REMI: Real-Time Intuitive Feedback of Movement.

Dominick Stephens

ISE

Science and Engineering

CSIS

University of Limerick

Submitted to the University of Limerick for

Final Year Project

1. Supervisor: Dr. Salaheddin Alakkari
Computer Science and Engineering
University *of* Limerick
Ireland

Abstract

This Final Year Project, REMI (**R**eal-Time Monocular 3D Pose **E**stimation and **M**esh Visualisation for **I**ntuitive Movement Feedback) investigates the effectiveness of real-time, marker less 3D human pose estimation and dynamic visualization for providing immediate, intuitive feedback on movement quality, particularly for applications in fitness and physiotherapy. Current accessible methods often lack detailed, multi-dimensional feedback. This project addresses this gap by exploring the central research question: How effectively can a real-time system combining monocular 3D pose estimation and dynamic 3D mesh visualization provide immediate, intuitive visual feedback for human movement analysis?

To answer this, an end-to-end pipeline was developed and evaluated. This system captures movement via standard monocular video, performs real-time 3D pose and shape estimation, and renders the corresponding 3D human mesh dynamically. The potential of this immediate visual representation itself as a valuable form of feedback is assessed.

A comprehensive literature review contextualizes the research within the expanding field of AI-powered movement analysis and visualization, driven by advancements in AI and computational power. The report details the methodology, system architecture, implementation challenges, and evaluation plan designed to assess the feasibility, performance, and qualitative output of the proposed visual feedback system.

Declaration

I, Dominick Stephens (Student ID: [22343288]), hereby declare that the thesis entitled

"REMI: Real-Time Monocular 3D Pose Estimation and Mesh Visualisation for Intuitive Movement Feedback"

submitted in partial fulfilment of the requirements for the award of BSc in Immersive Software Engineering at the University of Limerick is entirely my own work. Except where explicitly acknowledged, no part of this thesis has been copied or taken from any other source, nor has it been submitted in the same or similar form for any other degree or professional qualification. All sources used and assistance received—including the guidance of my supervisor—are fully acknowledged.

Limerick, May 2025

Contents

List of Tables

List of Figures

Glossary

3DPW Dataset of *3D Poses in the Wild*, in-the-wild video sequences with synchronized IMU data for 3D pose estimation benchmarks

AMASS Archive of Motion Capture as Surface Shapes; unified SMPL-based aggregation of multiple motion-capture datasets

CoM Centre of Mass; point representing the average location of a body's mass in stability and physics-based pose estimation

CoP Centre of Pressure; point of application of ground reaction force under the feet, used to enforce stable grounded poses

DES Double Exponential Smoothing; time-series filtering with level (α) and trend (β) parameters to smooth noisy inputs

EMG Electromyography; measurement of muscle activation via surface electrodes, used as an indirect benchmark for simulated muscle forces

FPS Frames Per Second; metric for system throughput measuring processed frames per second in real-time evaluation

HSMR Human Skeleton and Mesh Recovery; regression method for anatomically accurate SKEL model parameters from single images

Hyfydy High-performance C++–based biomechanical simulation engine with 100× speedups over OpenSim for real-time predictive simulations

IK Inverse Kinematics; computation of joint angles to achieve desired end-effector positions in biomechanical models

IPMAN Intuitive Physics for Human Pose Estimation; physics-infused 3D pose estimator enforcing CoM–CoP stability via pressure heatmaps

Isaac Sim NVIDIA Omniverse robotics and simulation platform for deploying musculoskeletal RL agents in physics-based environments

MASS Muscle-Actuated Skeletal System; musculoskeletal simulation framework driven by imitation learning modeling individual muscle activations

MPJPE Mean Per Joint Position Error; average Euclidean distance between predicted and ground-truth joint positions in 3D pose estimation

One-Euro Filter Speed-adaptive low-pass filter adjusting smoothing coefficient based on motion velocity to reduce jitter

PAF Part Affinity Fields; encoding of limb associations in bottom-up pose estimation for occlusion handling

PA-MPJPE Procrustes-Aligned MPJPE; MPJPE after rigid alignment (scale, rotation, translation) isolating shape accuracy

POCO 3D pose and shape estimator with per-sample confidence scores for uncertainty-aware tasks such as pseudo-labeling

RMSE Root-Mean-Square Error; statistical measure of error magnitude between predicted and reference values

SCAPE Shape Completion and Animation of PEople; early parametric human-surface model later extended by SMPL for animated meshes

SKEL Biomechanical Skeleton Model with realistic joint degrees of freedom learned from AMASS for improved kinematic realism

SMPL Skinned Multi-Person Linear Model; differentiable body model encoding shape and pose via linear blend shapes and learned skinning weights

SMPL-X Extension of SMPL incorporating hands, face, and eyes for richer avatar expressiveness

SPIN SMPLify-in-the-Loop; hybrid regression-optimization approach refining SMPL predictions via feedback loops

STAR Sparse Trained Articulated human body Regressor; compact body model achieving SMPL-level generalization with sparse parameters

VR Virtual Reality

YOLO You Only Look Once

1

Introduction

1.1 Motivation and Context

The intersection of computer vision and biomechanical analysis represents a rapidly expanding research frontier, driven by significant advancements in computational power and sophisticated modelling techniques. AI-powered systems offer unprecedented potential for understanding human movement, moving beyond traditional laboratory constraints towards accessible, data-driven insights applicable in fields ranging from rehabilitation to elite sports training. Within this domain, the accurate simulation and analysis of the human musculoskeletal system based on visual input holds particular promise for personalised feedback and performance optimisation.

However, existing approaches for assessing movement quality, particularly in fitness and physiotherapy contexts accessible to consumers, often rely on simplified metrics or conversely inaccessible hardware systems. Many current systems primarily focus on kinematic analysis, such as verifying if a joint angle reaches a certain threshold or if a full range of motion is achieved during an exercise (e.g., checking hip-knee alignment in a squat). Chida et al. (2024) describes validating 2D video analysis by comparing lower limb joints (hip, knee, and ankle joints) angles during lunge using both 2D video analysis. While useful, such methods provide a limited output and often miss the complex interplay of muscle engagement and coordination that defines optimal and safe movement execution. They

lack the granularity to inform users about *how* their muscles are contributing to the movement.

Considerable progress has been made independently in both real-time 3D human pose estimation from monocular video (using techniques ranging from regression methods to physics-informed models) and detailed musculoskeletal simulation (leveraging platforms like OpenSim, developed by Delp et al. (2007) or advanced systems such as the Muscle-Actuated Skeletal System (MASS) described by Lee et al. (2019)). Pose estimation techniques can now capture 3D body configurations using standard cameras, while simulation frameworks can model the intricate dynamics of muscle forces and skeletal articulation.

Despite these parallel advancements, a significant gap remains in these two domains' seamless, real-time integration. Therefore, this project aims to bridge this gap by developing and evaluating a system that integrates these components. The central goal is to construct and assess an end-to-end pipeline that captures movement via standard monocular video, performs real-time 3D pose estimation, potentially drives a simulation, and renders the results dynamically to provide users with immediate, visual feedback. This research explores the feasibility and effectiveness of such an integrated system for enhancing movement understanding and feedback.

1.2 Problem Statement and Research Question

Based on the context discussed previously, it is evident that markerless body tracking holds significant potential for fitness and physiotherapy applications. However, current implementations often fail to provide nuanced, multi-dimensional feedback required for practical analysis. Many existing systems primarily focus on gross kinematic tracking (e.g., simple joint positions and 2D angles) but struggle to offer insightful information about the whole 3D nature of the movement, coordination, or qualitative aspects of form that go beyond basic metrics. This limitation hinders the ability to provide users with specific, actionable visual feedback to help optimise their technique or understand complex movements. The lack of readily accessible, dynamic 3D visual feedback from markerless systems

represents a key barrier to fully leveraging visual tracking technology in these domains.

Therefore, this research sought to address this specific deficiency by exploring the capabilities of modern real-time 3D pose estimation and visualisation techniques. The central research question guiding this project is:

How effectively can a real-time system combining monocular 3D pose estimation and dynamic 3D mesh visualisation provide immediate, intuitive visual feedback for human movement analysis?

Answering this question involves investigating the feasibility, performance, and qualitative output of an end-to-end pipeline that captures movement via monocular video, performs real-time 3D pose and shape estimation, and dynamically renders the corresponding 3D human mesh. The aim was to assess the potential of this immediate visual representation itself as a valuable form of feedback for movement understanding in contexts like fitness and physiotherapy.

1.3 Objectives and Contributions

To address the central research question, this work focuses on the following objectives:

- **Develop an Integrated Real-Time System:** Design and implement an end-to-end pipeline capable of capturing monocular video, performing real-time 3D pose and shape estimation, and dynamically rendering the corresponding 3D human mesh for immediate visual feedback.
- **Evaluate Visual Feedback Effectiveness:** Assess the potential of the generated real-time 3D visualisation as an intuitive and informative feedback mechanism for users analysing their own movements, particularly in fitness or physiotherapy contexts. This includes exploring how insights can be effectively conveyed visually, including muscle activation patterns derived from integrated simulation.

- **Investigate Performance and Feasibility:** Quantify the performance (e.g., latency, throughput) and evaluate the practical feasibility of the developed system using standard hardware, identifying key bottlenecks and limitations for real-time application.

This contributions of this project to future research include developing and evaluating this integrated pipeline, providing insights into the effectiveness of real-time 3D visualisation for movement analysis, and highlighting the challenges and potential of bridging computer vision-based tracking with dynamic visual feedback systems.

1.4 Thesis Structure

This thesis is organised into six chapters, structured to present the research from background context through to conclusions logically:

Chapter 2 provides a comprehensive review of the relevant literature. It explores the background domains of computer vision, 3D human pose estimation, body modelling techniques, and simulation environments. Existing related work is examined to contextualise this project and then to identify and address any gaps in the research.

Chapter 3 details the methodology employed. This chapter outlines the research design, describes the architecture of the real-time pose estimation and visualisation system developed, specifies the technology stack used, elaborates on the implementation process and challenges, and defines the criteria for evaluating the system's performance and effectiveness.

Chapter 4 presents the results and findings obtained from the system implementation and evaluation. This includes quantitative performance benchmarks (such as latency and throughput), qualitative results from the dynamic 3D visualisations, findings from specific case studies and/or examples, and an analysis of observed errors and limitations.

Chapter 5 discusses the findings presented in the previous chapter. This involves interpreting the results in the context of the research question, comparing the achieved system and its performance with existing solutions, considering the implications of the work for research and practice, and reflecting on the project's limitations.

Finally, **Chapter 6** concludes the thesis. It summarises the main contributions of the project, provides answers to the research question based on the evidence presented, discusses the potential broader impact of the work, and outlines key directions for future research and development.

2

Literature Review

2.1 Introduction

2.1.1 Similar Previous Works

Previous works attempting to address the problem of movement assessment from visual input often focused on pose estimation to verify if a complete range of motion was achieved within a particular exercise. For example, a common approach might involve checking if a point generated at the hips aligns correctly with the knees during a squat. While useful for basic form checking, this method typically yields a binary output (correct/incorrect) and, as such, overlooks much of the nuance involved in executing an exercise optimally and safely. These simpler systems often do not capture key indicators of effective and safe movement, such as appropriate joint placement throughout the entire dynamic range, muscle engagement patterns, inter-joint coordination, and the qualitative aspects of movement fluidity. This limitation restricts the depth of feedback provided to users aiming to refine their technique or understand complex motor patterns.

In more recent years, the field has seen advancements towards more sophisticated analyses. Notably, research by Sun, Tian and Qin (2023), explored the concept of directly driving a muscle simulation from video input. This work represented an important step in bridging computer vision with biomechanical simulation. They leveraged their Bidirectional Temporal Motion Recovery (BTMR) model (Sun,

Tang, Qu and Qin, 2023) to estimate pose and drive the simulation. While their paper successfully demonstrated the feasibility of such a pipeline and provided benchmark results, its scope was primarily focused on establishing this technical possibility rather than extensively exploring the potential applications of this technology for real-time feedback or detailed movement analysis in contexts like physiotherapy or fitness.

The challenge of real-time 3D pose estimation from monocular video, a critical component for accessible movement analysis systems, saw significant progress with the introduction of PhysCap (Shimada et al., 2020). PhysCap built upon earlier methods like VNect (Mehta et al., 2017), with a key contribution being the integration of physical plausibility constraints into the estimation process. This was a notable advancement as it represented one of the first successful attempts to ensure that the estimated poses were more realistic, for instance, by enforcing stable foot-ground contact and proper ground collision responses. These were aspects often missing or poorly handled in earlier real-time monocular 3D pose estimation frameworks.

