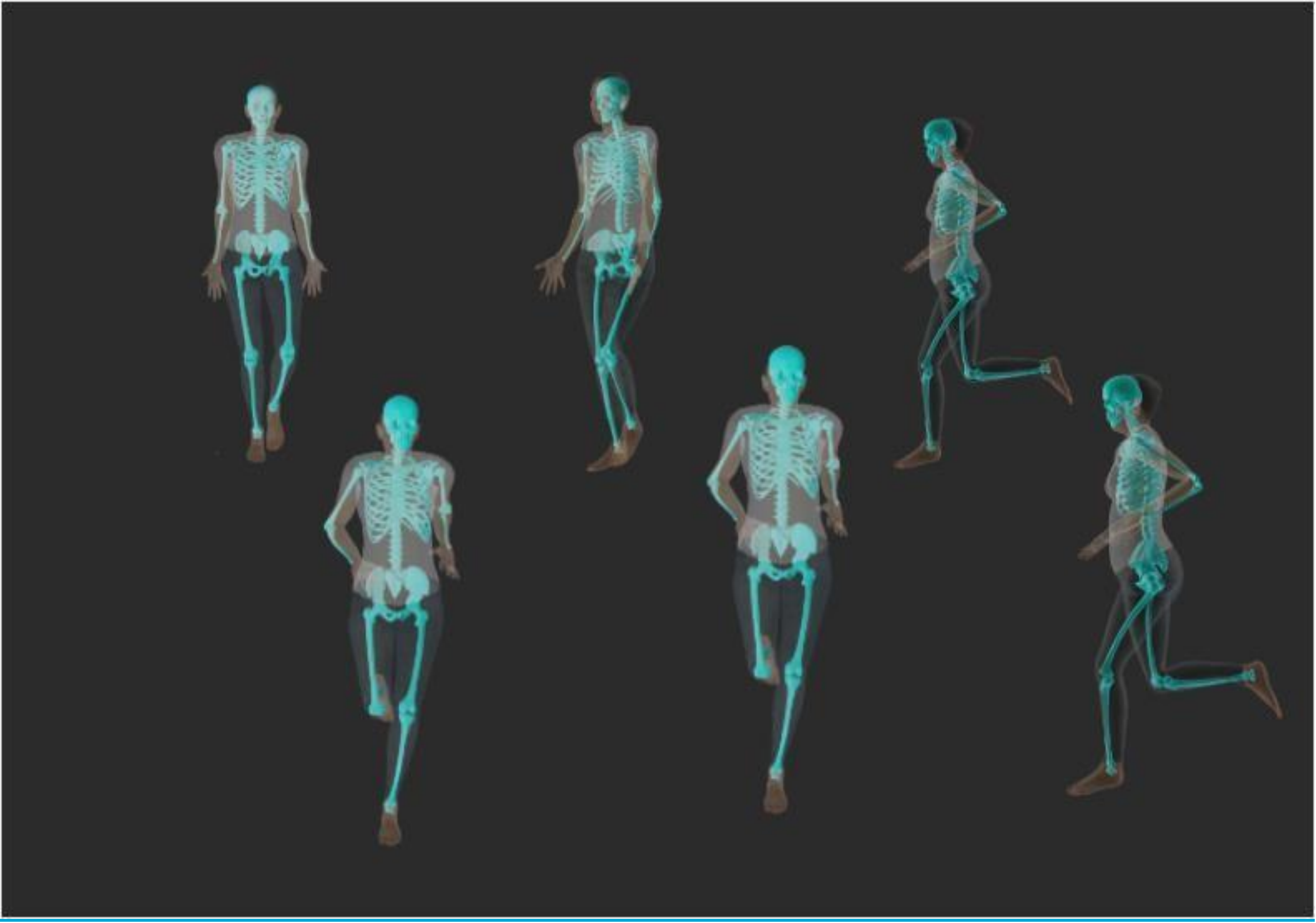
MSc Mechanical Engineering

SKINNING OF A MUSCULOSKELETAL MODEL AND A FEASIBILITY STUDY TO APPLY PIXEL LOSS REFINEMENT TO OPTIMIZE JOINT ANGLES

Akshath Ram Veeravalli Hari

TUDelft



Skinning Of A Musculoskeletal Model And A Feasibility Study To Apply Pixel Loss Refinement To Optimize Joint Angles

by

Akshath Ram Veeravalli Hari

to obtain the degree of Master of Science
at the Delft University of Technology.

Student number:	5277175	
Thesis committee:	Prof. dr. ir. E. van der Kruk,	TU Delft, supervisor
	Dr. A. Seth,	TU Delft, supervisor

Skinning Of A Musculoskeletal Model And A Feasibility Study To Apply Pixel Loss Refinement To Optimize Joint Angles

by

Akshath Ram Veeravalli Hari

to obtain the degree of Master of Science
at the Delft University of Technology.

Student number:	5277175	
Thesis committee:	Dr. E. van der Kruk,	TU Delft, supervisor
	Dr. A. Seth,	TU Delft, supervisor

SKINNING OF A MUSCULOSKELETAL MODEL AND A FEASIBILITY STUDY TO APPLY PIXEL LOSS REFINEMENT TO OPTIMIZE JOINT ANGLES

Akshath Ram Veeravalli Hari

Abstract - In biomechanics, human movement studies are carried out to assess the subject's kinematic and kinetic variables for a healthy gait. Currently, marker-based systems are the standardized method to extract the kinematic variables of subjects. The marker-based systems pose some serious challenges like cost and portability, and the calibration and synchronization of multiple cameras and sensors are among the other practical challenges. The AI technique often referred as markerless pose estimation methods can overcome these challenges and aid biomechanists and clinicians. Thus, there is a need to develop new deep-learning models that can regress the musculoskeletal model directly from images and videos. However, the deep-learning models are dependent on the quality and quantity of training data. In the current scenario, training data for markerless pose estimation are dependent on the redundant marker-based systems and the challenges persist. To aid this, it is necessary to create a statistical human model or a skinned human animated motion from a biomechanical model to build more training data. From the skinned virtual data consisting of realistic movements, deep-learning models can be trained. Therefore, the aim of the research was to build a pipeline to develop a human-animated model from a musculoskeletal model i.e., the OpenSim model. Two different motions such as walking and running are illustrated as qualitative results. The gait pattern for walking and running motions are realistic from both the frontal and sagittal planes. Furthermore, the deep learning model (D3KE) built by Marian et. al was also evaluated on the animated human motions eg. walking motion from the above pipeline to validate the model. The performance of D3KE is evaluated from different planes of camera views and also a comparison between the upper and lower extremities. The evaluation and comparison are based on two metrics MAE_{angles} (Mean Absolute Error of angles, in radians) and MPBLPE (Mean Per Bony Landmark Position Error, in cm). The MAE_{angles} and MPBLPE are better when observed from the frontal plane rather than from the sagittal plane as the plane of view. Also, the joint angles in the upper extremity show better results compared to the lower extremity. Although, the predictions of the joint angles are way off from the ground truth. This opens the way to perform a feasibility study to optimize joint angles by a pixel loss refinement technique. The findings and remarks on the pixel-loss refinement is tabulated as results.

