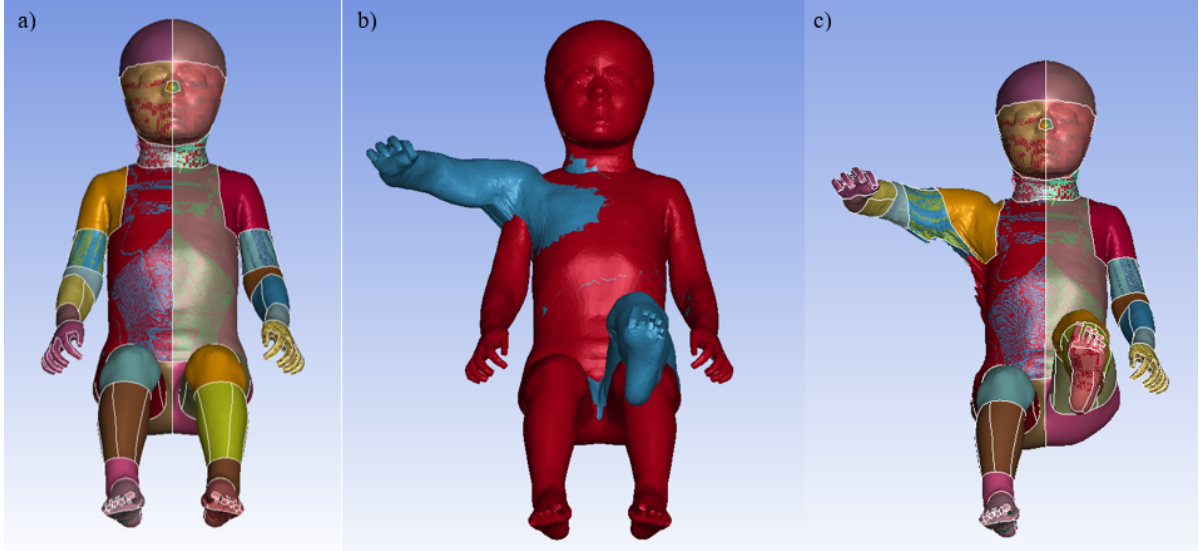


Figure 14. The THUMS 3YO V4 Occupant model. a) Complete base model b) Baseline position (red) and target position (blue) of skeleton and skin, positioned and decimated in Blender with decimation ratio 0.15 for the skeleton and 0.2 for the skin. c) Morphed model, morphed with linear radial function and no regularization.

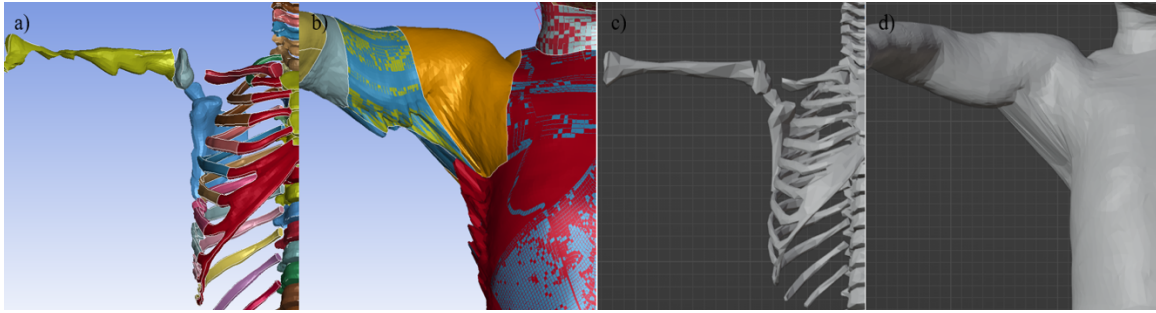


Figure 15. Problem areas after the interpolation (picture taken of model morphed with linear radial function, with no regularization) of the THUMS 3YO V4 Occupant model with decimation ratio 0.15 for the skeleton and 0.2 for the skin. a) The skeleton of morphed model. The arm bone has an unnatural shape, the ribs and scapula have wave-like forms. b) The skin of the morphed model. The skin is hanging from the upper arm and the skin is pulled out from the rib area. c) The skeleton of the model in Blender (before interpolation). d) The skin of the model in Blender (before interpolation). Problem with stretched skin by the armpit.

#### 4.4 Element quality check

The element qualities are presented in Table 6, below. In the table it can be seen that significantly more elements in the morphed models fail the tests compared to their corresponding base model. Additionally, all the morphed models contain elements with negative Jacobian values. It can also be seen that position 1 is the position that differs the least from its corresponding base model, when looking at violated elements with minimum Jacobian values of 0.1 and 0.7.