Other systems have approached the problem with different methodologies and focuses. For instance, Sports2D (Pagnon and Kim, 2024) provides tools to compute 2D human pose and joint angles from video, which is particularly effective for analysing movements that primarily occur in sagittal or frontal planes. While valuable for specific applications, this 2D approach inherently differs from systems aiming for full 3D motion representation. Concurrently, solutions like OpenSimRT (Stanev et al., 2021) aim to bring the power of detailed biomechanical simulation, builing upon OpenSim (Delp et al., 2007) toolkit, into real-time applications. These systems typically rely on high-fidelity input data from marker-based motion capture or IMU sensor arrays to drive complex musculoskeletal models for in-depth biomechanical analysis. While OpenSimRT and similar platforms offer significant analytical depth, their reliance on specialised hardware and the complexity of their models present different use cases compared to markerless, monocular video-based systems designed for broader accessibility and more qualitative visual feedback. These varied approaches, from physically plausible 3D

monocular capture to 2D planar analysis and detailed sensor-driven biomechanical simulation, illustrate the diverse efforts to leverage technology for movement understanding, setting the stage for systems like REMI that aim to synthesise aspects of these advancements for intuitive, real-time 3D visual feedback from a single camera.

2.2 Computer Vision

Computer vision has evolved from early heuristic-based algorithms in the 1960's to the data-driven methods enabled by convolutional neural networks (CNN'S). LeCun et al. (1989) first introduced the advent of back propagation training for CNNs and later advances in hardware acceleration, particularly GPU usage introduced by Krizhevsky et al. (2012), facilitated breakthroughs in real-time image processing. Frameworks like Viola and Jones (2001) pioneered fast object detection, laying the groundwork for deep learning-based methods such as YOLO from Redmon et al. (2016) and HRNET Wang et al. (2020). These developments enabled robust 2D keypoint detection at scale, a critical precursor to the 3D pose estimation techniques employed in modern biomechanics and simulation workflows.

2.3 Pose Estimation

Pose estimation, a general problem in computer vision, aims to identify the location and orientation of an item or human. In the case of human pose estimation, we typically accomplish this by estimating the locations of various key points like hands, heads, elbows, and so on. Our machine-learning models seek to track these key points in photos and videos.

2.3.1 2D Pose Estimation

Bottom-up vs Top-down All human pose estimation techniques fall into two main categories: bottom-up and top-down approaches. Bottom-up approaches

assess each body joint and organise them into a certain stance. Top-down approaches first run a body detector and then identify body joints inside the identified bounding boxes.

The model uses a bottom-up technique to identify every instance of specific key point (for example, all left hands) in an image before attempting to assemble clusters of key points into skeletons for different objects.

In contrast, a top-down technique involves first using an object detector to crop each instance of an item, after which the network estimates the key points within each cropped region.

Role of YOLO A prominent family of models often employed within the top-down framework, particularly when real-time performance is critical, is YOLO (You Only Look Once) Redmon et al. (2016). Originally groundbreaking for its speed and efficiency in object detection, the YOLO architecture has been extended in recent iterations to incorporate human pose estimation. These YOLO-based pose estimators typically first utilise the detection capabilities to draw bounding boxes around individuals and then predict the 2D keypoint locations within those boxes. This implements the top-down strategy, making YOLO variants popular for applications requiring fast, efficient, multi-person pose estimation from video streams.

Figure 2.1: YOLOv11 Pose Estimation (Redmon et al., 2016)

2.3.2 3D Pose Estimation

3D pose estimation expands 2D pose estimation while determining how this model fits into 3D space.

Monocular vs Multi camera This paper explored monocular approaches to solving 3D Pose Estimation. Pose2Sim proposed by Pagnon et al. (2022) is the leading paper and introduces a multi-camera approach to this subject. There are pros and cons to using multiple cameras for this application, namely that multiple cameras ensure that no parts of the capture are occluded and can be used to triangulate the 3D pose most accurately. While having multiple cameras brings greater accuracy with this approach, it increases the setup cost and makes applications in the wild more difficult. In addition to this, extensive calibration is needed before capture, which increases the barrier to generating 3D Poses. Sports2D, as outlined by Pagnon and Kim (2024), builds upon Pose2Sim and addresses some issues by introducing a single camera approach. However, it is limited by the fact that the motion must lie in the sagittal or frontal plane. While its outputs are impressive, this is too big of a limitation for an 'in-the-wild' approach, as the author describes in Pagnon and Kim (2024).

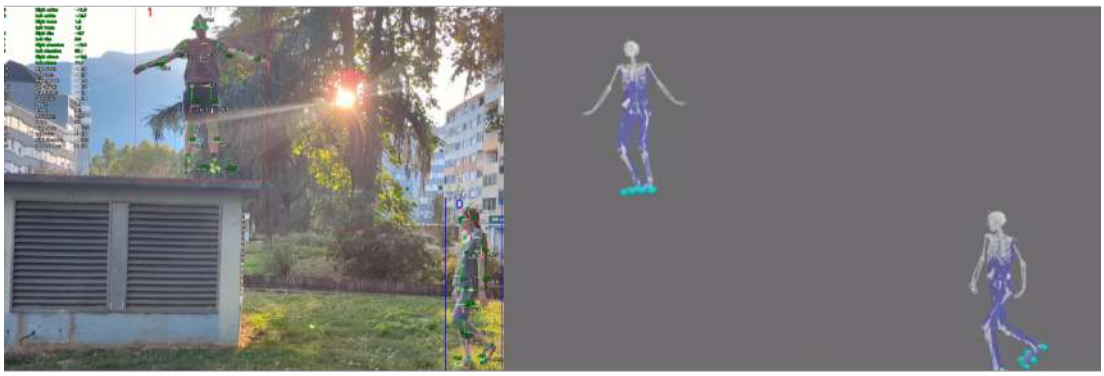

Figure 2.2: Sports2D (Pagnon and Kim, 2024)

Deep Learning Regression Methods Deep learning regression methods have revolutionised 3D human pose and shape estimation by enabling direct prediction

of 3D body parameters from single RGB images. A seminal approach in this domain is SMPLify, created by Bogo et al. (2016), which fits the SMPL model to 2D joint detections by minimising reprojection error, albeit with high computational cost per frame. To address efficiency concerns, SPIN attributed to Kolotouros et al. (2019) introduced a hybrid method that combines a neural network regressor with iterative optimisation, refining predictions through a feedback loop that enhances accuracy over time.

Building upon these foundations, PhysCap, presented by Shimada et al. (2020), integrates physics-based constraints into the estimation process, ensuring physically plausible and temporally stable 3D motion capture from monocular video at real-time speeds.

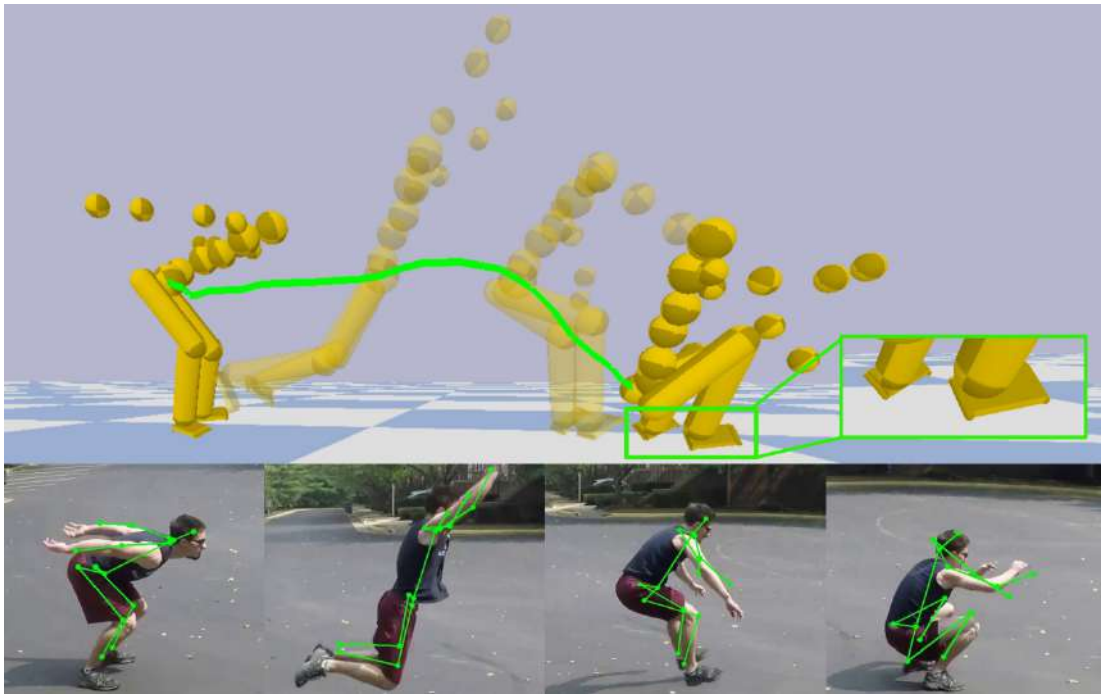

Figure 2.3: Physcap (Shimada et al., 2020)

In contrast, BTMR (Bidirectional Temporal Motion Recovery) introduced by Sun, Tang, Qu and Qin (2023) emphasises temporal consistency: it uses a transformer-based generator to capture both forward and backward temporal features and a Bi-LSTM discriminator to enforce realistic sequential motion, thus

producing smoother, more accurate pose trajectories. BTMR learns from motion datasets (e.g. AMASS) to improve metrics like joint error and acceleration, but does not itself impose explicit biomechanical forces.

Jiang, Xie and Li (2024) develops RTMW, a real-time approach to deep learning regression of 3D pose estimation, which extends real-time multi-person whole-body pose estimation by incorporating hierarchical encoding modules and achieving state-of-the-art performance on benchmarks like COCO-Wholebody created by Jin et al. (2020).

Complementing these methods, POCO from Dwivedi et al. (2023) introduces a framework that not only estimates 3D pose and shape but also provides per-sample confidence scores, facilitating downstream tasks such as pseudo-labelling and uncertainty-aware video inpainting. Collectively, these approaches demonstrate the progression from optimisation-based methods to sophisticated deep learning models that offer improved accuracy, efficiency, and applicability in diverse scenarios.

Figure 2.4: POCO Confidence Values (Dwivedi et al., 2023)

Physics and Biomechanical Inferred Models Recent literature has increasingly incorporated physics-based constraints into human pose recovery to ensure biomechanical plausibility. For example, the Ipman framework (Intuitive Physics for Human Pose Estimation) originating from Tripathi et al. (2023) infers a pressure "heatmap" on the body and explicitly computes the centre of pressure (CoP) and centre of mass (CoM) to encourage stable, ground-contacting poses. Its regression variant IPMAN-R extends a standard pose regressor with stability and contact losses based on the CoM–CoP relationship. It yields 3D reconstructions

that avoid unnatural floor penetration and exhibit more physically plausible support (stable, grounded poses).

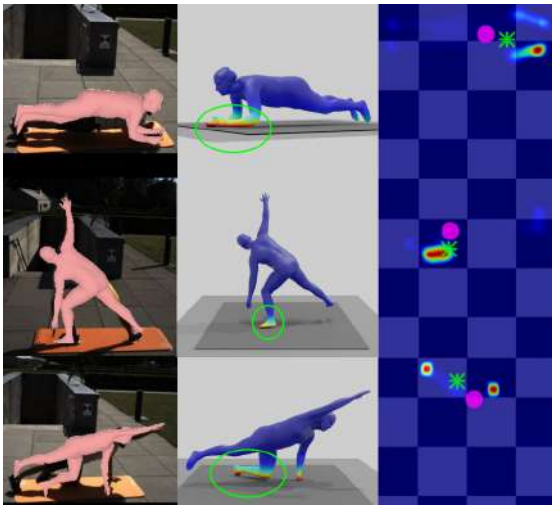

Figure 2.5: Ipman - Pose, CoM and CoP (Tripathi et al., 2023)

Another approach from Shin et al. (2024), WHAM (World-grounded Humans with Accurate Motion), recovers motion in a global coordinate frame by predicting the body's root trajectory and foot contacts. WHAM leverages previous motions and video cues to predict global orientation and root velocity, and explicitly detects foot–ground contact events to refine the trajectory on uneven terrain. By modelling foot contact and refining the root motion, WHAM produces a physically plausible world-space motion even with moving cameras or stairs, effectively grounding the motion in the scene.