I. INTRODUCTION

Biomechanics is a field of research that has evolved over the years to analyze human movement in the fields of medicine, sports, virtual reality, and product development such as shoes and prostheses. In sports, the kinematic and kinetic of an athlete, such as joint angles and their derivatives, torque in the joints, can be evaluated to increase performance by improving technique, as well as to prevent injury (Taborri et al., 2020). In a medical and clinical setting, the biomechanical model aids physicians in assessing the gait pattern variations after a stroke or physical abnormalities (Nadeau et al., 2013). Dynamic simulations of movement make it possible to examine athletic performance, research neuromuscular synchronization, and calculate the internal loading of the musculoskeletal system. Simulations can be utilized to determine the causes of pathological movement and create a rationale based on science for treatment planning. OpenSim is one such open-source physics-based computational tool for musculoskeletal modelling, simulation and analysing the kinematic and dynamic models (Delp et al., 2007a) for various human movements.

Human Motion Capture is the process of capturing the global position of the human movement kinematics such as the movement of the head, arms, torso, and legs. Over the years, motion capture technologies have progressed from manually annotating photos to wearable suits like marker-based optical trackers and inertial sensor-based devices (Colyer et al., 2018). The derived positional data of markers can be transferred into a musculoskeletal model in Opensim for further gait analysis and studies. The derived positional data of markers are transformed into generalized coordinates i.e., the joint angles using an inverse kinematics problem in OpenSim (Delp et al., 2007a).

Therefore, there is a need to look at the available human motion capture systems generally used in biomechanics. The current motion capture systems possess numerous advantages and disadvantages which are listed in the subsequent paragraphs. Optoelectronic Measurement Systems (OMS), Electromagnetic systems (EMS) and Ultrasonic Localisation systems are some of the marker-based systems available in the market today. These systems capture the human kinematics (3D locations of bony landmarks) using reflective markers and specialized camera systems. Apart from the marker-based systems other wearables such as IMUs are implemented for human motion capture (Colyer et al., 2018; van der Kruk & Reijne, 2018). Though optoelectronic systems are regarded as the gold standard motion capture technique in human motion analysis, there are some drawbacks are using a marker-based system. In the discipline of sports biomechanics and rehabilitation, these markers are intrusive, hindering the natural movement of the subject. Also, the effect of skin artifacts is an important factor in marker-based systems. Skin artifacts are caused by non-rigid skin tissues that stretch during highly dynamic human motions (Maletsky et al., 2007; Stancic et al., 2013; Windolf et al., 2008).

Some of the major challenges and limitations of marker-based methods are the high cost and limited portability of the marker systems. These systems requires a lot of pre-processing and post-processing time and also the capture space is limited. To overcome these challenges, the next step in human motion capture would be to incorporate artificial intelligence techniques such as machine learning and deep learning to study the locomotory system function, gait analysis, joint and bone mechanics (Mouloodi et al., 2021). Pose2Sim (Pagnon et al., 2021, 2022), is a method to reconstruct the musculoskeletal model from multiple camera images or videos. OpenCap (Uhlrich et al., n.d.), is another method to estimate the musculoskeletal OpenSim model from multiple mobile cameras without any use of specialized hardware. The above two methods are multi-step processes that incorporate state-of-the-art markerless motion capture systems or pose estimation methods such as OpenPose (Cao et al., 2018) or AlphaPose (Fang et al., 2016). The first step involves the extraction of key points and the second step is the inverse kinematics method similar to marker-based systems. These key points do not perfectly correspond to the musculoskeletal model’s joint locations. The key points are pixel points over the skin and clothing of the subject. Also, more than one camera is involved in

Pose2Sim (Pagnon et al.) and OpenCap (Uhlrich et al.) methods which require the calibration of cameras. The methods described above employ a multistep approach from an image to 3D key points and finally the musculoskeletal model. These key points do not perfectly correspond to the musculoskeletal model's joint locations. The key points are pixel points over the skin and clothing of the subject. Therefore, there is a need to develop new deep-learning models that can regress the musculoskeletal model directly from images and videos.

Marian et al., 2022 developed a single-step method called D3KE to estimate musculoskeletal kinematics from videos. The results demonstrate that the suggested end-to-end training is reliable and significantly beats a custom baseline method in terms of joint angle inaccuracy. The method also shows that it requires only one-camera and can run in real-time. The deep learning and machine learning algorithms are data-driven which requires tons of data to train on. Thus, this gives birth to the first aim of this paper where there is a need to develop a statistical human model or skinned human animated motions from a biomechanical model to build more synthetic training data. A database of synthetic data can be utilized to train deep-learning models. The aim of the study is to construct a pipeline to skin the musculoskeletal OpenSim model and create a statistical human animated model for various types of human movements which can be useful to build diverse datasets/databases.

These datasets are then useful to train deep-learning algorithms. However, the deep learning paradigm suffers from a major drawback called generalization. In fact, it becomes cumbersome to train a deep-learning algorithm for every gait movement and scenario. Therefore, the next aim of the study is to investigate the feasibility of a refinement step to optimize the biomechanical variables i.e., joint angles based on pixel loss as a post-processing step.

II. METHODS

This section consists of two sub-sections namely the skinning of the musculoskeletal model pipeline and the performance of D3KE on the synthetic data produced in the former step. Also, the feasibility study of the pixel loss refinement technique is explained as a follow-up topic on the later sub-section.

A. SKINNING OF MUSCULOSKELETAL MODEL PIPELINE

The process of "skinning" a musculoskeletal model, often referred to as "surface meshing" or "wrapping," entails fastening a mesh representation (usually a skin mesh) to the model's underpinning in order to visualize the model's motion with the skin (Murai et. Al, 2016). This method is frequently used in animation and biomechanics to produce lifelike representations of musculoskeletal activity.

The workflow employed to skin the musculoskeletal model incorporates three stages: Musculoskeletal model setup, extracting transformation of bodies, and the Human animation model setup. The workflow is summarized in Figure 1.

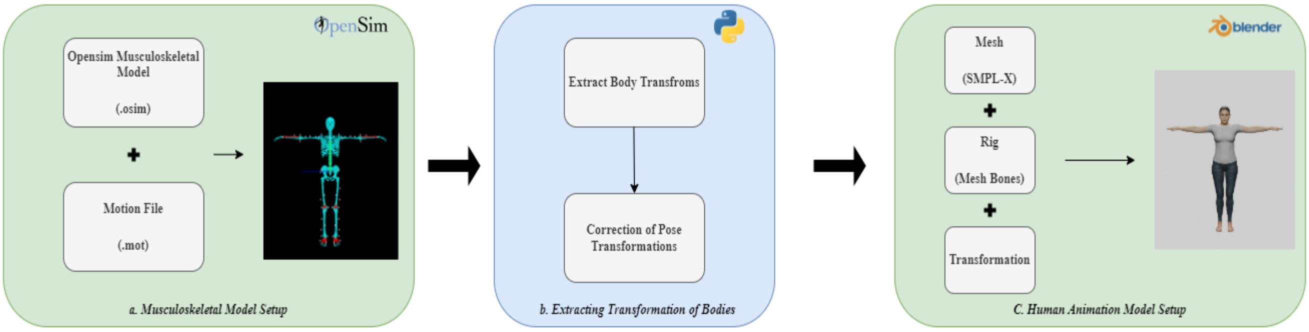

OpenSim
Opensim Musculoskeletal Model
(.osim)
Motion File
(.mot)
a. Musculoskeletal Model Setup
Extract Body Transfroms
Correction of Pose Transformations
b. Extracting Transformation of Bodies
blender
Mesh
(SMPL-X)
Rig
(Mesh Bones)
Transformation
C. Human Animation Model Setup

Figure 1. *The proposed pipeline to skin the biomechanical model involves three stages: Musculoskeletal model setup, Extracting Transformation of bodies, and the Human animation model setup.*

a. Musculoskeletal Model Setup

The first step in the pipeline is to define and set up a musculoskeletal model. The Full-Body Musculoskeletal Model developed by Rajagopal et al., 2016, is utilized further in this study. These bodies are illustrated in Figure 2 which make up the 3D skeletal model of a human.