Table 6. Results of quality checks of the shell elements of the morphed models and the original base models, for comparison. The element quality was checked in LS-PrePost using the Jacobian with minimum allowable Jacobian value of 0.0, 0.1 and 0.7, indicated in parenthesis in the first row. Violated (%) are the percentage of the elements that had a Jacobian value smaller than the minimum allowable Jacobian value. Min. val and Max. val are the minimum and maximum Jacobian value for the models.

Model	Violated (%) (0.0)	Violated (%) (0.1)	Violated (%) (0.7)	Min. val	Max. val
<i>AF05 V4.02</i> <i>Pedestrian</i> <b>Base model</b>	0 (0%)	2 (0.000358%)	10,750 (1.92%)	0.099	1
<i>AF05 V4.02</i> <i>Pedestrian</i> <b>Position 1</b>	7 (0.00125%)	22 (0.00394%)	12,031 (2.15%)	-0.0954	1
<i>AF05 V4.02</i> <i>Pedestrian</i> <b>Position 2</b>	27 (0.00483%)	100 (0.0179%)	16,463 (2.95%)	-0.256	1
<i>3YO V4</i> <i>Occupant</i> <b>Base model</b>	0 (0%)	0 (0%)	12,479 (2.09%)	0.178	1
<i>3YO V4</i> <i>Occupant</i> <b>Position 3</b>	3 (0.000502%)	43 (0.0072%)	18,178 (3.04%)	-0.0189	1



## 5 Discussion

In this chapter the method and the following results of this project are discussed. The discussion includes rigging, weight painting, decimation, and positioning of the models in Blender, the results of the morphing of the HBMs to the new position by an RBF interpolation and some recommendations for future development.

### 5.1 Rigging, weight painting and positioning tests in Blender

The skin and skeleton were extracted from the THUMS HBMs. The number of nodes and elements in the extracted skin and skeleton, for both models, are presented in section 4.1. The skin and skeleton were extracted manually in LS-PrePost. As the HBMs consist of multiple layers, different parts could have been chosen for the skin and skeleton. Therefore, because the extraction was done manually, the extraction is a source of errors. Another extraction may result in a different number of nodes and different elements in the extracted skin and skeleton.

The skin and skeleton models were imported to Blender where they were rigged, weight painted and then positioned. For the rigging, two different methods were used. Both methods used a meta-rig as a base. The child model was rigged with the rigify function “Generate rig” and the adult model was rigged by directly parenting the meta-rig to the skeleton and skin. The rigify function generates more movement controls compared to the parenting method. However, all the movement controls were not needed for the positioning done in this project.

After the models were rigged, they were weight painted. We did not manage to find a way to automate the process of weight painting, hence the weight painting was done manually. The manual process led to the weight painting being a source of errors. Errors both in getting the weight painting perfect and in yielding diverse results for different users. Furthermore, the weight painting process differed slightly between the two different rigging methods, with the weight painting being more time consuming for the model rigged with the rigify function compared to the parenting method.

After the models were rigged and weight painted, positioning tests were done, and documented for the adult model in Figure 5-7. As shown in the figures the rigged adult model exhibited unnatural movements. For example, the forearm penetrated the upper arm beyond a certain rotation angle, the shoulders caved in when rotating the shoulder joint upwards and the bones protruded through the skin in multiple positions. To resolve the issue of caved in shoulders and bones protruding through the skin, sculpting of the positioned model was tried, where sculpting means to manually adjust nodes with the help of built in tools in Blender. However, sculpting was time consuming, challenging to get right and had to be repeated for each new position. To avoid the need to repeat the sculpting for each new position, a shape key can be used. A shape key works by applying a function, to a model that has been sculpted, that changes the sculpting when the body part is moved. However, shape keys were more challenging to get right than just sculpting alone and it got complex when multiple areas had to be sculpted which led to multiple shape keys that affected the same nodes. For these reasons, neither sculpting nor shape keys were used in the final method.