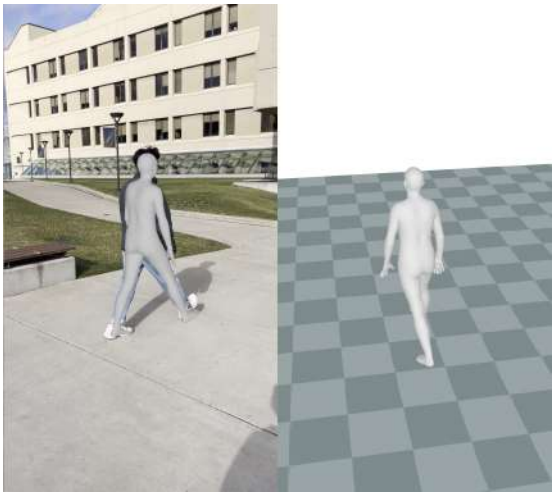

Figure 2.6: WHAM (Shin et al., 2024)

Similarly, emphasising physics, Sun, Tian and Qin (2023) construct a full musculoskeletal simulation from video: they first build a Lagrangian dynamics model of a human (skeleton + muscles) and used pose estimation to extract a reference motion. A hierarchical controller then drives the simulation with two layers—one tracking the reference trajectory and another adjusting muscle torques—to minimise pose and force errors.

This muscle-driven control enforces that the reproduced motion obeys realistic muscle forces and joint torques and remains stable under perturbations, yielding high-fidelity, physically grounded reconstructions. Finally, Yan et al. (2025)showcases the HSMR (Human Skeleton and Mesh Recovery) method, which explicitly incorporates an anatomically accurate skeleton: it regresses the parameters of the SKEL body model, a contribution from Keller et al. (2023) (which has biomechanically realistic joints) from a single image using a transformer network. By enforcing the SKEL model's joint limits and structure, HSMR regularises pose estimates so that joint rotations remain realistic (especially in extreme poses), overcoming the unnatural articulations often seen in simpler body models such as SMPL as outlined by Loper et al. (2015), which will be discussed further in section 2.4.

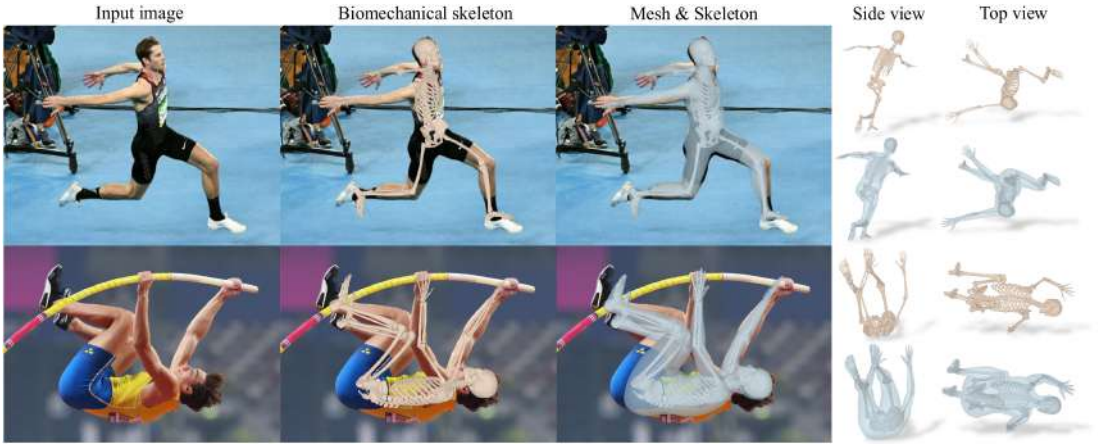

Figure 2.7: HSMR with SKEL Yan et al. (2025)

Together, these approaches illustrate a progression from purely kinematic reconstructions toward models that embed pressure, contact, CoP/CoM considerations and full musculoskeletal dynamics, ensuring that inferred motions and poses respect the laws of physics and anatomy.

2.3.3 Applications of 3D Pose Estimation

3D human pose estimation applications have expanded rapidly across domains such as sports training, virtual reality, and telepresence, driven by improved accuracy, robustness, and real-time performance. Integrating Internet of Things (IoT) technology with deep learning-based pose estimation enables real-time athletic performance monitoring, injury prevention, and motion optimisation in sports training. Xi et al. (2024) demonstrated how a dual-channel network combining Spatiotemporal Transformers and Temporal Convolutional Networks could process multi-modal IoT data streams to accurately and efficiently estimate complex, high-speed human motions, overcoming the latency and accuracy challenges traditional CNN- or RNN-based approaches face in dynamic, cluttered environments.

Meanwhile, in immersive virtual reality, Jiang et al. (2024) introduced SAD-Net, a self-attention distillation network capable of real-time 2D pose estimation from monocular cameras, enabling low-latency, pose-driven 3D avatar generation that can be seamlessly integrated into VR headsets like the VIVE Cosmos.

By coupling lightweight pose mapping algorithms with knowledge distillation techniques, SADNet addresses the computational and robustness limitations of prior VR avatar generation methods that depended heavily on costly sensor arrays and multiview rigs.

These approaches illustrate a broader trend: as real-time monocular pose estimation matures, its applications extend beyond offline motion capture toward interactive systems that demand high throughput, low latency, and spatially coherent outputs. Critically, both fields—sports training and VR immersion—rely not merely on raw pose estimation accuracy but on preserving biomechanical plausibility and smooth temporal transitions, highlighting the importance of robust multi-frame reasoning and integrating camera and scene models into modern 3D pose pipelines.

Kang et al. (2024) demonstrates a compelling application of 3D human pose estimation in the context of rehabilitation exercise movements, particularly for elderly patients recovering from conditions such as stroke. This work addresses the significant challenges faced in rehabilitation, such as long recovery periods, difficulties for patients in perceiving progress, and mobility limitations hindering access to traditional monitoring. The authors argue that while sensor-based or mechanical systems exist, they often suffer from high costs, lack of adaptability to individual patient needs, and difficulties in deployment outside clinical settings.

Kang et al. (2024) propose using computer vision, specifically a multi-stage 3D human pose estimation method based on a monocular camera, as a more accessible and adaptable alternative. Their system aims to provide detailed, quantitative data on patient movements, which can help clinicians monitor recovery, optimise personalised rehabilitation plans, and provide better feedback to patients.

They utilise a combination of HRNet (Wang et al. (2020)) for initial high-resolution 2D keypoint detection, followed by a Transformer-based architecture (OCTransformer) to lift the 2D poses to 3D while capturing spatio-temporal relationships.

Crucially, their model incorporates specific modules designed to handle everyday challenges in real-world and clinical settings, such as limb self-occlusion during movement and occlusion caused by environmental factors or rehabilitation aids (like walking frames). By explicitly addressing these occlusion scenarios

using techniques like their PAF correction module, they aim to improve the accuracy and reliability of the 3D pose data, making the system more robust for practical use.

2.4 Filtering Techniques for Real-time Pose Estimation

Real-time pose estimation systems commonly suffer from jitter and noise that can obscure subtle movements and degrade user experience. This section reviews prominent filtering techniques to mitigate these issues while maintaining responsiveness to actual movements.

One-Euro Filter introduced by Casiez et al. (2012*a*), provides a simple yet effective speed-based low-pass filter explicitly designed for noisy input in interactive systems. This filter adapts its smoothing coefficient based on movement speed, allowing it to reduce jitter during slow movements while remaining responsive during fast movements. The One-Euro filter has demonstrated effectiveness in virtual and augmented reality applications where real-time visual feedback is critical. Recent implementations, such as Wu et al. (2023), have shown their efficacy in handling jitter during hand motion tracking for virtual model control in real-time applications.

Single exponential smoothing applies a weighted average to the most recent observation and the previous smoothed value. Aldrin and Damsleth (1989) Outlined that a single parameter (α) controls the degree of smoothing, which determines the weight given to the most recent observation versus historical data. Dangkham (2019) illustrated that higher alpha values (e.g., 0.7) create more responsive filters that react quickly to changes but retain more noise, while lower values (e.g., 0.3) provide smoother output at the cost of increased latency.

Double exponential smoothing extends the single exponential approach by incorporating a second equation to account for trends in the data. This technique

utilises two parameters: α for the level and β for the trend component (Aldrin and Damsleth (1989)). Studies evaluating the performance of double exponential smoothing as a filter suggest that parameter values close to 0.5 are most appropriate when DES functions as a noise reduction mechanism rather than a forecasting tool, as shown by Mukherjee et al. (2019). The additional trend component makes this approach suitable for tracking movements with consistent directional changes.

Moving average filters smooth data by taking the mean of a fixed window of consecutive observations. The window size directly impacts the balance between smoothness and responsiveness. Larger windows (e.g., 10 frames) provide significant smoothing but introduce greater latency, while smaller windows (e.g., three frames) maintain responsiveness at the cost of less effective noise reduction. Comparative studies in mobile augmented reality applications (Dangkham (2019)) have shown that exponential moving average typically outperforms simple moving average regarding error reduction when handling sensor orientation data.

Each filtering technique presents a trade-off between noise reduction and responsiveness. The appropriate choice depends on application-specific requirements, with more aggressive smoothing beneficial for visualisation and less smoothing preferred for applications requiring immediate feedback. As presented in subsequent sections, the quantitative evaluation of these filters guides the selection of the optimal filtering approach based on specific performance metrics.

2.5 Body Modelling and Simulation

2.5.1 Parametric Statistical Body Models

A significant approach in human body modelling involves representing the human surface shape using a relatively small set of interpretable parameters, typically learnt from large datasets of 3D body scans. While powerful for capturing statistical variations in shape and pose across populations, this data-driven methodology inherently risks inheriting biases in the specific scan datasets used for training (e.g., demographics, body types) may struggle to generalise to body shapes or poses outside the training distribution.

The Max Planck Institute for Intelligent Systems and Loper et al. (2015) significantly advanced the field by introducing the Skinned Multi-Person Linear Model (SMPL). SMPL encodes body shape and pose variations using linear blend shapes and learned skinning weights, offering a differentiable model widely adopted in computer vision. However, its reliance on linear transformations is a simplification that may not fully capture the complex, non-linear ways human bodies deform, particularly during complex movements or for individuals with diverse body compositions. Furthermore, its accuracy is fundamentally tied to the thousands of scans it was trained on, raising questions about its applicability to under-represented populations.

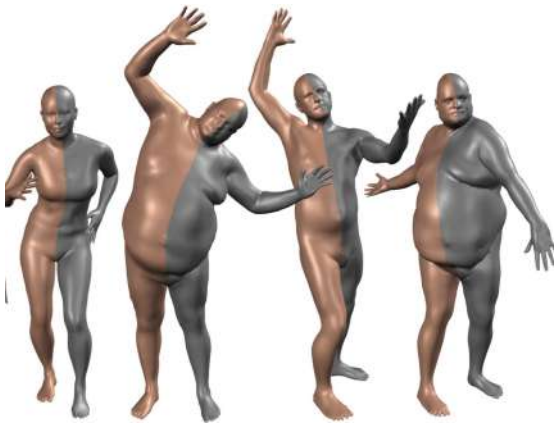

Figure 2.8: SMPL Loper et al. (2015)

Extensions like SMPL-X attributed to Pavlakos et al. (2019) aimed to increase

expressiveness by adding articulated hands, face, and eyes. While this allows for richer virtual avatars, the added complexity must be critically evaluated against the specific application's needs. For tasks focused on gross body movement and biomechanics, the fidelity of hand or facial detail may be an unnecessary overhead, and the quality of this added articulation may not match specialised models.

Figure 2.9: SMPL-X Pavlakos et al. (2019)

Critiques of SMPL's potentially redundant parameterisation led to models like STAR (A Sparse Trained Articulated Human Body Regressor) by Osman et al. (2020). STAR achieves comparable or even better generalisation performance using significantly fewer parameters (around 20% of SMPL's), suggesting SMPL might overfit the training data.

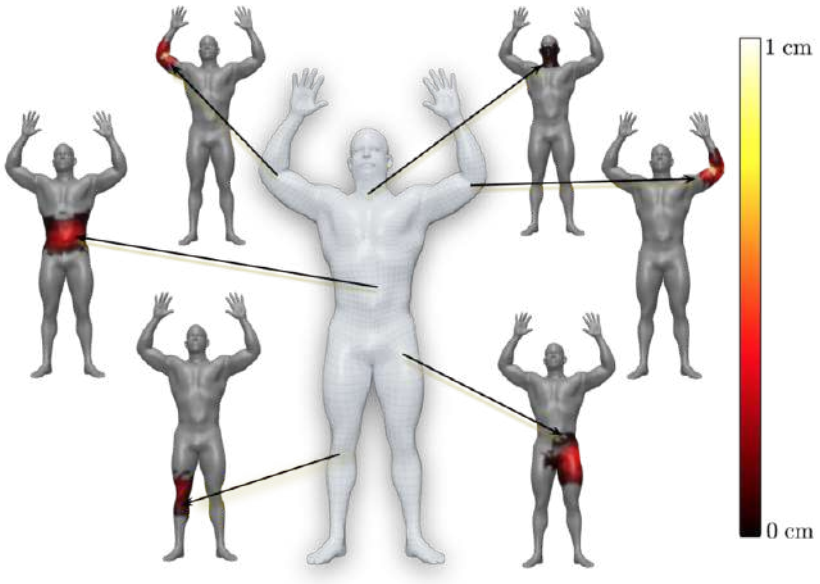

Figure 2.10: STAR Osman et al. (2020)

Crucially, for applications demanding biomechanical insight, these parametric surface models possess a fundamental limitation: they represent only the external body shape, lacking any explicit representation of the underlying anatomical skeleton, joints, or musculature. Their parameters describe surface geometry, not internal mechanics. Therefore, while useful for visualisation or as an initial geometric target, their direct applicability to simulating muscle forces or joint torques is limited. Inaccuracies or biases in the fitted SMPL or STAR shape could propagate errors into subsequent biomechanical analyses that rely on estimating anatomical landmarks from the surface. This necessitates moving beyond purely statistical surface models towards approaches that explicitly incorporate anatomical structure, as discussed in the following section.