The musculoskeletal model consists of 22 bodies namely a pelvis, a right and left femur, a tibia, talus, patella, calcaneus, and toes which represent the lower body, and a combined head and torso and right and left humerus, ulna, radius and hand to represent the upper body. From Figure 2, the bodies of interest are colored in blue and the bodies removed further in this study are the right and left patella, calcaneus, and radius which is colored in red.

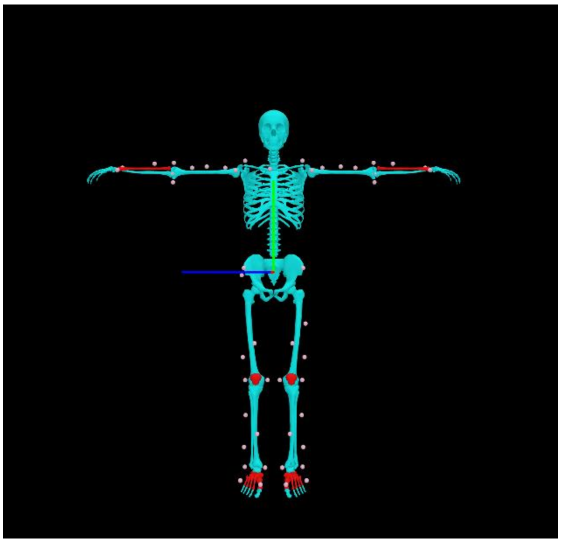


Figure 2. *The Full-Body Musculoskeletal Model. Blue-coloured are the bodies of interest and the red-coloured bodies are ignored further in this study.*

The model has 20 degrees of freedom (dof) in the lower body in which 14 dof for the two legs and six for the pelvis which includes three translational with respect to the origin. The upper body consists of 14 dof with the torso/lumbar joint having three dof. The joint kinematics or the degrees of freedom are presented in a motion file (.mot) as joint angles for each time frame. The joint kinematics along with a scaled musculoskeletal model can be used to study specific gait movements or patterns.

b. Extracting Transformation of Bodies

In the next step, we extract the transformations using the OpenSim python API. The transformation matrix stores the information on the rotation and translation of the bodies with respect to the world coordinates or the origin. The transformations of the bodies are extracted for every time frame of motion. The transformation matrix can be represented as

$$T_{body} = \begin{bmatrix} R_{body} & t_{body} \\ 0 & 1 \end{bmatrix} \quad (1)$$

where, R_{body} is a 3x3 matrix containing the 3D rotations with three column vectors for rotation about the x, y and z-axis. The t_{body} contains the translational elements t_x, t_y and t_z. From Figure. 3, the transformations of bodies are visualized where the global x-axis pointing out of the screen, the y-axis is the vertical axis and the z-axis is the horizontal axis. In Figure. 3, the local axis of each body is illustrated which has a rotation and translation from the global axis or the origin.

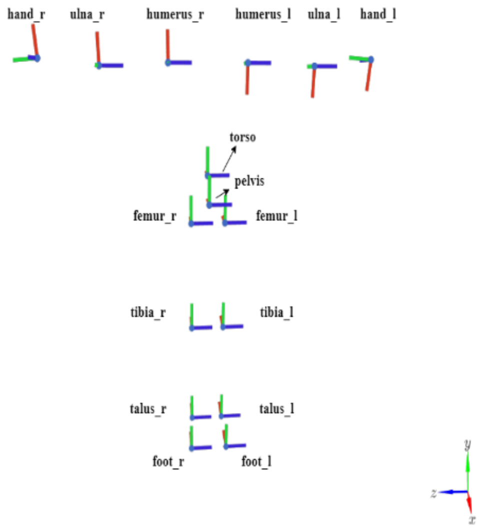


Figure 3. *Visualizing the extracted transformations of skeletal bodies from OpenSim*

Now, there is a need to correct the rotational offset of the transformation matrix to copy the matrix from OpenSim to the animation software called Blender. The correction of the rotational offset is to align the body axis in OpenSim with the rig axis in blender which is discussed in the next section.

From Figure 3, the lower body including the bodies pelvis, torso, right and left femur, tibia, talus and foot is rotated about 90^{o} using the Tait-Bryan angles order of axis rotation (XYZ) (Allgeuer & Behnke, 2018).

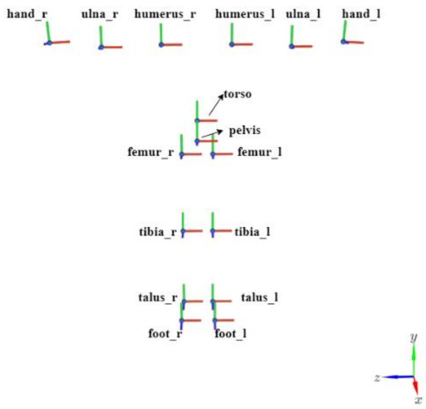


Figure 4. *Visualizing the corrected transformations of skeletal bodies from OpenSim*

The same XYZ axis rotation is employed to correct the rotations of the right humerus and ulna which is rotated about 90^{o} about the x and y-axis. The left humerus and ulna are rotated -90^{o} about the x-axis and 90^{o} about the y-axis. The right hand is rotated about 90^{o} about the z-axis and the left is rotated about 180^{o} about the x-axis and 90^{o} about the z-axis. These Tait-Bryan angles are then converted to a rotation matrix denoted by R$^{’}$ The new rotation matrix is given by

$$R_{new} = R_{body} * R' \qquad (2)$$

The new rotation matrix, R$_{new}$ is replaced in equation 1 and is given as follows,

$$T_{body} = \begin{bmatrix} R_{new} & t_{body} \\ 0 & 1 \end{bmatrix} \qquad (3)$$

The process of correcting the transforms is repeated for every frame or time step in the motion file. Therefore, the corrected transforms are ready to be served as an input in the next step where the musculoskeletal model is skinned.

c. Human Animation Model Setup

In the final step, the extracted transforms from opensim are copied to the virtual avatar to animate realistic human movements. In this step, the Blender computer graphics software is chosen for the current application due to its open-source nature. The two main ingredients of the human animation model are the mesh and the rig. The mesh chosen in this project is the SMPL-X (Pavlakos et al., 2019) human model which is built up of vertices derived from 3D scans of human subjects. The SMPL (Skinned Multi-Person Linear) is a popular deformable 3D human body model used in computer graphics, computer vision, and biomechanics. It uses a shape blend model and a linear blend skinning (LBS) model to describe the human body. The model is made up of a neutral body shape template mesh and a number of blend shapes (also known as morph targets) that represent various body types and poses.

The SMPL model uses weights to regulate how bones and joints affect each vertex of the template mesh. They show the degree to which each joint influences a particular mesh vertex when the skeleton is positioned. Controlling the deformation and movement of the skin requires these weights. Usually, a weight matrix is used to illustrate the weight distribution. Each vertex on the template mesh is represented by a row in the weight matrix, and each bone or joint on the skeleton is represented by a column. The weight (effect) of each bone or joint on the corresponding vertex is represented by the value in each cell of the matrix. The SMPL model's weight assignment procedure entails solving a regression problem to identify the ideal weights for every vertex. A sizable dataset of 3D scans of human bodies and the related skeletal positions is used to carry out the operation. Through a technique known as optimization, the SMPL model is trained on this dataset by adjusting the weights to reduce the disparity between the deformed mesh and the original 3D scans for various positions and forms. The weights are employed in real-time applications to distort the template mesh in accordance with the pose of the skeleton after they have been learned and allocated to each vertex. A realistic and natural deformation of the skin is produced when the skeleton is posed because each bone or joint's influence on the mesh's vertices is estimated based on the precomputed weights.