## 5.2 Decimation and positioning in Blender

After the skin and skeleton models were rigged in Blender, the models were to be positioned and then used in an RBF interpolation to morph the complete HBM into the new position. However, as reported in section 4.1, the THUMS models consist of many nodes and the RBF interpolation has a time complexity proportional to the number of nodes cubed. Therefore, to be able to run the interpolation on regular personal computers, the number of nodes in the models had to be decimated. The decimations were done in Blender, before the positioning of the models, to make sure that the same nodes were removed from the baseline and target positions. As for the decimation ratios used, the reason for using a decimation ratio of 0.125 for the adult model was because that for all higher decimation ratios, that was tried, the baseline and target models had a difference in the number of nodes when the models were imported to LS-PrePost (the baseline and target models have to have the same number of nodes to be able to execute the interpolation). The node difference did not show in Blender and the reason for this problem is not clear. For the child model a few different decimation ratios were tested, and the ratio chosen (see Table 2) gave the best results (the least number of nodes while keeping the structure of the model). The head, hands, and feet of the skeleton were removed from the child model before the decimation to decrease the overall number of nodes in the mesh, and therefore made it possible to use higher decimation ratios for the remaining parts. The child model did not have the problem with a different number of nodes between the target and baseline models.

After the decimation, the models were scaled up to their original size and then positioned, in Blender. The adult model was placed in two different positions, showcased in Figure 8 (position 1) and Figure 11 (position 2). The difference between the two positions is that position 2 is moved more from the base position than position 1. The main difference is that the upper arms are moved in position 2, but not moved at all in position 1. This difference is showcased because that in all trials before the final method, problems arose around the rib and upper arm areas after the morphing was done. Similar problems were found for the child model, but only one position was presented, see Figure 14.

## 5.3 Morphing HBMs by RBF interpolation

After positioning, the skeleton and skin models were used in an RBF interpolation to morph the complete HBMs into the new positions. The provided RBF interpolation program had four different radial basis functions that could be chosen (the same radial basis functions as presented in Table 1, section 2.6). The choice of whether to perform the interpolation with regularization or not, was also available. In all of the interpolations presented in this project, the linear radial basis function was used, as the other functions did not improve the end results, and regularization was not used, as it increased the computing time a lot.

The results of the morphing of the adult model to position 1, presented in Figure 8-10, yielded much better visual results than the morphing to position 2, presented in Figure 11-13. The rib area of the morphed model in position 1 (see Figure 10a and b) has no visual distortions, neither does the complete morphed model seen in Figure 9. Both the morphed and the original models forearm bones look twisted (seen in Figure 10c and d, respectively), but the morphed models forearm bones look more twisted. The difference between the models is small but it probably affects the element quality.

The adult model morphed to position 2 yielded some problems, especially in the rib area

which are showcased in Figure 13a and c. However, the problems were not visible in the corresponding areas of the model viewed in Blender, before the morphing (see Figure 13b and d).

The morphed child model, shown in Figure 14 and 15, has deformations in the rib and the upper arm areas, on both the skeleton and skin. These deformations are more prominent in the child model, but they look very similar to the deformations of the adult model in position 2. Also, like the adult model in position 2, the same problems did not show in Blender before the interpolation of the child model. However, the child model has an obvious problem with the skin in the armpit area in Blender before the interpolation, as seen in Figure 15d.

As seen in Table 6, all the morphed models contain elements with negative Jacobian values, which would cause problems if a simulation was to be run with these models. Even though the child model got the fewest elements with negative Jacobian values, the adult model in position 1 has the best overall element quality among the three morphed models. The main difference between position 1 and the other positions is that in position 1, the upper arms were not moved from the baseline position.

The fact that the problems of the adult model in position 2, see Figure 13, and of the child model, see Figure 15, did not show in Blender, indicates that the problem arose during the interpolation and hence, better weight painting would not solve these problems. Better weight painting could, however, remove the problem in the armpit area of the child model, seen in Blender before the interpolation, see Figure 15d. A possible solution to the problems is to run the interpolations without decimating the number of nodes in the models beforehand. This would probably remove the wave patterns, see Figure 13 and 15, in the morphed models, as these wave patterns most likely occur due to the nodes in the affected areas being too close to each other, in the baseline position. The closeness and the fact that all nodes were not used to generate the displacement function, via the RBF interpolation, causes some nodes to be mapped in between the two body parts. Making it look like some upper arm nodes are drawn towards the ribs and vice versa. Regularization would not solve this problem because the nodes are not resembling noisy data, the nodes from the different body parts are not overlapping, they are separated but close to each other. However, to run the interpolation without first decimating the number of nodes requires a lot of computing power and would not resolve the problem that the skin is not moving relative to the skeleton in a natural manner, when positioning the models in Blender. For example, when the upper arm is moved, the upper arm's bone gets closer to the skin on one side of the arm and further away from the other side, compared to the base position. This is shown to be happening, to some extent, in the positioning tests, see Figure 5-7, where the skeleton eventually protrudes through the skin. This phenomenon limits how much movement that can be done and probably causes the morphed HBMs to get stretched elements on one side, and compressed elements on the other side of the bones, which yield worse element quality.