2.5.2 Skeleton Based Biomechanical Models

Recognising the limitations of purely surface-based parametric models for biomechanical analysis, alternative approaches focus on integrating a parametric mesh with a more anatomically accurate skeleton. This enables realistic joint articulation and paves the way for downstream biomechanical calculations.

One such approach is SKEL (From Skin to Skeleton) by Keller et al. (2023), which explicitly improves upon SMPL by incorporating a Biomechanical Skeleton Model (BSM) featuring more realistic joint Degrees of Freedom (DoFs) (46 vs. SMPL's 72), learned from the AMASS (Archive of Motion Capture As Surface Shapes) dataset created by Mahmood et al. (2019).

Figure 2.11: SKEL Keller et al. (2023)

While SKEL offers enhanced kinematic realism at the joint level, it still lacks the explicit representation of muscle actuators needed for dynamic force and activation analysis. Other methods, like BASH (Biomechanical Animated Skinned Human), focus on visualisation by scaling a surface mesh (SCAPE (Shape Completion and Animation of People) developed by Anguelov et al. (2005)) to match proportions from established biomechanical toolkits like OpenSim by Delp et al. (2007), rigging an articulated skeleton, and mapping simulated muscle activations onto the mesh. This provides valuable visualisation but often relies on offline or separate simulation steps, potentially sacrificing the fidelity of a fully integrated dynamic simulation.

A more comprehensive approach is embodied by the Muscle-Actuated Skeletal System (MASS) from Lee et al. (2019), a simulation software designed to model the complex interplay between muscle actuations and skeletal movement. Its key technical contribution lies in a scalable, two-level imitation learning algorithm that controls a complex full-body musculoskeletal model with hundreds of muscles. However, this increased anatomical fidelity comes at a significant cost in terms of computational complexity and the difficulty of personalising such detailed models to individual subjects. The feasibility of using such complex models

in real-time feedback scenarios, crucial for this thesis, remains challenging without substantial optimisation or model simplification.

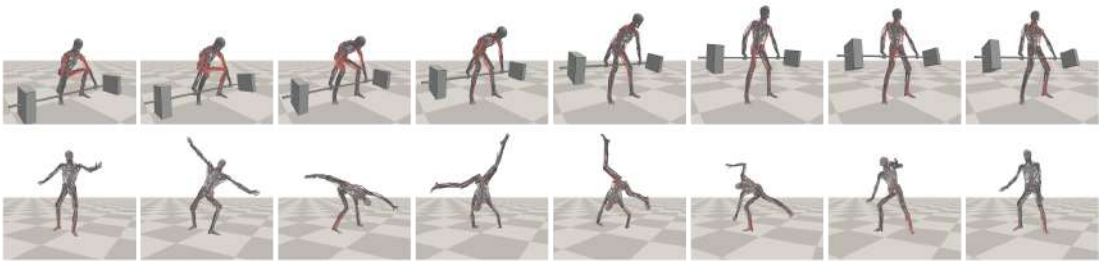

Figure 2.12: MASS Lee et al. (2019)

Applications leveraging MASS highlight both its potential and the associated complexities. Nvidia and Xu et al. (2022) showed that using MASS within Isaac Sim™ for reinforcement learning demonstrated the possibility of training agents to perform tasks like running. However, achieving stable and biologically plausible gaits through reinforcement learning often requires intricate reward engineering. It can sometimes result in unnatural movement strategies that exploit simulation physics rather than replicating human biomechanics. Similarly, Sun, Tian and Qin (2023) presented a novel pipeline to drive MASS simulations from input video. Their two-phase approach, using pose estimation followed by inverse kinematics and motion imitation, is promising but introduces multiple stages where errors can accumulate. The accuracy of the initial pose estimate and the ability of IK and control policies to faithfully replicate complex or contact-rich human motion remain critical hurdles.

Furthermore, acknowledging the inherent simplifications within even these sophisticated biomechanical models, such as the Hill-type Muscle model commonly used in muscle simulations, is crucial. Muscle paths are often idealised representations (e.g., lines or splines), tissue properties may be simplified, and joint constraints might not capture the full range of individual human variation or laxity. Yeo et al. (2023) raises numerical instability issues within the Hill-type muscle model. These necessary abstractions can influence the accuracy of simulated muscle forces and activation patterns.

The most significant challenge lies in the validation of these models. Directly measuring in vivo muscle forces or activations across the human body remains impractical. Researchers must often rely on indirect validation methods, such as comparing simulated activations with surface Electromyography (EMG) data (which has its limitations in spatial resolution and crosstalk as described by Farina et al. (2004)) or assessing kinematic consistency. This makes definitively quantifying the ground-truth accuracy of simulated biomechanics extremely difficult.

Finally, the performance of these detailed simulations is susceptible to the quality of input motion data. Driving complex musculoskeletal models with potentially noisy, sparse, or inaccurate joint trajectories derived from sources like monocular video pose estimation introduces a significant risk of generating physically implausible or unstable simulation results, a key challenge addressed later in this research.

2.5.3 Simulation Environments and Engines

Modern biomechanical research leverages a suite of complementary simulation environments and engines, each presenting distinct trade-offs between visualisation capabilities, simulation fidelity, performance, and ease of use. Selecting the appropriate toolchain requires careful consideration of the specific research goals and constraints.

For visualisation, tools like AITViewer created by Kaufmann et al. (2022) offer a native Python interface and interactive GUI for rapidly loading, editing, and playing back 3D mesh sequences (SMPL, SMPL-X, etc.). While excellent for visualising motion data or pose estimation outputs at high frame rates and integrating with image-based pipelines, AITViewer lacks built-in physics simulation capabilities.

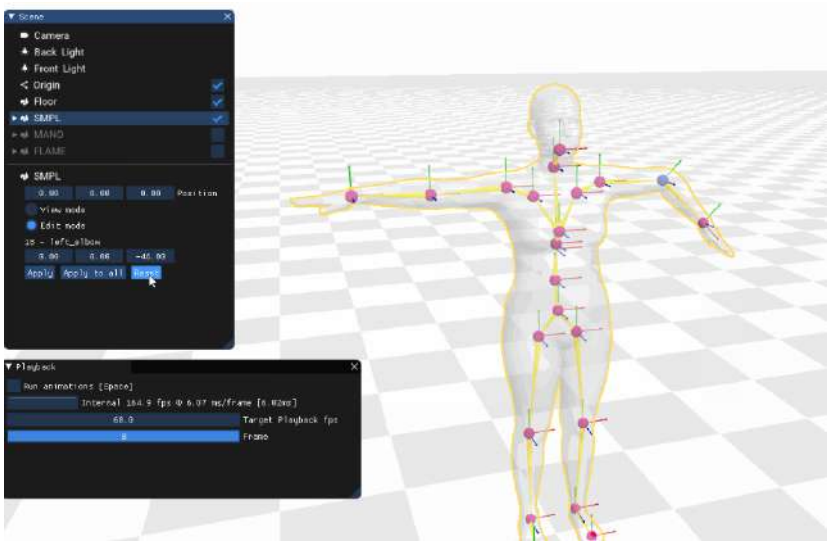

Figure 2.13: Aitviewer Kaufmann et al. (2022)

At the core of detailed musculoskeletal modelling lies OpenSim from Delp et al. (2007), a free, extensible, and widely adopted toolkit. It enables the construction of complex models and supports both inverse and forward dynamics simulations, backed by a vast library of published models and a large user community. However, OpenSim is known for its significant computational demands, often hindering real-time applications, and possesses a steep learning curve that can be a barrier for new users.

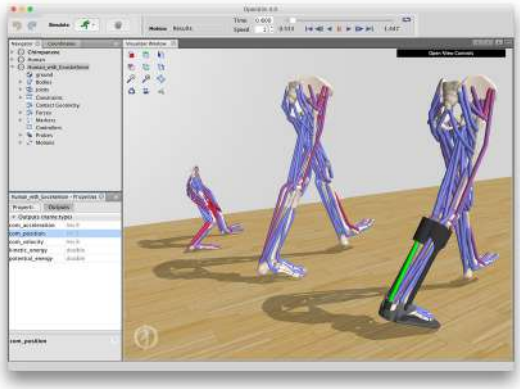

Figure 2.14: Opensim (Delp et al., 2007)

For applications demanding high-speed simulation, particularly those involving frequent contact dynamics common in robotics and potentially biomechanics, MuJoCo ("Multi-Joint dynamics with Contact") introduced by Todorov et al. (2012) provides a specialised physics engine. It is designed for fast contact resolution using convex optimisation and offers high accuracy in generalised coordinates. While powerful, adapting MuJoCo for detailed, subject-specific musculoskeletal models standard in biomechanics may require considerable expertise compared to frameworks explicitly designed for that purpose.

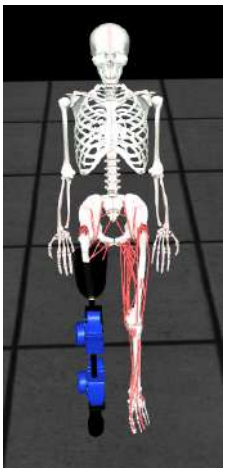

Figure 2.15: Mujoco Todorov et al. (2012)

Attempting to bridge the gap between OpenSim's biomechanical detail and real-time performance requirements, Hyfydy from Geijtenbeek (2021) implements similar muscle and contact models but uses optimised C++ code and adaptive integration to achieve significant speedups (reportedly 100x). This makes it promising for large-scale predictive simulations and reinforcement learning. However, ensuring consistent validation against established OpenSim benchmarks across diverse models and maintaining broad compatibility remain important considerations for such accelerated platforms.

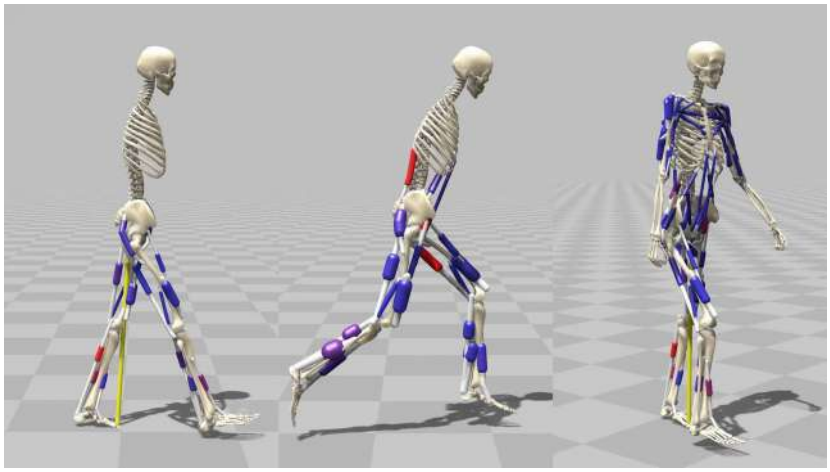

Figure 2.16: Hyfydy Geijtenbeek (2021)

To manage complex simulation workflows, SCONE (Software for Control aNd Estimation)from the work of Geijtenbeek (2019) acts as an orchestration layer. It allows researchers to define scenarios bundling models, controllers, optimisation objectives, and solver settings to automate gait or balance optimisation tasks. While powerful for predictive simulations, SCONE introduces another level of abstraction and its specific interface and scripting language that users must learn.