For simulations and animations of the human body to be precise and realistic, the weight distribution of the SMPL model is essential. In many computer graphics and computer vision applications that incorporate human body motion and deformation, it enables the model to adapt to different positions and body forms. The mesh is also equipped with clothing for the model to look realistic. To enable the movement and deformation of the mesh, a rig must be implemented over the mesh. The rig also known as the armature mimics the individual bodies of the musculoskeletal model (see Figure. 2 and Figure. 5) over which the mesh is placed for the so-called baseline T-Pose.

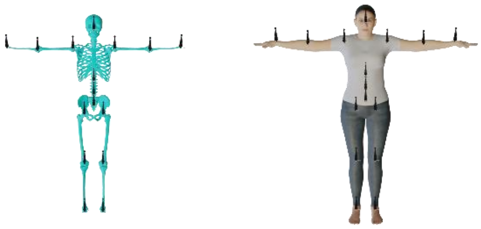


Figure 5. *Rig and Mesh structure where the Rig (black octahedral) mimics the bodies of the musculoskeletal model*

Apart from the 16 bodies of the musculoskeletal model, an additional 2 bodies as the rig is incorporated to control the upper torso and the head (see Figure. 6). The default SMPL mesh and rig does not have control points for the head and upper torso. The additional bodies are added to make the virtual avatar have realistic upper body movements.

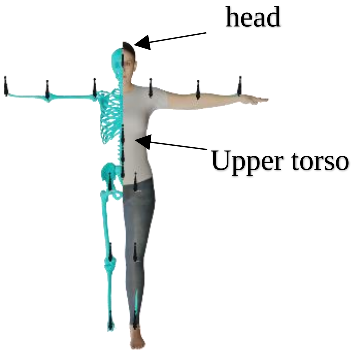


Figure. 6 *Illustrating the new control points for the head and upper torso*

The mesh is controlled by the rig with each rig having control over certain vertices of the mesh. The process of coupling the mesh and the rig is called parenting and the vertices of the mesh are allocated weights that are distributed over a certain region for each rig in the form of a heatmap as discussed earlier. For example, the weight distribution for the right and left femur is shown in Figure. 7.

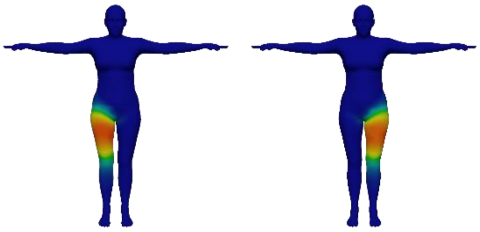


Figure 7. *The weight distribution of the right and left femur on the SMPL-X mesh*

Now, applying the corrected transformations of the bodies from the previous step to the rig in blender, will change the pose of the human animation model. The human movement animation is possible by applying the transformations for each time step and keyframing. Keyframing is an animation technique to store the value of an attribute, in this case, the transformation matrix at each instant of time. The pipeline can be evaluated by animating various kinds of human movement such as walking and running. The walking and running motion from the study of Rajagopal et. al research is used to evaluate the synthetic/animated movement of the person. The results of the skinning of the musculoskeletal model is evaluated qualitatively by observing the gait pattern. Also, overlaying the mesh on top of the skeletal system will give an overview of how well the skeletal system is inside the mesh without any bones sticking out.

B. PERFORMANCE OF D3KE AND FEASIBILITY STUDY OF PIXEL LOSS REFINEMENT

In this sub-section, the performance of D3KE on the synthetic videos produced in the previous section is evaluated. Deep neural networks must have the ability to generalize in order to be used in clinical applications. The network should be able to make accurate predictions on an unknown portion of data in addition to during training. This is crucial for applications in the therapeutic field because kinematic estimation must not be influenced by factors like a subject's body type, sex, or ethnicity. It is essential to use a sizable dataset and set aside a portion of it for testing.

The D3KE (Marian et al.,2022) is a deep neural network that takes a single-view video as input and directly estimates the joint kinematics i.e., the joint angles. The network predicts the musculoskeletal model parameters per frame by inferring the joint angles using a convolutional neural network. The global position of the full body is fixed and not predicted in this method. Also, the body scales are pre-determined which are the assumptions in this method. The predicted joint angles are smoothened by leveraging a sequence network by refining the joint angles. The D3KE workflow is illustrated in Figure 8.

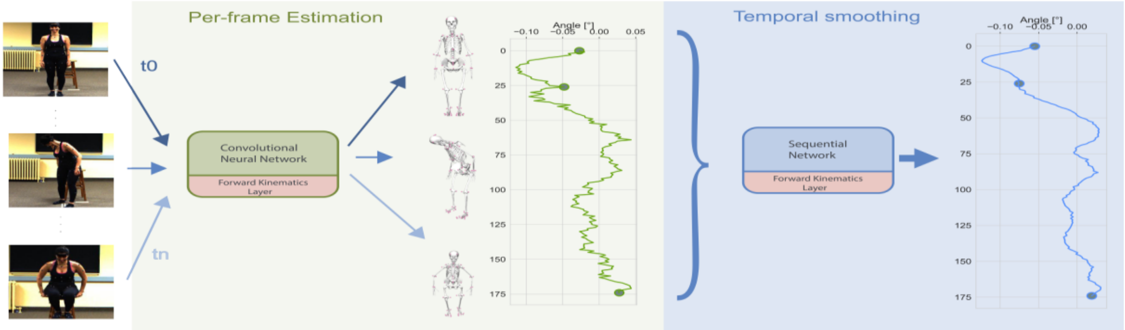


Figure 8. *Workflow of D3KE*

The performance of the above model is evaluated on unseen data which is a walking motion of the synthetic human. During normal walking or running, gait cycles are highly repetitive and cyclic. This repetition allows researchers and clinicians to capture the fundamental patterns of motion, muscle activation, and joint forces within a single cycle. If the gait pattern is consistent across cycles, analysing one cycle could provide a representative understanding of the entire gait pattern. In clinical settings, where the goal is to diagnose movement abnormalities or assess rehabilitation progress, a single gait cycle might be enough to detect significant deviations from a normal pattern. Clinicians often focus on identifying clear deviations or asymmetries that are readily apparent within a single cycle. Therefore, only one gait cycle of the walking motion is chosen for this study which is illustrated in Figure 9. separate experiments where the D3KE model takes a single-view video as input from the frontal and the sagittal plane. The metrics chosen to test the model are $\text{MAE}_{\text{angles}}$ Mean Average Error of Angles (in radians). The $\text{MAE}_{\text{angles}}$ are the average of joint angles estimated over all the frames in the gait cycle. Also, the mean per bony landmarks position error (MPBLPE) is used to evaluate the global position of the body. The MPBLPE is the average Euclidean distance of the markers present in the kinematic model over all the frames. Also, the performance of D3KE to predict the joint kinematics in the upper and lower extremity is compared.