## 5.4 Final thoughts and future development

The end results of this project were not satisfactory. As discussed above, the HBMs positioned according to the described method had visual distortions and their element qualities were not satisfactory. However, the results indicates that it is possible for this approach to work, as smaller movements (position 1) did yield better results. Why larger movements yielded worse results is probably because the skin and skeleton moved relative to each other. To solve the problem of relative movement between the skin and skeleton, a

better way of attaching the skin to the skeleton should be found. The solution should enforce a constant distance between the skeleton and skin, while still allowing the skin to stretch and deform naturally. Also, with a more powerful computer, it should be tested to do the RBF interpolation without decimating the number of nodes beforehand, to see if that improves the results. However, as discussed earlier, when the adult model was imported to LS-PrePost with a decimation ratio higher than 0.125, the baseline and target models had a different number of nodes, hence, interpolation without decimation would not work for the adult model, unless a solution to this problem is found. Furthermore, further investigation should be made on the element quality to see which individual elements that has bad quality.

Another recommendation for future development is to, instead of using the premade meta-rig, construct a custom rig from individual bones, similar to the meta-rig, and introduce additional bones for the knee and elbow joints. And then use the method of parenting the rig to the skeleton. This would allow for more control over the folding movements of the arms and legs.

## **6 Conclusion**

This project failed in developing a new, fully functional, method to position human body models via the 3D graphics program Blender combined with an RBF interpolation.



## 7 References

- [1] Centers for Disease Control and Prevention, ‘Road Traffic Injuries and Deaths—A Global Problem CDC’, *Road Traffic Injuries and Deaths—A Global Problem*.  
<https://www.cdc.gov/injury/features/global-road-safety/index.html#:~:text=Each%20year%2C%201.35%20million%20people,on%20roads%20around%20the%20world.&text=Every%20day%2C%20almost%203%2C700%20people,pedestrians%2C%20motorcyclists%2C%20or%20cyclists>. (accessed May 13, 2023).
- [2] C. Giordano, X. Li, and S. Kleiven, ‘Performances of the PIPER scalable child human body model in accident reconstruction’, *PloS One*, vol. 12, no. 11, pp. e0187916–e0187916, 2017, doi: 10.1371/journal.pone.0187916.
- [3] B. Pipkorn *et al.*, ‘Validation of the SAFER Human Body Model Kinematics in Far-Side Impacts’, in *Conference proceedings International Research Council on the Biomechanics of Injury, IRCOBI*, 2021, pp. 444–476. Accessed: May 20, 2023. [Online]. Available: <https://research.chalmers.se/en/publication/529443>.
- [4] E. Hwang *et al.*, ‘Development, Evaluation, and Sensitivity Analysis of Parametric Finite Element Whole-Body Human Models in Side Impacts’, *Stapp Car Crash J.*, vol. 60, pp. 473–508, Nov. 2016.
- [5] C. Klug *et al.*, ‘EURO NCAP VIRTUAL TESTING - CRASHWORTHINESS’.
- [6] G. Bacquaert, C. Bach, D. Draper, S. Peldschus, and F. Duddeck, ‘Positioning human body models for crashworthiness using model order reduction’, *Comput. Methods Biomech. Biomed. Engin.*, vol. 23, no. 11, pp. 734–743, Aug. 2020, doi: 10.1080/10255842.2020.1763321.
- [7] J. John, C. Klug, M. Kranjec, E. Svenning, and J. Iraeus, ‘Hello, world! VIVA+: A human body model lineup to evaluate sex-differences in crash protection’, *Front. Bioeng. Biotechnol.*, vol. 10, 2022, Accessed: Apr. 11, 2023. [Online]. Available: <https://www.frontiersin.org/articles/10.3389/fbioe.2022.918904>
- [8] M. Iwamoto, Y. Nakahira, and H. Kimpara, ‘Development and Validation of the Total Human Model for Safety (THUMS) Toward Further Understanding of Occupant Injury Mechanisms in Precrash and During Crash’, *Traffic Inj. Prev.*, vol. 16, no. sup1, pp. S36–S48, Jun. 2015, doi: 10.1080/15389588.2015.1015000.
- [9] N. Lindgren, Q. Yuan, B. Pipkorn, S. Kleiven, and X. Li, ‘Development and Whole-Body Validation of Personalizable Female and Male Pedestrian SAFER Human Body Models’. arXiv, May 11, 2023. doi: 10.48550/arXiv.2305.13918.
- [10] T. Janák, Y. Lafon, P. Petit, and P. Beillas, ‘A Method to Use Kriging With Large Sets of Control Points to Morph Finite Element Models of the Human Body’, *J. Biomech. Eng.*, vol. 143, no. 2, Nov. 2020, doi: 10.1115/1.4048575.
- [11] T. Xu, X. Sheng, T. Zhang, H. Liu, X. Liang, and A. Ding, ‘Development and Validation of Dummies and Human Models Used in Crash Test’, *Appl. Bionics Biomech.*, vol. 2018, pp. 1–12, Nov. 2018, doi: 10.1155/2018/3832850.
- [12] ‘What is a 3D Mesh Model? (Definition & Examples)’.  
<https://www.danthree.studio/en/blog-cgi/what-is-a-3d-mesh-model-definition-examples> (accessed May 20, 2023).
- [13] L. Flavell, *Beginning Blender: Open Source 3D Modeling, Animation, and Game Design*. Apress, 2011.
- [14] ‘Finite element method | Elsevier Enhanced Reader’.  
<https://reader.elsevier.com/reader/sd/pii/B9780125250962500076?token=DDA9F59AC EB81A7E85A762C23F80FBCD6AE553F99F02218BC7F8B30AC33F19A9A3905626 CFFFDEA69ECA8143F52A3D83&originRegion=eu-west->