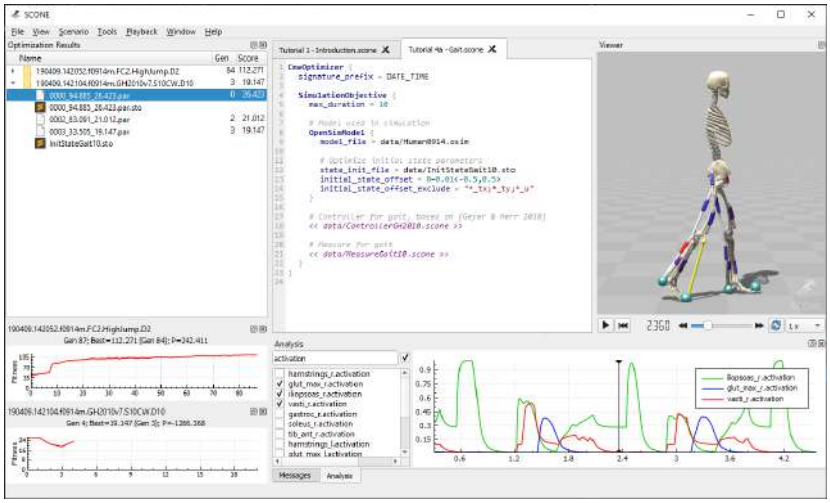

Figure 2.17: Scone Geijtenbeek (2019)

A significant practical challenge is the lack of seamless interoperability between these environments. Transferring models, controller parameters, or simula-

tion results between platforms like OpenSim (Delp et al., 2007), MuJoCoTodorov et al. (2012), and visualisation tools frequently requires manual conversion steps, custom scripting, and careful validation due to differing file formats and internal representations.

Ultimately, for the specific context of this thesis, aiming for real-time movement feedback driven by visual input, the choice of simulation engine is critical. The inherent latency in standard OpenSim or the complexity of adapting MuJoCo suggests that achieving real-time performance requires exploring highly optimised engines like Hyfydy, developing significantly simplified biomechanical models, or potentially focusing on post-processing analysis rather than live simulation feedback. Visualisation tools like AITViewer remain highly relevant for displaying the outputs, regardless of the chosen simulation approach.

2.6 Datasets and Evaluation Metrics

The evaluation of modern 3D pose estimation and biomechanical analysis methods relies heavily on benchmark datasets and standardised metrics. While essential for tracking progress and comparing approaches, it is crucial to critically assess the limitations inherent in both the available data and the metrics commonly employed.

3DPW created by von Marcard et al. (2018), provides valuable 'in-the-wild' video sequences captured with moving cameras and synchronised IMU data, offering challenging, realistic scenarios. However, its reliance on IMUs limits direct comparison with vision-only methods. While useful for kinematic evaluation in natural settings, it lacks ground truth for kinetic or muscle activity analysis.

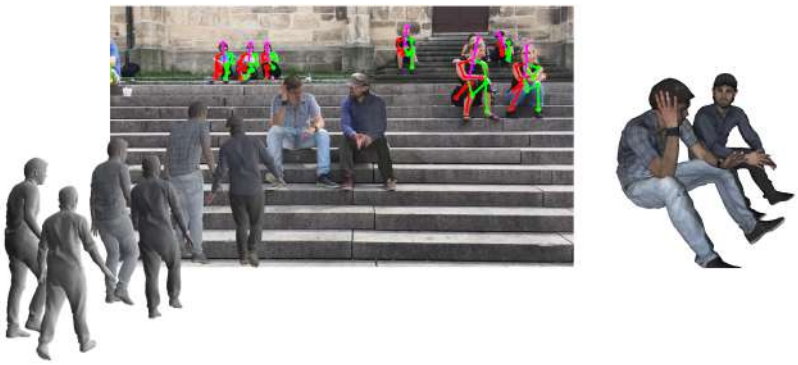

Figure 2.18: 3DPW von Marcard et al. (2018)

Human3.6M from Ionescu et al. (2014) remains a de facto standard in this area, comprising millions of frames of indoor motion-capture data with precise 3D joint annotations. This has standardised metrics like Mean Per Joint Position Error (MPJPE) and its Procrustes-aligned variant (P-MPJPE). Critically, however, Human3.6 M's controlled lab environment, limited actor diversity, and constrained action set may not fully reflect real-world complexity. Furthermore, MPJPE primarily measures geometric accuracy and may not adequately capture the physical plausibility, temporal smoothness, or dynamic correctness crucial for biomechanical simulation.

AMASS, a contribution from Mahmood et al. (2019), aggregates numerous motion capture datasets into a standard SMPL-based format, enabling large-scale training and evaluation. While offering unprecedented scale, this aggregation process inherently combines data from varied sources with different capture technologies and protocols, potentially introducing inconsistencies or biases.

Huang et al. (2022) proposes RICH (Real scenes, Interaction, Contact and Humans), which focuses specifically on human-scene interaction, providing multiview video, markerless motion capture, 3D scans, and importantly, dense contact labels. This is vital for benchmarking contact-aware models, though the complexity of the capture setup may limit dataset scale and diversity.

MOYO (The Mocap Yoga Dataset), as outlined by Tripathi et al. (2023), presents high-resolution multiview data emphasising challenging poses and contact scenarios, uniquely including ground truth SMPL-X fits, pressure mat readings, and body Centre-of-Mass (CoM) measurements. This kinetic data is highly valuable for validating the physical realism required in biomechanical simulation, but like RICH, the intricate capture setup impacts scalability.

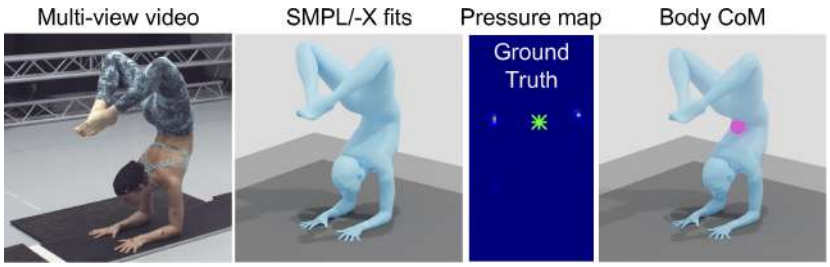

Figure 2.19: MOYO Tripathi et al. (2023)

Collectively, these datasets enable evaluation using metrics like MPJPE/P-MPJPE, acceleration error (which can be sensitive to noise in estimated poses), and contact detection rates. However, these metrics primarily assess kinematic fidelity or simple contact events. They fall short of providing direct evaluation for core biomechanical variables like muscle activation levels, joint torques, or metabolic energy.

2.7 Research Gaps

While substantial advances have been made in both 3D pose estimation from monocular video and musculoskeletal simulation independently, there remains a notable lack of real-time integration between these two domains. Most existing biomechanical simulation frameworks, such as OpenSim (Delp et al., 2007), rely on motion capture data or offline-optimised trajectories, which are unsuitable for live feedback applications. Conversely, real-time pose estimation models like SPIN (SMPL oPtimization IN the loop) (Kolotouros et al. (2019)) can deliver 3D

joint predictions at interactive rates but lack the biomechanical detail required for accurate muscle activation or force analysis. Bridging these systems in real time poses significant challenges: minor errors in pose estimation can accumulate into physically implausible joint torques or muscle activations, and the computational complexity of detailed simulations often exceeds the latency thresholds necessary for interactive use. As a result, there is a critical research gap in developing pipelines that can both infer accurate, physically plausible biomechanical states and operate at real-time speeds from raw monocular input, a gap this project seeks to address.

3

Methodology

3.1 Research Design

Building on the gap identified in the previous section, the absence of a pipelined, real-time system that directly regresses SMPL body parameters from monocular video and showcases this real-time body for intuitive feedback, this project adopts an experimental approach. An end-to-end pipeline was constructed and evaluated that captures live video, estimates 3D human pose through SMPL parameters, and immediately presents this captured pose for live analysis. By treating each module (pose regression, parameter conversion, simulation) as an integrated whole, the stand-alone performance of the SMPL regressor and its impact on downstream biomechanical plausibility under real-time constraints was measured.

3.1.1 Objectives and Hypotheses

The primary objectives of this research are:
1. **Real-Time SMPL Regression:** Achieve monocular SMPL parameter inference at $\geq$20 FPS with latency under 100 ms per frame.

We test the following hypotheses:

H1: A lightweight deep regressor (e.g., Ipman (Tripathi et al., 2023) fine-tuned on SMPL outputs) can deliver $\geq$20 FPS inference on commodity GPU

hardware.

H2: The fully integrated pipeline can sustain interactive performance (end-to-end latency ≤100 ms) without catastrophic errors in pose or bodywhe simulation.

H3: The integrated system can achieve a pose estimation accuracy targeting ≤25 mm MPJPE) on real-time frame-to-frame pose comparisons.

H4: The system will demonstrate robustness by maintaining performance (FPS drop ≤10%), accuracy (MPJPE increase ≤5%), and simulation stability (simulated joint torques variation ≤15%) under conditions of occlusion, varied lighting, and rapid motion.

3.1.2 Evaluation Criteria

To validate these hypotheses, we define the following criteria:
- **Inference Throughput:** Measured in frames per second (FPS) for SMPL regressor alone and the combined pipeline (pose → simulation).
- **Pose Accuracy:** MPJPE and Procrustes-aligned MPJPE (P-MPJPE) on real-time frame to frame pose comparison, targeting ≤25 mm.
- **End-to-End Latency:** Total elapsed time from frame acquisition to visualisation update, with an upper bound of 100 ms to ensure interactive responsiveness.
- **Robustness Tests:** Performance under occlusion (simulated by masking input regions), varied lighting, and rapid motion—benchmarked by drop in FPS (≤10 %), increase in MPJPE (≤5 %), and stability of simulated joint torques (≤15 %).

For my project, an experimental methodology was used to develop and validate the simulation's ability to mimic human input movements and to see how these movements mapped to different muscle activation levels. Experiments will be conducted to test the accuracy and robustness of pose estimation under varying conditions, including different camera angles, lighting, and subject variability.

3.2 System Architecture

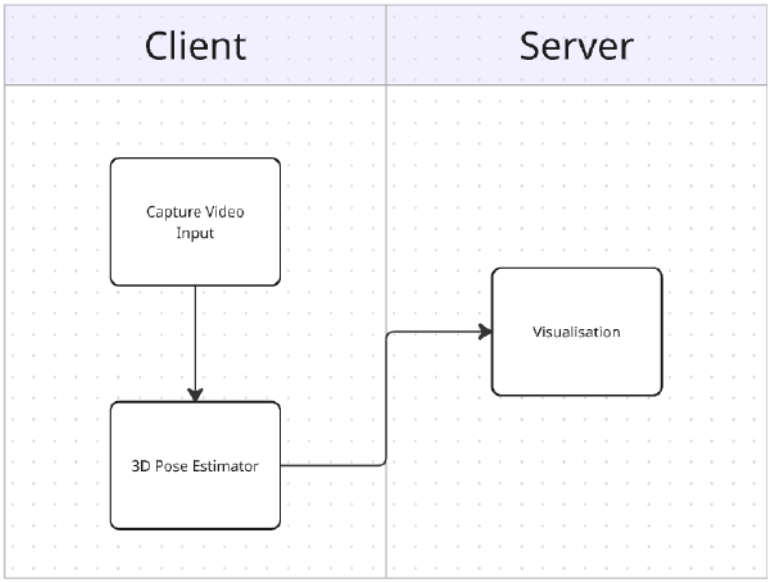

Figure 3.1: Overview Diagram of System Architecture

The three subsystem components illustrated above are:

- Input Video
- 3D Pose Estimator
- Visualisation

Each plays a critical role in the function and performance of the system as a whole and can be described as follows:

Input Video Monocular video capture is an essential part of the research question. This project used a standard consumer-level webcam, specifically the Ugreen 2k 30fps webcam. This is important as it proves the consumer-ready nature of the project's area of development. OpenCV (Bradski, 2000) is used to read the captured frames of the camera, which will be used for the subsequent downstream tasks. Accurate frame capture is vital, as the concept of garbage in and garbage out is a key consideration, as all tasks build upon this frame capture.

3D Pose Estimator With this project attempting to address the research question of real-time 3D pose estimation and its effectiveness in physiotherapy and sports science, accuracy vs inference speed is a key consideration. For that reason, the Ipman model was used as it provided adequate inference speeds while providing a high level of accuracy and confidence scores for its prediction as will be shown in the results and findings. This model places a focus on tough positions and is trained on the MOYO (Tripathi et al., 2023) dataset which consists of Various yoga positions which makes it ideal for this paper with the focus of physiotherapy and sports science. As will be discussed further in Limitations of this project, the available GPU had memory and computational power constraints. However, the Ipman model, with its efficient architecture, could fit within these limitations, allowing for successful execution and evaluation on the available hardware.

Visualisation While the 3D pose estimator can overlay its prediction over the captured frame data, this loses the benefits of a 3D body model. Using a Simulation Environment, various benefits for the end user are discovered, namely, exploring the pose of the body that may not be facing the camera, the ability to focus on particular sections, and the re-playability within the environment. As a SMPL (Loper et al., 2015) body pose was used as the result of the predictions, aitviewer (Kaufmann et al., 2022) was an obvious choice due to its design for this family of parametric body models. Aitviewer was utilised in its server mode, which means that a client-server approach can be utilised where the 3D Pose Estimator can run as a client and upload its predictions to the aitviewer server. This allows for efficient, fast and seamless data transfer using the PICKLE format for serialisation.