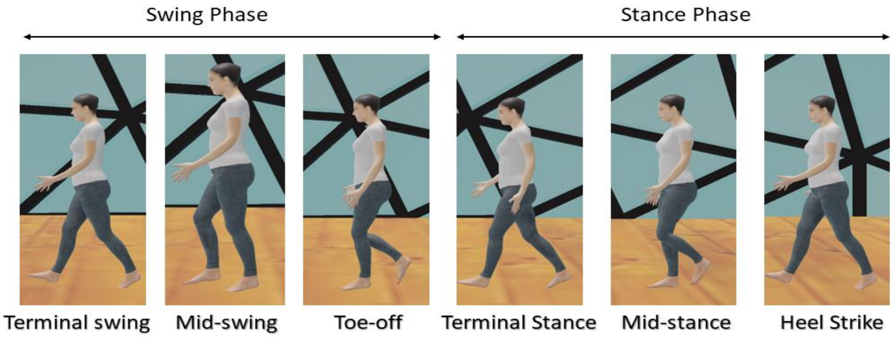


Figure 9. *Gait Cycle for evaluating the performance of D3KE*

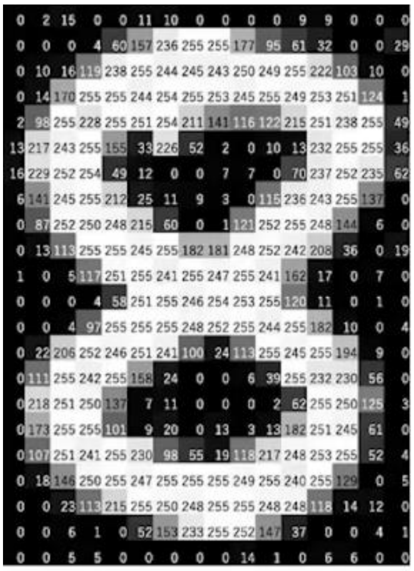
The deep learning models in general outperform the datasets they were trained and the performance of the deep learning model is less during the prediction of unseen examples. Therefore, the deep learning model, in our case the D3KE can be employed to derive a good initial estimate of joint kinematics namely the joint angles but the error between the ground truth and the predicted model can be further minimized by employing a refinement technique. The refinement step optimizes the biomechanical variables i.e., joint angles based on pixel loss or image matching between the ground truth image and the predicted image. The predicted image is rendered by skinning the predicted musculoskeletal model and joint kinematics.

First, from the ground truth image a mask of the human is extracted by removing the background from the image and converted to a grayscale image. The grayscale image has intensities ranging from 0 to 255 where 0 is a black pixel and 255 is a white pixel, refer Figure 10. The varying intensities of the pixel over a wider resolution give the grayscale image output. The next step is to formulate an objective function. The objective function renders a new image by using the initial estimate of the joint angles. By exploiting the use of skinning of the musculoskeletal model, we ca employ the pixel loss refinement. Finally, the difference between the pixel intensities of the actual image and the predicted image is returned as the pixel loss from the objective function, see Figure 11.

A non-linear least square fit function is set up to minimize the pixel loss by optimizing the joint angles. The pixel loss can be given as

$$\textit{Pixel Loss} = \left| \hat{Y}(q) - Y \right| \qquad (4)$$

where $\hat{Y}(q)$ is the model image or the predicted image intensity values that changes with the joint angles q and Y is the ground truth image intensities. The images are represented based on the intensity values. The brightness or darkness of each pixel can be interpreted as the intensity value in a grayscale image. Imagine it as a scale with pure black on one end and pure white on the other, with many tones of gray in between, refer Figure 10. The brightness level of each pixel in the image is determined by the intensity value that is allocated to it. Areas with higher intensity values are brighter, whereas those with lower intensity values are darker. The visual details and contrast in the image are produced by these intensity values.



0	2	15	0	0	11	10	0	0	0	0	9	9	0	0	0
0	0	0	4	60	157	236	255	255	177	95	61	32	0	0	29
0	10	16	119	238	255	244	245	243	250	249	255	222	103	10	0
0	14	170	255	255	244	254	255	253	245	255	249	253	251	124	1
2	98	255	228	255	251	254	211	141	116	122	215	251	238	255	49
13	217	243	255	155	33	226	52	2	0	10	13	232	255	255	36
16	229	252	254	49	12	0	0	7	7	0	70	237	252	235	62
6	141	245	255	212	25	11	9	3	0	115	236	243	255	137	0
0	87	252	250	248	215	60	0	1	121	252	255	248	144	6	0
0	13	113	255	255	245	255	182	181	248	252	242	208	36	0	19
1	0	5	117	251	255	241	255	247	255	241	162	17	0	7	0
0	0	0	4	58	251	255	246	254	253	255	120	11	0	1	0
0	0	4	97	255	255	255	248	252	255	244	255	182	10	0	4
0	22	206	252	246	251	241	100	24	113	255	245	255	194	9	0
0	111	255	242	255	158	24	0	0	6	39	255	232	230	56	0
0	218	251	250	137	7	11	0	0	0	2	62	255	250	125	3
0	173	255	255	101	9	20	0	13	3	13	182	251	245	61	0
0	107	251	241	255	230	98	55	19	118	217	248	253	255	52	4
0	18	146	250	255	247	255	255	255	249	255	240	255	129	0	5
0	0	23	113	215	255	250	248	255	255	248	248	118	14	12	0
0	0	6	1	0	52	153	233	255	252	147	37	0	0	4	1
0	0	5	5	0	0	0	0	0	14	1	0	6	6	0	0

Figure 10. *Grayscale image represented as intensities*

The least-square problem is set up in python using the scipy library which takes in the objective function and bounds for the joint angles which are optimized. The optimization of joint angles takes place until the maximum iterations are reached. From figure 11, after each iteration or function evaluation, a new joint angle is produced which again loops back into the objective function to find the new model image $\hat{Y}(q)$. The optimization result produces the optimized joint angles based on the pixel loss objective function.

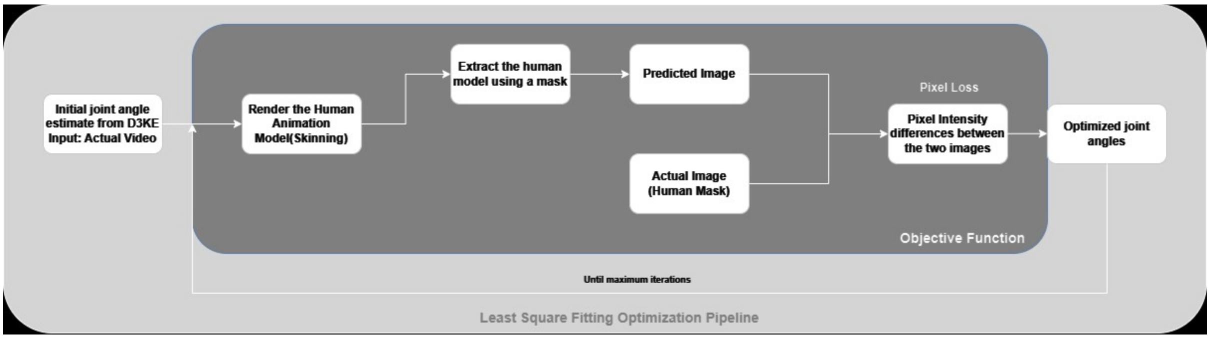

Initial joint angle estimate from D3KE Input: Actual Video
Render the Human Animation Model(Skinning)
Extract the human model using a mask
Predicted Image
Actual Image (Human Mask)
Pixel Loss
Pixel Intensity differences between the two images
Optimized joint angles
Objective Function
Until maximum iterations
Least Square Fitting Optimization Pipeline

Figure 11. *Least Square Fitting Optimization Hypothesis based on pixel loss where the video is chopped into frames. The initial joint angles are estimated from D3KE with the actual human animated video as input. These joint angles control the kinematics of the musculoskeletal model and a predicted virtual human model using the skinning technique is produced. A human mask separates the virtual human from the background from both the predicted and actual image. The difference between the two images is called the pixel loss.*

The feasibility of the hypothesis i.e., see Figure 11 is evaluated by designing different objective functions. The absolute mean of the image difference where the objective function outputs a scalar. The absolute mean of the image difference is a measure of the average absolute change in intensity between corresponding pixels of two images. It's often used to quantify the overall dissimilarity or variation between two images. The formula for calculating the absolute mean of the image difference is as follows:

$$Absolute\ Mean\ of\ Image\ Difference = \frac{1}{N}\sum|\hat{Y}(x,y) - Y(x,y)| \quad (5)$$

Where,
N is the total number of corresponding pixels in the images,
Σ represents the sum over all corresponding pixels,
$\hat{Y}(x, y)$ and $Y(x, y)$ are the intensity values of the corresponding pixels at coordinates (x, y) in the predicted and actual image respectively.