- 1&originCreation=20230520143626 (accessed May 20, 2023).
- [15] A. Leledakis, J. Östh, J. Davidsson, and L. Jakobsson, ‘The influence of car passengers’ sitting postures in intersection crashes’, *Accid. Anal. Prev.*, vol. 157, p. 106170, Jul. 2021, doi: 10.1016/j.aap.2021.106170.
  - [16] P. Beillas *et al.*, ‘D3.8 Final version of the personalization and positioning software tool with documentation. PIPER EU Project’, 2017. Accessed: May 20, 2023. [Online]. Available: <https://www.semanticscholar.org/paper/D3.8-Final-version-of-the-personalization-and-tool-Beillas-Lafon/4f38f3c8be013e6b132b24326dc1f6b752d00a77?p2df>
  - [17] X. Li *et al.*, ‘Personalization of human body models and beyond via image registration’, *Front. Bioeng. Biotechnol.*, vol. 11, 2023, Accessed: May 25, 2023. [Online]. Available: <https://www.frontiersin.org/articles/10.3389/fbioe.2023.1169365>
  - [18] T. M. CORPORATION, ‘Toyota Offers Free Access to THUMS Virtual Human Body Model Software | Corporate | Global Newsroom’, *Toyota Motor Corporation Official Global Website*. <https://global.toyota/en/newsroom/corporate/32665896.html> (accessed Feb. 23, 2023).
  - [19] Q. Yan and P. Ho, ‘Introduction to LS-PrePost’, 2018.
  - [20] ‘Blender 3.4 Reference Manual — Blender Manual’. <https://docs.blender.org/manual/en/3.4/> (accessed May 22, 2023).
  - [21] T. Henneron, A. Pierquin, and S. Clénet, ‘Mesh Deformation Based on Radial Basis Function Interpolation Applied to Low-Frequency Electromagnetic Problem’, *IEEE Trans. Magn.*, vol. 55, no. 6, pp. 1–4, Jun. 2019, doi: 10.1109/TMAG.2019.2896623.
  - [22] K. Anjyo, J. P. Lewis, and F. Pighin, ‘Scattered data interpolation for computer graphics’, in *ACM SIGGRAPH 2014 Courses*, Vancouver Canada: ACM, Jul. 2014, pp. 1–69. doi: 10.1145/2614028.2615425.
  - [23] ‘Mesh Quality Checks - 2021 - SOLIDWORKS Help’. [https://help.solidworks.com/2021/english/SolidWorks/cworks/c\\_Mesh\\_Quality\\_Checks.htm?id=05b3b186be0744548d9cd59df543553d#Pg0](https://help.solidworks.com/2021/english/SolidWorks/cworks/c_Mesh_Quality_Checks.htm?id=05b3b186be0744548d9cd59df543553d#Pg0) (accessed May 24, 2023).
  - [24] ‘THUMS | Toyota Motor Corporation’. <https://www.toyota.co.jp/thums/> (accessed Apr. 20, 2023).

## Appendix 1: setIKstrech.py

```
import bpy

# Get a reference to the active armature object in the scene
armature = bpy.context.view_layer.objects.active

# Check if the active object is an armature
if armature.type == 'ARMATURE':

    # Switch to pose mode
    bpy.ops.object.mode_set(mode='POSE')

    # Iterate over each bone in the armature
    for bone in armature.pose.bones:

        # Set the IK stretch to 0
        bone["IK_Stretch"] = 0.0
```