3.3 Technology Stack

3.3.1 Software (languages, frameworks, libraries)

Through the course of this research project, the following tools and technologies will be utilised.

Linux 24.04 and Python 3.7 were used for this project, and a full requirements.txt is included with the code.

Input Video

- OpenCV (Bradski, 2000)

3D Pose Estimator

- Ipman (Tripathi et al., 2023)
- Pytorch (Pytorch, n.d.)

Visualisation

- Aitviewer (Kaufmann et al., 2022)

3.3.2 Hardware and Infrastructure

As this is an AI Inference project, GPU is the key factor in system performance; the hardware that was used is the following:

PC

- AMD Ryzen 5800X
- GeForce RTX 2070 8GB VRAM
- 64GB DDR4 Memory

Peripherals

- UGREEN 2k 30fps Webcam

3.4 Implementation

The core implementation of the created systems resides in three main scripts: Inference.py, Relay.py, and AIT_server.py.

The relay file was used to solve dependency issues between the *Anaconda Software Distribution* (2020) environments of the packages necessary for aitviewer and Ipman.

3.4.1 Ipman Predictions

The Ipman model was selected for its demonstrated balance of accuracy and inference speed, alongside its ability to provide prediction confidence scores and operate within the available hardware constraints (Nvidia RTX 2070 8GB), as discussed in Section 3.3. In contrast to POCO which first performs 2D pose estimation and then crops the image to isolate the detection and performs the 3D pose estiamtion based on this detection. Ipman performs a holistic approach of generating the predicted 3D pose from the rgb frame

After cropping the captured image to 224 x 224 as required by the Ipman model, This processed image is fed into the model to generate a prediction. torch.no_grad() is used here to tell PyTorch (Pytorch, n.d.) not to calculate gradients during the forward pass, which means there is reduced memory usage and faster computation. These benefits are essential for real-time inference.:

```
1    with torch.no_grad():
2        # pred_rotmat: (B, 24, 3, 3) - Rotation matrices
3        # pred_betas: (B, 10) - Shape parameters
4        # pred_camera: (B, 3) - Weak perspective camera [s, tx, ty]
5        pred_rotmat, pred_betas, pred_camera = model(norm_img)
```

From this output, the predicted pose, shape, and camera parameters can be extracted after some post-processing to clean up and prepare the pose for the viewer. This pose is serialised and sent over a websocket, where Relay.py receives it.

3.4.2 Aitviewer

AitViewer (Kaufmann et al., 2022) provides library code that enables the creation of a RemoteSMPLSequence. Relay.py uses this first to create a RemoteSMPLSequence, wait for the connection from Inference.py, unpickle the received serialised code, validate its data, and then update the fame of the RemoteSMPLSequence using this pose. This flow will repeat for as long as there is a connection to the aitviewer server and the Inference client connection. Relay.py acts as both a client and a server in this instance.

```
smpl_sequence = RemoteSMPLSequence(
    v,
    poses_body=initial_pose_body,
    poses_root=initial_pose_root,
    betas=initial_betas,
    trans=initial_trans,
    gender='neutral',
    model_type='smpl',
    name="Relayed SMPL",
    color=(0.8, 1.0, 0.6, 1.0),
)
```

The aitviewer server consisted of the standard viewer code, with the addition of configuring the server_enabled set to True, this enables the client server aspect of this process to take place.

```
if __name__ == "__main__":
    print("Starting Custom Viewer Server...")
    try:
        C.update_conf(
        {"server_enabled": True, "server_port": 8417}
        )
        v = ControllableViewer(
          title="SMPL Viewer (via Relay)"
        )
        v.scene.floor.enabled = True
```

```
        print(
          f"Viewer Server running on port {C.server_port}. Waiting for
          ↪  relay client (remote_relay.py)..."
        )
        v.run()
    except Exception as e:
         print(f"ERROR starting viewer server: {e}")
         traceback.print_exc()
    finally:
        print("Viewer server stopped.")
```

3.4.3 Data Pipelines

WebSockets, combined with pickle serialisation, were chosen for data transmission to enable efficient, low-latency streaming suitable for real-time updates between the pose estimation client and the Aitviewer server. This leverages Aitviewer's built-in remote capabilities, detailed further in Section 3.6.

3.5 Acceptance Criteria

To determine whether this research project has been successful, it is essential to clearly define acceptance criteria.

Table 3.1: System Tasks and Acceptance Criteria

Task	Criteria
Real-time Pose Estimation Performance	The end-to-end system shall estimate human pose at an average throughput of ≥20 Frames Per Second (FPS). Under defined optimal conditions (e.g., subject clearly visible within 1-5 meters, well-lit environment, minimal occlusion), the pose accuracy shall achieve a Mean Per Joint Position Error (MPJPE) of ≤25 mm.

Continued on next page

Table 3.1: System Tasks and Acceptance Criteria – Continued

Task	Criteria
Interactive System Latency	The total end-to-end latency, measured from frame acquisition by the camera to the corresponding visual update of the simulated body on screen, shall be $\leq$100 milliseconds to ensure a responsive interactive experience.
System Stability and Simulation Integrity	The fully integrated pipeline shall operate continuously for a defined period (e.g., at least 5 minutes) without catastrophic errors, such as crashes, unrecoverable tracking failures, or physically implausible bodywhe simulation outputs (e.g., joint explosions, persistent severe jitter) under normal operating conditions.
Robustness to Visual Perturbations	When subjected to defined challenging conditions (e.g., partial occlusion simulated by masking up to 20% of input image regions, varied lighting, and rapid subject motion): • The system's FPS shall not degrade by more than 10% compared to performance under optimal conditions. • The pose estimation MPJPE shall not increase by more than 5% compared to performance under optimal conditions. • The stability of simulated joint torques shall be maintained, with variations not exceeding 15% from their baseline average during stable motion.
Pose Estimation Reliability (Qualitative)	Under optimal conditions, the system shall successfully recognise and track the main person's pose for at least 95% of the frames during a continuous 2 minute observation period when the person is within 10 meters and not significantly obstructed by other objects or people.

3.6 Evaluation Plan

To rigorously validate the real–time SMPL-driven (Loper et al., 2015) biomechanical pipeline, we adopt a three-pronged evaluation strategy encompassing quantitative metrics, qualitative case studies, and robustness/error analysis. This comprehensive plan ensures we assess not only raw accuracy and performance, but also our system's practical fidelity and resilience under varied conditions.

3.6.1 Quantitative Metrics

Pose accuracy is measured using mean per-joint position error (MPJPE) and Procrustes-aligned MPJPE (PA-MPJPE) on the standard Human3.6M (Ionescu et al., 2014) and 3DPW (von Marcard et al., 2018)test sets. These metrics quantify the Euclidean distance between predicted and ground-truth joints before and after rigid alignment. For the purposes of this paper an amdeded version of MPJPE will be used, where the previous frame will be treated as the ground-truth to create a measure of stability. This is not as accurate as ground-truth data but is effective for the comparisons required for this paper.**Performance** is evaluated by recording the average end-to-end latency (in milliseconds) from image capture to simulation output, and by computing frames-per-second (FPS) throughput for both the SMPL-regressor alone and the whole pipeline (pose→simulation→render). Finally, **simulation fidelity** was assessed by gathering qualitative data of the end-to-end pipeline across different scenarios, here features such as Observed Visual Smoothness, Observed Responsiveness, Observed Jitter/Artifacts, and Observed Intuitiveness of Feedback. These metrics will provide an evaluation of how usable the system from a users perspective of intuitive feedback.

3.6.2 Qualitative & Case Studies

To illustrate practical utility, case studies were conducted on three representative movements: a deep squat, a biceps curl, and a normal gait cycle. These formed the basis for the Qualitative feedback as it provided a wide range of movements that were pertinent to the area of research.

3.6.3 Error Analysis & Robustness

System resilience was probed by systematically introducing common perturbations:

- *Occlusion tests:* Randomly mask 10–30% of the image area to simulate partial body occlusion.
- *Lighting variation:* Reduce lighting in observation room to assess robustness under non-uniform illumination.
- *Fast motion:* Increase subject speed (e.g., Jumping Jacks) to evaluate SMPL-regressor latency and tracking stability.

For each scenario, changes in the modified MPJPE, FPS, and simulation fidelity metrics were recorded. In addition, ablation studies to isolate the contribution of key components were performed:

- *Smoothing filter:* Remove the Kalman-based jitter filter to measure its impact on pose-to-simulation stability.
- *SMPL variant:* Replace the SMPL-X model with standard SMPL to compare parameter expressiveness vs. computational cost.
- *Simulator backend:* Swap MASS (Lee et al., 2019) with the faster Hyfydy engine (Geijtenbeek, 2021) to quantify trade-offs between biomechanical accuracy and speed.

These analyses reveal the conditions under which our pipeline maintains performance and highlight areas for future optimisation.

3.7 Real-time Pose Visualization Subsystem (Ipman + aitviewer)

This section details the development of a subsystem focused on real-time 3D human pose and shape estimation from a live webcam feed and its dynamic visualisation using the Ipman (Tripathi et al., 2023) model and aitviewer framework (Kaufmann et al., 2022).

This subsystem warrants detailed description due to the inherent complexity of integrating real-time machine learning inference (Ipman), live video processing with OpenCV (Bradski, 2000), 3D graphics rendering (aitviewer), and robust client-server networking (WebSockets), and because its success is critical for the project's core aim of providing immediate visual feedback to the user.

3.7.1 Objective

The primary objective was to create a system capable of performing real-time 3D human pose and shape estimation from a live webcam feed and visualising the results dynamically in a 3D viewer. This involved integrating the Ipman monocular 3D human mesh recovery model with the aitviewer 3D visualisation framework.

3.7.2 Core Components

The subsystem utilised the following core components:

- **Ipman (Intuitive Physics-based HuMAN:** A machine learning model used for estimating 3D human mesh parameters (specifically SMPL (Loper et al., 2015) pose, shape, and translation) from a single RGB image. The pre-trained `impan-checkpoint.pt` checkpoint was utilised.
- **aitviewer:** A versatile, Python-based 3D visualisation tool designed for human motion and mesh sequence rendering. It provides functionalities for loading body models (SMPL, SKEL (Keller et al., 2023), STAR (Osman et al., 2020), etc.) and rendering sequences.
- **Webcam Interface:** Standard webcam input using OpenCV (`cv2`).
- **Person Detection/Tracking:** Utilised libraries like `multi_person_tracker` (MPT) for detecting bounding boxes of people in the webcam feed. Mkocabas (n.d.)

- **Communication Protocol:** Explored standard sockets with JSON, ultimately settling on WebSockets with Python's `pickle` serialisation for communication between the Ipman processing client and the aitviewer server, leveraging aitviewer's built-in server capabilities.
- **Programming Environment:** Python, PyTorch, NumPy, SciPy (for rotation conversions), OpenCV.

3.7.3 System Architecture and Workflow

A client-server architecture was implemented to decouple the potentially heavy Ipman inference from the 3D rendering. The client captures data and performs inference, sending the results over a standard socket connection to the server, which manages the aitviewer display.