When comparing two images, it's important to understand how they differ in terms of pixel intensities. The absolute mean of the image difference provides a straightforward way to measure this dissimilarity. For each pair of corresponding pixels in the images, the absolute difference between their intensities is calculated. The absolute value is used to ensure that both positive and negative differences contribute equally to the measure.

Next, row and column-wise mean of the image difference where the objective function is a vector. For each row r in the image, calculate the average absolute difference between the corresponding pixel values in $\hat{Y}$ and Y along that row. Then, calculate the mean of these row-wise absolute differences. The vector contains elements corresponding to the number of rows.

$$Row - Wise\ Mean = \frac{1}{N_r}\sum|\hat{Y}(R,y) - Y(R,y)| \quad (6)$$

where,

N_r is the number of pixels in row R,
y ranges over the column pixels in row R,
$\hat{Y}$ (R,y) and Y(R,y) are the intensity values of corresponding pixels in the two images.
Finally, a bounding box and sliding window approach were also implemented.

$$Column - Wise\ Mean = \frac{1}{Nc}\sum|\hat{Y}(x,C) - Y(x,C)| \quad (6)$$

where,

N_c is the number of pixels in column C,
y ranges over the column pixels in column C,
$\hat{Y}$ (x,C) and Y(x,C) are the intensity values of corresponding pixels in the two images.

By calculating these row-wise and column-wise means of the absolute differences, you can get insights into how much the images differ along different directions.

Finally, a bounding box and sliding window approach were also implemented. To capture mean intensities using a sliding window approach in an image, you'll move a window across the image and calculate the mean intensity of the pixels within the window at each position. Here's how you can describe the formula for calculating mean intensities using a sliding window:

Given an image with width (W) and height (H), and a sliding window with width (W_w) and height (W_h), and step sizes (S_w) for horizontal movement and (S_h) for vertical movement, the formula for calculating the mean intensity within the sliding window at each position (i, j) can be expressed as:

$$Mean\ Intensity\ at\ Position\ (i, j) = \frac{1}{N} \sum_{x=i-1} \sum_{y=j-1} I(x + i, y + j) \quad (7)$$

where,

i ranges from 0 to (H - W_h) / S_h
j ranges from 0 to (W - W_w) / S_w
N is the total number of pixels within the sliding window (N = W_w * W_h)
(x, y) represents the pixel coordinates within the sliding window at position (i, j)
I(x, y) = $\hat{Y}$(x,y) - Y(x,y) is the intensity value of the pixel of the image differences at coordinates (x, y)

In simpler terms, for each position of the sliding window, you sum up the intensity values of all pixels within the window and then divide by the total number of pixels in the window to get the mean intensity. This sliding window approach allows you to calculate mean intensities at different positions across the image, helping you analyze variations in intensity and detect regions of interest based on their average brightness.

III. RESULTS

A. Results of Musculoskeletal Model Skinning

The musculoskeletal model skinning is evaluated qualitatively to visualize the proper transfer of transformations from OpenSim to the Human Animation model. The pipeline is evaluated for various kinds of human movements such as walking and running.

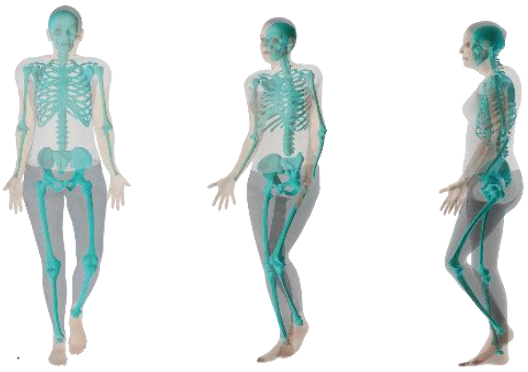


Figure 11. *The skin over the musculoskeletal model for the walking motion during mid-stance phase (a) Frontal View, (b) Isometric View and (c) Sagittal View*

First, the human walking motion is illustrated in Figure 11. The motion file corresponding to the walking motion is adapted from Rajagopal et al., 2016. Figure 11 depicts the human walking during the mid-stance phase of a normal gait cycle. The human animation model during the heel-strike phase is also illustrated in Figure 12. The overlay of the mesh on the skeletal system gives the readers an overview of the quality of the skinning method. Only the bones or bodies which controlled the mesh for various kinematic movements are incorporated in the overlay. For example, the feet, hands and radius bones or bodies are not utilized in this method to skin the musculoskeletal model because the other bodies were able to solve the purpose.

Further, the running motion is chosen where the flexion and extension of the arms, elbow, hip and knee are more when compared to the walking motion. The musculoskeletal skinning is illustrated in Figure 13.

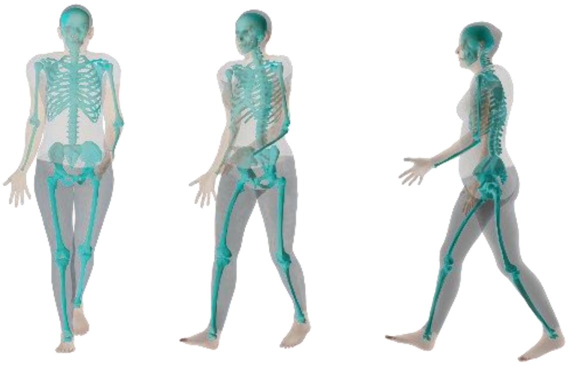


Figure 12. *The skin over the musculoskeletal model for the walking motion during the heel-strike phase (a) Frontal View, (b) Isometric View and (c) Sagittal View*

The below illustrations are for running motion during the mid-stance phase according to the normal human gait cycle.

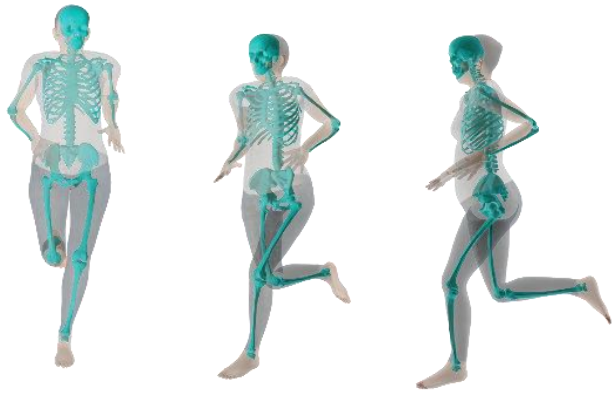


Figure 13. *The skin over the musculoskeletal model for the running motion during mid-stance phase (a) Frontal View, (b) Isometric View and (c) Sagittal View*

Also, the skinning of the skeletal model is qualitatively evaluated for the toe-off phase of the running motion. Figure 14 illustrates the toe-off phase during running human motion.