Client (`demo.py`):

- Captures frames from a webcam using OpenCV.

```
1 cap = cv2.VideoCapture(args.cam_id)
```

Figure 3.2: Webcam Capture

- Performs a heuristic crop based on image size, simulating a single-person detection.

```
img_processed_np = crop(frame_rgb_float, center, scale,
↪  [args.img_res, args.img_res])
```

- Runs HMR (Ipman) inference on the cropped image to obtain SMPL parameters (rotation matrices, betas, camera).

```
pred_rotmat, pred_betas, pred_camera = model(norm_img)
```

- Converts the predicted rotation matrices to axis-angle format and calculates translation from camera parameters.

```
axis_angle_i = rotation_matrix_to_axis_angle(rotmat_i)
```

- Packages the extracted SMPL parameters (axis-angle poses, betas, translation) into a dictionary.

```
data_to_send = {
    "poses_body": poses_body_send,
    "poses_root": poses_root_send,
    "betas": betas_send,
    "trans": trans_send,
}
```

- Serializes the data dictionary using `pickle` (this happens inside the `send_data` function).
- Connects to the relay server via a standard TCP socket.

```
    sock.connect((RELAY_HOST, RELAY_PORT))
```

- Sends the serialised messages to the server using the custom `send_data` function.

```
    if not send_data(sock, data_to_send):
```

Figure 3.3: SMPL Model Overlay Over Webcam Capture

Server (`ait_server.py`):

- Initialises a `ControllableViewer` instance, which incorporates the aitviewer server functionality when enabled via configuration.

```
    v = ControllableViewer(title="SMPL Viewer (via Relay)")
```

- The viewer utilises a pre-configured SMPL layer for rendering.
- Intercepts incoming messages from connected clients within the overridden `ControllableViewer.process_message` method.
- Passes the received message data (type, remote UID, args, kwargs) to the default aitviewer server's message processing handler.

```
    self.server.process_message(type, remote_uid, args, kwargs,
    ↪  client)
```

- The default server handlers interpret the message type (`Message.SMPL`, `Message.ADD_FRAMES`, `Message.DELETE`) and interact with the `viewer.scene` accordingly (e.g., creating/updating `SMPLSequence` nodes or removing nodes).
- The `viewer.run()` method manages the main event loop, handling incoming connections, processing messages forwarded by `process_message`, and automatically rendering the scene contents.

```
    v.run()
```

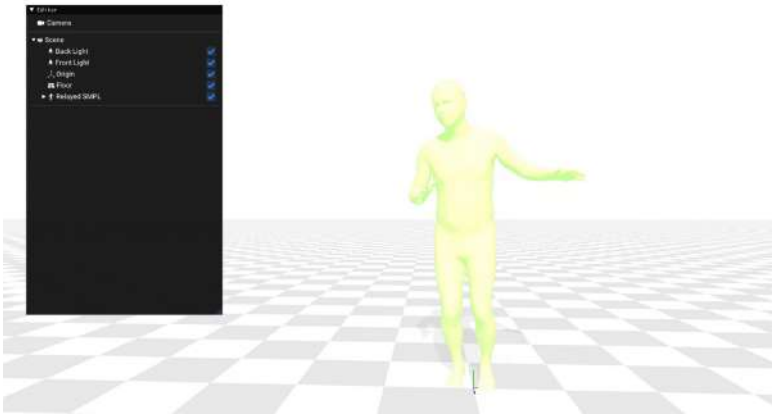

Figure 3.4: AMPL Body rendering in AitViewer

3.7.4 Implementation Details and Challenges

Key technical steps and challenges encountered during implementation included:

- **Adapting Static Processing for Real-Time Input:** The reference Ipman (Tripathi et al., 2023) demonstration script (`demo.py`) was designed for offline processing of static video files or image folders. Significant modification was required to enable continuous, frame-by-frame processing from

a live webcam feed, ensuring low-latency operation suitable for interactive use.

- **Client-Server Communication:** Initial attempts using standard sockets and JSON for communication proved less efficient for the high-frequency data transfer required. Therefore, the system was refactored to leverage `aitviewer's` (Kaufmann et al., 2022) built-in `ViewerServer` with `WebSockets` and `pickle` serialisation, offering better integration and performance for updating the remote renderables..

- **aitviewer Server Setup:** Configuring and running aitviewer in server mode, including managing dependencies and correctly setting model paths (e.g., `smplx_models` in `aitvconfig.yaml`). Debugging path issues was a significant step.

- **Ensuring Cross-Platform Data Integrity:** The transfer of complex data structures (pose parameters, shape parameters, translation vectors) between the client (performing inference) and the server (performing rendering) necessitated robust data handling. This included implementing reliable serialisation (`pickle`) and deserialisation routines, and critically, ensuring consistent data type and shape conversions between NumPy arrays, standard Python lists, and PyTorch (Pytorch, n.d.) tensors across both ends of the communication channel. Debugging runtime errors, such as attribute errors on list objects (e.g., 'list' object has no attribute 'shape'), highlighted the need to validate data structures upon receipt carefully.

- **Implementing Custom Message Handling Logic:** The aitviewer server required custom logic to interpret incoming messages formatted according to the `aitviewer.remote.message` protocol. Specific handlers were implemented on the server-side to process different message types (e.g., `Message.ADD_SMPL`, `Message.ADD_FRAMES`, `Message.DELETE`) and translate them into appropriate actions on the `SMPLSequence` objects within the aitviewer scene graph (e.g., adding new tracked persons, updating existing poses, removing persons who left the scene). To ensure these handlers were executed reliably within aitviewer's main event loop, it was necessary

to subclass the standard `Viewer` to intercept and process the message queue before each frame render.

- **Investigating Body Model Compatibility:** An exploratory phase involved attempting to utilise the SKEL (Keller et al., 2023) body model within aitviewer, driven by its potential for more biomechanically accurate skeletal representation. However, this revealed a fundamental incompatibility: Ipman outputs parameters for the SMPL model, which are structurally different from those required by SKEL. Bridging this gap would necessitate a complex, real-time SMPL-to-SKEL parameter conversion algorithm. Such a conversion was deemed computationally prohibitive for real-time application. Therefore, the pragmatic solution was adopted: matching the server-side model (SMPL) to the format of the data being generated by the client (Ipman's SMPL output)

3.7.5 Outcome

The final system successfully demonstrates real-time capture of webcam video, estimation of SMPL (Loper et al., 2015) parameters for detected persons using Ipman (Tripathi et al., 2023), transmission of these parameters over a network using WebSockets, and dynamic visualisation of the corresponding 3D human models within the aitviewer framework (Kaufmann et al., 2022). The functionality of this subsystem was verified through qualitative visual inspection, confirming that the 3D model rendered in aitviewer accurately mirrored the movements captured by the webcam feed in real-time during various test motions. This provides a flexible real-time platform for visualising monocular 3D human pose estimation results. Refer to Appendix A for an image of the outdoor testing of this full system.

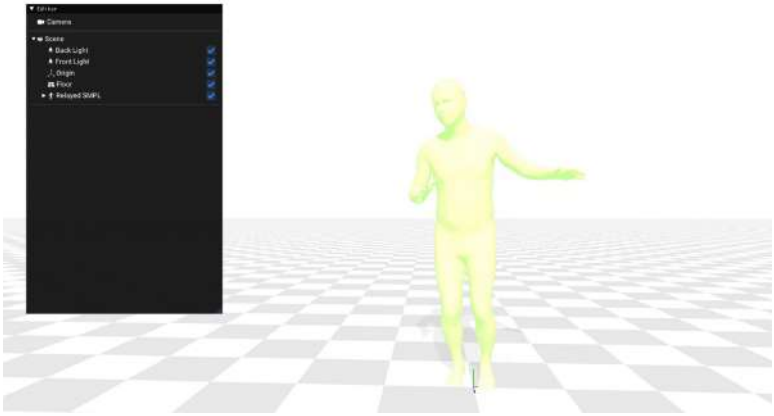

Figure 3.5: Full Outcome of System

3.8 Development Steps

There were three major milestones to this project.

- Real-Time Pose Estimation
- Human Body Simulation Visualisation
- Integrating these two modules

These three central components were crucial to the development of this project and can be realised in the following development plan.

1. Design and Development of the Video Input Module: To capture and process video from a standard monocular camera.
2. Design and Development of the 3D Pose and Shape Estimation Module: Implementing the integration of pose estimation models to derive 3D pose and shape parameters from the input video frames.
3. Design and Development of the 3D Visualisation Module: To render the 3D human mesh dynamically using the estimated parameters, providing real-time visual feedback.
4. Integration of Modules: Combining the video input, pose/shape estimation, and visualisation modules into a cohesive real-time pipeline.
5. System Testing and Evaluation: Assessing the performance, accuracy, and intuitiveness of the feedback provided by the complete system.

Following these development steps facilitates the success of this research project and keeps it in line with its proposed requirements.

4

Results and Findings

4.1 Quantitative Metrics of Pose Estimation

While each model mentioned in this paper has its own results when compared to datasets such as Moyo (Tripathi et al., 2023), 3DPW (von Marcard et al., 2018), and Human 3.6M (Ionescu et al., 2014). While these metrics are shown below, the primary results and findings of this paper will be an analysis of frame-to-frame pose changes, and filtering for the purposes of the proper implementation of an effective real time 3D pose estimation.

The below table shows a comparison of two models POCO (Dwivedi et al., 2023) and Ipman-R (Tripathi et al., 2023), other models such as WHAM (Shin et al., 2024), HSMR Yan et al. (2025) and CameraHMR (Patel and Black, 2025) were initially trailed but did not perform on the system requirements, this is explored further in the limitations sections of the thesis. A testing suite was built to measure the following metrics for the pose estimation process.

- Processing FPS (FPS)
- End-to-End Latency (ms)
- Δ (change in) Pose Change (Euclidean Dist.)
- Δ (change in) Translation Change (mm)
- Δ (change in) Joint Position Change (mm)

- Shape Parameter Stability (Avg Variance)
- Detection/Tracking Rate (%)

Figure 4.1: Pose Testing Platform

While testing the real-time performance on ground truth data from datasets was not possible for this project, a frame-by-frame comparison process was used. By performing the process on two subsequent frames of identical poses, it can be determined that any change in detection is due to errors brought on by the real time 3D pose estimation process. Each of these metrics were calculated over subsequent frames over a period of 5 seconds across different conditions. These conditions provide a thorough testing for an 'in the wild' scenario of these models. The conditions were as follows:

- Baseline (Optimal Conditions)
- Low Light
- Partial Occlusion
- Fast Motion

- Static + Moving Occlusion

Under the listed scenarios and conditions the following metrics were generated:

Table 4.1: POCO - Comparison of Pose Performance Metrics (FPS: Frames Per Second; Δ: Change).

Metric	Baseline	Low Light	Partial Occlusion	Fast Motion	Moving Occlusion
Processing FPS (FPS)	4.0013	4.0023	3.9876	3.9896	3.9394
End-to-End Latency (ms)	249.9939	249.9019	250.8645	250.7335	254.1032
Δ Pose Change (Euclidean Dist.)	0.2635	0.5819	0.3862	5.8667	0.916 213 7
Δ Translation Change (mm)	2.9649	11.3687	5.3096	121.2970	13.0980
Δ Joint Position Change (mm)	4.2926	11.1660	6.7366	173.9608	16.431 961
Shape Parameter Stability (Avg Variance)	0.4629	0.3747	0.2280	0.4301	0.536 963 34
Detection/Tracking Rate (%)	100	100	100	100	100

Table 4.1 shows the system exhibits significantly reduced performance across all challenging conditions, with processing FPS dropping to around 4 and end-to-end latency increasing substantially to approximately 250 ms. Additionally, the Δ pose, translation, and joint position changes are markedly higher, indicating instability in output pose estimations. These results highlight the model's current limitations in handling dynamic or sub-optimal environments, where both accuracy and responsiveness degrade sharply.

To address these issues, TensorRT was integrated to accelerate model inference and improve real-time performance. As shown in the updated results below, this optimization led to a substantial increase in processing FPS across all scenarios, with the system achieving over 28 FPS under optimal conditions and maintaining high frame rates even in low light and fast motion environments. End-to-end latency was significantly reduced, remaining below 50 ms in most cases. While

a slight trade-off in accuracy was observed—particularly in pose and joint position change, the overall detection and tracking rate remained consistently high. These improvements highlight the effectiveness of TensorRT in enhancing runtime efficiency, making the system more suitable for real-time applications while maintaining acceptable accuracy levels.

Table 4.2: POCO with TensorRT - Comparison of Pose Performance Metrics (FPS: Frames Per Second; Δ: Change).

Metric	**Baseline**	**Low Light**	**Partial Occlusion**	**Fast Motion**	**Moving Occlusion**
Processing FPS (FPS)	28.1308	20.4117	27.5562	26.4461	27.7
End-to-End Latency (ms)	37.9094	49.9615	39.4092	41.5531	39.2012
Δ Pose Change (Euclidean Dist.)	0.2252	0.3662	0.4428	1.4723	0.662 048
Δ Translation Change (mm)	4.7456	10.5170	14.7675	36.4185	13.5516
Δ Joint Position Change (mm)	8.8177	17.0829	16.3127	99.5186	32.619 675
Shape Parameter Stability (Avg Variance)	0.3365	0.6463	0.4576	0.5299	0.502 798 5
Detection/Tracking Rate (%)	100	100	98.4127	100	98.5507

Building on the performance gains achieved through TensorRT optimisation, a new pose estimation model; Ipman, was evaluated to further enhance real-time tracking capabilities and accuracy. The results in Table 4.3 showcase Ipman's performance across a similar range of challenging conditions.

Table 4.3: Ipman - Comparison of Pose Performance Metrics (FPS: Frames Per Second; Δ: Change).