Final visualizations of the joint kinematics of the musculoskeletal model along with the human animation model from three views are illustrated. In all the illustrations the OpenSim skeletal model is superimposed with the corresponding skin to visually evaluate the method of skinning the musculoskeletal model.

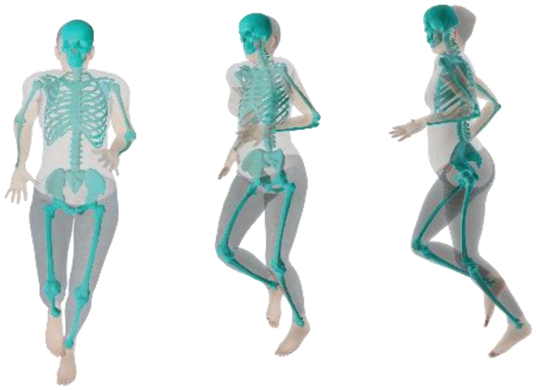


Figure 14. *The skin over the musculoskeletal model for the running motion during the toe-off phase (a) Frontal View, (b) Isometric View and (c) Sagittal View*

B. Results of D3KE performance and the feasibility of pixel loss refinement

The D3KE is tested on the virtual data. First the $\mathrm{MAE}_{\mathrm{angles}}$ and MPBLPE from two plane of view namely the sagittal and the frontal plane is given in Table 1.

Plane of Camera View	$\mathrm{MAE}_{\mathrm{angles}}$ (in degrees)	MPBLPE (in cm)	$\mathrm{MAE}_{\mathrm{angles}}$ (in degrees)	MPBLPE (in cm)
Sagittal	17.18	15.89	3.58	3.77
Frontal	16.15	13.15	3.54	3.69

Table 1. *MAE_{angles} and MPBLPE from the sagittal and frontal plane of the synthetic video.*

The inference of how well the upper and lower extremities performed is given in Table 2.

Plane of Camera View	Upper Extremity $\mathrm{MAE}_{\mathrm{angles}}$ (in degrees)	Lower Extremity $\mathrm{MAE}_{\mathrm{angles}}$ (in radians)
Sagittal	8.59	28.24
Frontal	6.87	24.57

Table 2. *MAE_{angles} of the upper and lower extremities from sagittal and frontal plane of the synthetic video.*

The inference of feasibility study of the pixel loss refinement designed based on different objective functions is tabulated and the possible explanations are given in Table 3. The objective function was iterated with different settings but the joint angles were not optimized. The parameter vector had several variables to optimize and the optimization problem becomes huge in terms of computation. This was the main possible reason for the failure of the method.

Objective Funtions	Findings and Remarks
Absolute mean of the image difference	This method is computationally costly and takes a lot of iteration to arrive at the solution. This is because the image is a matrix of 256x256 or any other resolution. In every iteration, thousands of gradient vector needs to be calculated making the problem complex and computationally costly.
Row and column wise mean of the image difference	This is an objective function with output as a vector. The jacobian matrix explodes with many pixels in the image.
Bounding Box and sliding window – Mean of each window	The problem persists and a solution is not obtained even with image compression

Table 3. *Feasibility study of pixel loss refinement based on different objective function design.*

IV. DISCUSSIONS

The aim of the study was to develop a pipeline to animate the musculoskeletal models by skinning the models with statistical human shapes. These virtual models can be a tool to create virtual datasets to train machine learning algorithms in the future. The statistical human shape employed is the SMPL-X model (Pavlakos et al., 2019). The method was evaluated qualitatively for walking and running motions. The human-animated model only requires the transformations of the musculoskeletal bodies to animate or transfer the joint kinematics to the underlying statistical mesh. The proper skinning of the biomechanical model is visualized in Figures 9-12. The main advantage of this method is the auto-scaling of the underlying mesh. The mesh and the rig needed to be parented manually which is a major drawback. Any change of mesh involves the process of rigging i.e., establishing the weights of the mesh vertices to the corresponding rig body. The feet and hands can be utilized further in the study to incorporate wrist and angle motions which are eliminated in this study. Fingers can also be animated using this method by treating each bones in the hands and feet indivually to run the underlying mesh/skin.

Unrealistic deformation during elbow flexion can be noticed in the elbows by comparing the walking and running motion from Figure 9-12. The deformation is evident in Figures 11(c) and 12(c) while viewed from the sagittal plane. The unrealistic deformation can be due to the improper weights in the elbow region of the mesh. As the mesh weights are constant during the time sequence, the deformations of the mesh are not dependent on the transformations of the bodies. Linear blend skinning is a technique by which the soft tissue dynamics of the skin can be incorporated into a mesh. Instead of the SMPL-X model, the SCAPE mesh model would provide improved results by eliminating the hyperflexion of the elbows (Anguelov et al., n.d.; Schleicher et al., 2021). Also, an auto-rigging approach can be employed in the future to avoid errors during manual rigging. A method that allows learning the auto-rigging of statistical human shapes will be a big leap in computer graphics animation. MoSh (Mathew et al., 2014) is a method that regresses human body shape and motion from sparse markers. This eliminated the step to extract transformations from the OpenSim model and can directly regress the shape and motion. The BMI of the person can also be incorporated providing meaningful results.

The method of skinning can be quantitatively evaluated in the future by acquiring 3D body scans of the participant and the corresponding motion of the person. The skinning and human animation can be inferred from the motion data compared across the 3D scan via a pixel loss metric. The human-animated model along with other environmental factors such as lighting, camera viewpoints and background information can build a rich and diverse dataset for machine learning and deep learning models.

The performance study of D3KE illustrates that deep learning models do not always generalise to the data it is trained on.The $\mathrm{MAE}_{\mathrm{angle}}$ from the sagittal plane is 0.3 radians which is higher when compared to the frontal plane. Also, the global position is better inferred from the frontal plane from the MPBLPE metric. Though, the D3KE underperforms and do not generalise well. This is due to the difference in training and the tested video data. From Table 2, the upper extremity is predicted better compared to the lower extremity where the $\mathrm{MAE}_{\mathrm{angle}}$ is less than 8^{o}. The performance can be improved by training the model with rich and diverse dataset created by skinning the musculoskeletal models.

Since pixel loss optimization lacks the constraints and structure needed to precisely anticipate joint angles from picture or video frames, it is not commonly becomes cumbersome for joint angle estimation. The goal of pixel loss optimization, which approaches joint angle estimation as a pixel-wise regression issue, is to reduce the pixel-level discrepancies between the predicted and ground truth joint angle images. However, it ignores the biomechanical limitations and connections that control the angles and motion of human joints. Joint angles require a more organized approach that takes into account the kinematic and anatomical characteristics of the human body because they are not clearly visible in pixel intensity values. With the increase in number of parameters needed to be optimized, the pixel loss refinement suffers badly from approaching a solution. Images' pixel intensity values can be influenced by a number of things, including the lighting, the subject's attire, the camera angle, and occlusions. Due of the ambiguity and noise this brings into the pixel-wise differences, it is difficult to determine joint angles with accuracy using only pixel loss optimization. To accurately depict the dynamic character of human motion, joint angle estimation frequently needs temporal information. Pixel loss optimization has limited performance in capturing joint angle changes over time because it does not readily include temporal dependencies. Joint angle estimation produces a high-dimensional output space by estimating multiple angles at various joints. Pixel loss optimization performs poorly and adds to the computational complexity when dealing with high-dimensional output spaces. The lack of interpretability in pixel loss optimization makes it challenging to comprehend how the model arrived at its conclusions. In biomechanics applications, where comprehension of the underlying biomechanical principles is crucial for analysis and decision-making, this can be a serious restriction. Pixel-wise differences optimization is computationally expensive and may not be suitable for real-time applications due to the high resource requirements.

V. CONCLUSION

- A pipeline to skin the musculoskeletal model is developed and evaluated qualitatively for different human movements such as walking and running. Qualitative results shows that realistic movements were achieved and further suggestions to evaluate the method quantitatively is discussed.
- With the synthetic data created by skinning the musculoskeletal model, the performance of D3KE was evaluated based on different planes of camera views. The D3KE model performed better on seen samples but had huge deviations in predicting the joint angles when tested in the virtual data. Also, the influence of the upper and lower extremity on

the performance was compared. The upper extremity was estimated better than the lower extremity.

- Finally, the findings and remarks of the pixel loss refinement feasibility study have been discussed in this paper. The pixel loss refinement suffers to find a solution due to the computational complexity of the problem.

REFERENCES

Allgeuer, P., & Behnke, S. (2018). *Fused Angles and the Deficiencies of Euler Angles*. https://doi.org/10.0/Linux-x86_64

Anguelov, D., Srinivasan, P., Koller, D., Thrun, S., Rodgers, J., & Davis, J. (n.d.). *SCAPE: Shape Completion and Animation of People*.

Cao, Z., Hidalgo, G., Simon, T., Wei, S.-E., & Sheikh, Y. (2018). *OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields*. http://arxiv.org/abs/1812.08008

Colyer, S. L., Evans, M., Cosker, D. P., & Salo, A. I. T. (2018). A Review of the Evolution of Vision-Based Motion Analysis and the Integration of Advanced Computer Vision Methods Towards Developing a Markerless System. In *Sports Medicine - Open* (Vol. 4, Issue 1). Springer. https://doi.org/10.1186/s40798-018-0139-y

Delp, S. L., Anderson, F. C., Arnold, A. S., Loan, P., Habib, A., John, C. T., Guendelman, E., & Thelen, D. G. (2007a). OpenSim: Open-source software to create and analyze dynamic simulations of movement. *IEEE Transactions on Biomedical Engineering*, *54*(11), 1940–1950. https://doi.org/10.1109/TBME.2007.901024

Delp, S. L., Anderson, F. C., Arnold, A. S., Loan, P., Habib, A., John, C. T., Guendelman, E., & Thelen, D. G. (2007b). OpenSim: Open-source software to create and analyze dynamic simulations of movement. *IEEE Transactions on Biomedical Engineering*, *54*(11), 1940–1950. https://doi.org/10.1109/TBME.2007.901024

Fang, H.-S., Xie, S., Tai, Y.-W., & Lu, C. (2016). *RMPE: Regional Multi-person Pose Estimation*. http://arxiv.org/abs/1612.00137

Keller, M., Zuffi, S., Black, M. J., & Pujades, S. (n.d.). *OSSO: Obtaining Skeletal Shape from Outside*. https://osso.is.tue.mpg.de,

Loper, M., Mahmood, N., & Black, M. J. (2014). MoSh: motion and shape capture from sparse markers. *ACM Trans. Graph.*, *33*(6), 220-1.

Maletsky, L. P., Sun, J., & Morton, N. A. (2007). Accuracy of an optical active-marker system to track the relative motion of rigid bodies. *Journal of Biomechanics*, *40*(3), 682–685. https://doi.org/10.1016/j.jbiomech.2006.01.017

Bittner, M., Yang, W. T., Zhang, X., Seth, A., van Gemert, J., & van der Helm, F. C. (2022). Towards Single Camera Human 3D-Kinematics. Sensors, 23(1), 341.

Mouloodi, S., Rahmanpanah, H., Gohari, S., Burvill, C., Tse, K. M., & Davies, H. M. S. (2021). What can artificial intelligence and machine learning tell us? A review of applications to equine biomechanical research. *Journal of the Mechanical Behavior of Biomedical Materials*, *123*. https://doi.org/10.1016/j.jmbbm.2021.104728

Murai, A., Endo, Y., & Tada, M. (2016). Anatomographic volumetric skin-musculoskeletal model and its kinematic deformation with surface-based SSD. IEEE Robotics and Automation Letters, 1(2), 1103-1109.

Nadeau, S., Betschart, M., & Bethoux, F. (2013). Gait analysis for poststroke rehabilitation: The relevance of biomechanical analysis and the impact of gait speed. In *Physical Medicine and Rehabilitation Clinics of North America* (Vol. 24, Issue 2, pp. 265–276). https://doi.org/10.1016/j.pmr.2012.11.007

Pagnon, D., Domalain, M., & Reveret, L. (2021). Pose2sim: An end-to-end workflow for 3D markerless sports kinematics—Part 1: Robustness. *Sensors*, *21*(19). https://doi.org/10.3390/s21196530

Pagnon, D., Domalain, M., & Reveret, L. (2022). Pose2Sim: An End-to-End Workflow for 3D Markerless Sports Kinematics—Part 2: Accuracy. *Sensors*, *22*(7). https://doi.org/10.3390/s22072712

Pavlakos, G., Choutas, V., Ghorbani, N., Bolkart, T., Osman, A. A. A., Tzionas, D., & Black, M. J. (2019). *Expressive Body Capture: 3D Hands, Face, and Body from a Single Image*. http://arxiv.org/abs/1904.05866

Rajagopal, A., Dembia, C. L., DeMers, M. S., Delp, D. D., Hicks, J. L., & Delp, S. L. (2016). Full-Body Musculoskeletal Model for Muscle-Driven Simulation of Human Gait. *IEEE Transactions on Biomedical Engineering*, *63*(10), 2068–2079. https://doi.org/10.1109/TBME.2016.2586891

Schleicher, R., Nitschke, M., Martschinke, J., Stamminger, M., Eskofier, B. M., Klucken, J., & Koelewijn, A. D. (2021). BASH: Biomechanical animated skinned human for visualization of kinematics and muscle activity. *VISIGRAPP 2021 - Proceedings of the 16th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, *1*, 25–36. https://doi.org/10.5220/0010210600250036

Stancic, I., Supuk, T. G., & Panjkota, A. (2013). Design, development and evaluation of optical motion-tracking system based on active white light markers. *IET Science, Measurement and Technology*, *7*(4), 206–214. https://doi.org/10.1049/iet-smt.2012.0157

Taborri, J., Keogh, J., Kos, A., Santuz, A., Umek, A., Urbanczyk, C., van der Kruk, E., & Rossi, S. (2020). Sport biomechanics applications using inertial, force, and EMG sensors: A literature overview. *Applied Bionics and Biomechanics*, *2020*. https://doi.org/10.1155/2020/2041549

Uhlrich, S. D., Falisse, A., Kidziński, Ł., Muccini, J., Ko, M., Chaudhari, A. S., Hicks, J. L., Delp, S. L., & Uhlrich, S. (n.d.). *OpenCap: 3D human movement dynamics from smartphone videos*. https://doi.org/10.1101/2022.07.07.499061

van der Kruk, E., & Reijne, M. M. (2018). Accuracy of human motion capture systems for sport applications; state-of-the-art review. In *European Journal of Sport Science* (Vol. 18, Issue 6, pp. 806–819). Taylor and Francis Ltd. https://doi.org/10.1080/17461391.2018.1463397

Windolf, M., Götzen, N., & Morlock, M. (2008). Systematic accuracy and precision analysis of video motion capturing systems-exemplified on the Vicon-460 system. *Journal of Biomechanics*, *41*(12), 2776–2780. https://doi.org/10.1016/j.jbiomech.2008.06.024