Metric	Baseline	Low Light	Partial Occlusion	Fast Motion	Moving Occlusion
Processing FPS (FPS)	30.8497	14.5088	30.8156	30.5509	30.7942
End-to-End Latency (ms)	32.4830	68.9419	32.4693	33.1876	32.4903
Δ Pose Change (Euclidean Dist.)	0.1752	0.3902	0.1921	1.6717	0.4725
Δ Translation Change (mm)	53.8830	200.7227	86.3862	585.1612	153.0838
Δ Joint Position Change (mm)	3.6429	7.4517	4.1693	38.0599	9.4923
Shape Parameter Stability (Avg Variance)	0.5846	0.4534	0.3755	0.6350	0.6351
Detection/Tracking Rate (%)	100.0000	100.0000	100.0000	100.0000	100.0000

Table 4.3 presents the quantitative performance metrics for the Ipman system across various challenging scenarios compared to a baseline. Under baseline conditions, the system achieves high processing speed (∼30.85 FPS) and low end-to-end latency (∼32.48 ms), indicating efficient operation. Frame-to-frame changes in pose, translation, and joint position are minimal, reflecting stable tracking. The most notable observation is the system's robustness in maintaining subject tracking, as evidenced by the 100.0000% Detection/Tracking Rate across all tested challenging conditions. While the system maintains high processing speed and low latency in Partial Occlusion, Fast Motion, and Moving Occlusion scenarios (comparable to baseline), performance is significantly impacted under Low Light conditions, where Processing FPS drops to ∼14.51 and End-to-End Latency increases to ∼68.94 ms.

It should be noted that for the Fast Motion Scenario only; Processing FPS, End-to-End Latency, and Detection Rate can be compared against the other scenarios, this is due to the nature of the other metrics relying on delta changes in the pose and shape parameters. This is not possible in this scenario as it

inherently has motion involved.

Furthermore, the challenging conditions as descried in Table 4.3 generally led to increased variability in pose, translation, and joint positions between frames, with Low Light and Moving Occlusion resulting in the largest increase across these frame-to-frame change metrics. Shape parameter stability shows less dramatic variation across conditions. Overall, the data demonstrates Ipman's capability to reliably detect and track the subject even under severe challenges, although temporal and spatial accuracy can be affected depending on the specific nature of the difficulty.

Comparison of POCO and Ipman Comparing POCO (Table 4.1) and Ipman (Table 4.3) for real-time pose estimation, Ipman consistently outperforms POCO in temporal performance. Under baseline, Ipman achieves significantly higher FPS (∼30.85 vs ∼14.80) and lower Latency (∼32.48 ms vs ∼62.35 ms). This advantage holds across challenging conditions like Partial Occlusion, Fast Motion, and Moving Occlusion, where Ipman maintains high speed and low latency, while POCO operates at generally lower rates. Although both systems achieve a perfect 100% Detection Rate, Ipman's superior speed and responsiveness make it better suited for applications requiring robust real-time inference.

4.2 Quantitative Metrics of Real-time Filtering

While the underlying models achieve high accuracy, jitter introduced in the real-time viewer can impair intuitive movement feedback, potentially obscuring small movements within the noise. Consequently, a filtering step is introduced to mitigate this noise in the raw pose and shape estimations. A test suite was developed to quantify the performance of various filtering approaches by measuring the following key metrics, using unfiltered (raw) values as a baseline where applicable:

- Avg Relay Packet Processing (FPS)
- Avg Filter Latency (ms)
- Avg Filtered Pose Body Change

- Avg Filtered Pose Root Change
- Avg Filtered Trans Change (mm)

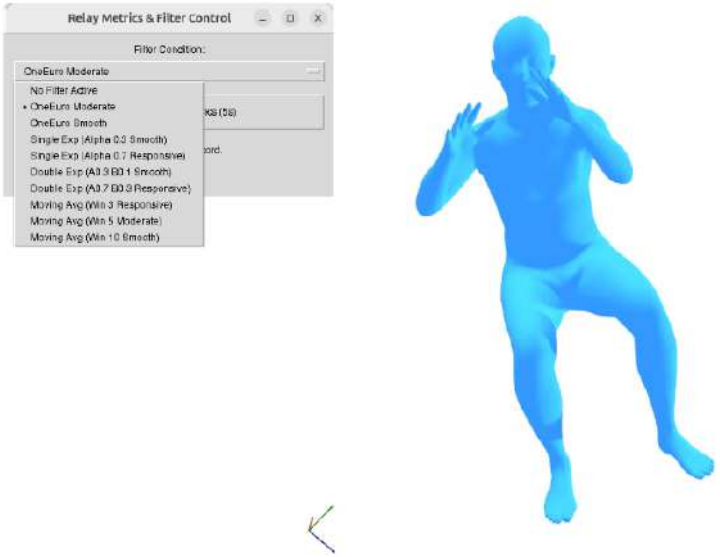

Figure 4.2: Filter Testing Platform

These metrics, analogous to model evaluation metrics under static conditions, quantify frame-to-frame changes while the subject remains stationary. Therefore, any detected changes in pose or translation are attributable to prediction errors or noise, rather than actual subject movement. Each metric is calculated between two subsequent frames and then averaged over a 5-second interval to provide a representative assessment of each algorithm's performance.
A diverse set of filtering algorithms was evaluated, typically with 'responsive' and 'smooth' variants for each. The 'responsive' variants aimed to maintain greater reactivity to input changes while still attenuating significant noise spikes, whereas the 'smooth' variants prioritized more substantial noise reduction to yield a highly consistent output. The selection of algorithms, detailed below, was guided by their prevalence and established use in related domains:

- OneEuro Moderate

- OneEuro Smooth
- Single Exp (Alpha 0.3 Smooth)
- Single Exp (Alpha 0.7 Responsive)
- Double Exp (A0.3 B0.1 Smooth)
- Double Exp (A0.7 B0.3 Responsive)
- Moving Avg (Win 3 Responsive)
- Moving Avg (Win 5 Moderate)
- Moving Avg (Win 10 Smooth)

Table 4.4: Filtered Comparison of Filter Performance Metrics (FPS: Frames Per Second; Δ: Change).

Filter Setting	**Relay FPS**	**Latency (ms)**	**Filt. Pose Body Δ**	**Filt. Pose Root Δ**	**Filt. Trans. Δ (mm)**
No Filter Active	29.849	0.0161	0.1436	1.1158	13.368
OneEuro Moderate	29.7366	0.1196	0.0201	0.3668	1.6143
OneEuro Smooth	29.6261	0.1174	0.0079	0.096	0.8152
Single Exp (Alpha 0.3 Smooth)	29.8365	0.0504	0.0294	0.2659	2.8281
Single Exp (Alpha 0.7 Responsive)	29.7354	0.0496	0.0755	0.5731	7.7655
Double Exp (A0.3 B0.1 Smooth)	29.7173	0.0669	0.0283	0.2228	2.8333
Double Exp (A0.7 B0.3 Responsive)	29.736	0.0736	0.0742	0.6836	8.1298
Moving Avg (Win 3 Responsive)	29.7499	0.1312	0.0423	0.3763	4.0757
Moving Avg (Win 5 Moderate)	29.7326	0.1337	0.0517	0.2126	3.6485
Moving Avg (Win 10 Smooth)	29.7839	0.1436	0.0125	0.109	1.2307

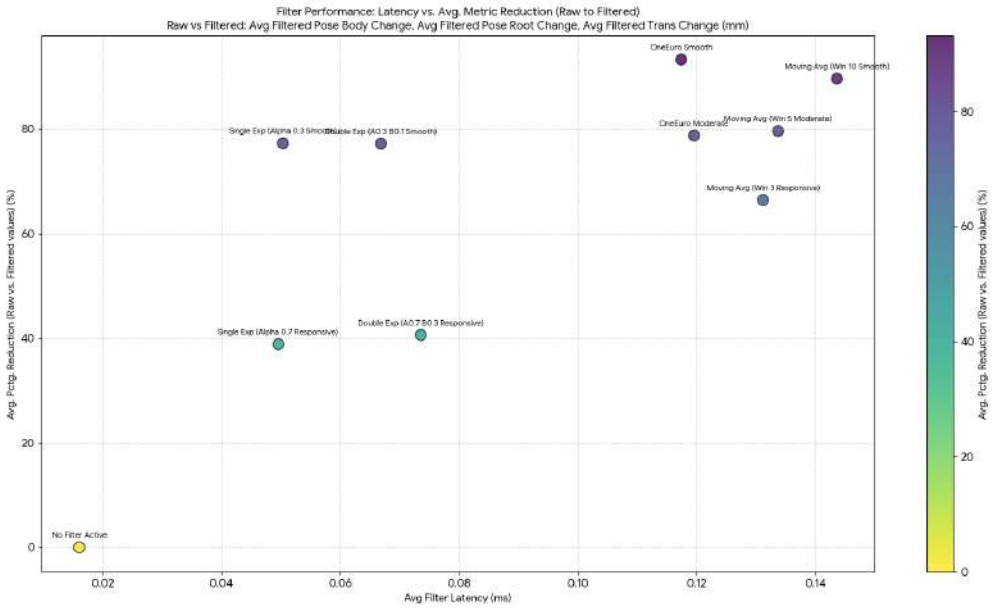

Figure 4.3: Filter Performance: Latency vs Avg. Metric Reduction

The quantitative metrics gathered from the test suite, presented in Table 4.4, offer a clear comparison of the efficacy of various filtering algorithms in mitigating jitter from the raw pose and shape estimations. The primary objective was to significantly reduce frame-to-frame changes in a static model—indicative of prediction noise—across pose body, pose root, and translational movements, while maintaining acceptable real-time performance in terms of processing throughput (FPS) and filter-induced latency.

Performance Metrics: Throughput and Latency All evaluated filtering algorithms demonstrated a marginal impact on the average relay packet processing throughput, with FPS values remaining high and generally comparable to the 'No Filter Active' baseline (29.849 FPS). For instance, the OneEuro Smooth filter recorded an FPS of 29.6261, indicating a negligible reduction in processing speed.

However, the introduction of filtering inherently adds a degree of latency. The 'No Filter Active' scenario exhibited minimal latency (0.0161 ms). In contrast,

the OneEuro Smooth filter introduced a latency of 0.1174 ms. While this is an increase, it remains a very small absolute value. Comparatively, other 'smooth' algorithms like Single Exponential (Alpha 0.3 Smooth) and Double Exponential (A0.3 B0.1 Smooth) demonstrated lower latencies (0.0504 ms and 0.0669 ms, respectively). Conversely, the Moving Average (Win 10 Smooth) filter showed a slightly higher latency (0.1436 ms) than OneEuro Smooth. This positions OneEuro Smooth in the mid-range for latency among the 'smooth' class of filters tested.

Noise Reduction: Pose and Translational Stability The paramount concern of jitter reduction is addressed by examining the average filtered changes in pose body, pose root, and translation, where lower values signify greater stability. When compared to the "No Filter Active" baseline (Pose Body Δ: 0.1436; Pose Root Δ: 1.1158; Trans. Δ: 13.368 mm), the OneEuro Smooth algorithm achieved exceptionally significant reductions in noise across all these metrics:

- Average Filtered Pose Body Change (Δ): Reduced to **0.0079**, an approximate 94.5% reduction from the unfiltered state.
- Average Filtered Pose Root Change (Δ): Reduced to **0.096**, an approximate 91.4% reduction.
- Average Filtered Translational Change (Δ): Reduced to **0.8152 mm**, an approximate 93.9% reduction.

These figures demonstrate a profound stabilisation effect. Furthermore, when compared against other 'smooth' profile algorithms, OneEuro Smooth consistently outperformed or was among the top performers in noise cancellation. For instance, its Pose Body Δ of 0.0079 is considerably lower than that of Single Exponential Smooth (0.0294) and Double Exponential Smooth (0.0283). Similarly, for Pose Root Δ and Translational Δ, OneEuro Smooth (0.096 and 0.8152 mm, respectively) yielded superior or highly competitive results compared to Moving Average (Win 10 Smooth) (0.109 and 1.2307 mm), and markedly better than the exponential smoothers.

Justification for OneEuro Smooth Based on these quantitative results, the OneEuro Smooth filter is identified as the most suitable method for the stated goal of providing intuitive movement feedback by minimizing jitter. While it introduces a modest latency of 0.1174 ms, the substantial gains in pose and translational stability are paramount. The algorithm's ability to reduce high-frequency noise, evidenced by the lowest frame-to-frame change metrics (Pose Body Δ, Pose Root Δ, and Trans. Δ), is crucial for ensuring that small, intentional movements are not obscured by prediction jitter.
While other algorithms, such as the Single and Double Exponential smoothers, offer lower latency, they do not achieve the same level of noise suppression as OneEuro Smooth. The Moving Average (Win 10 Smooth) filter, while effective, imposes a slightly higher latency and does not surpass OneEuro Smooth in overall stability. Therefore, the trade-off of slightly increased latency for significantly enhanced smoothness is deemed highly beneficial for the application's objective of clear and intuitive visual feedback.
The selection of OneEuro Smooth aligns with the requirement to effectively reduce noise from raw pose and shape, thereby enhancing the user's ability to perceive and respond to subtle aspects of their movement as reflected by the 3D mesh. The qualitative visualisation results in the Section 4.3 will discuss the effects of these models on small and fast movements.

4.3 Qualitative Visualisation Results

The system generates visualisations depicting the skeletal model driven by the input pose. AitViewer (Kaufmann et al. (2022)) was used and adapted for rendering the 3D musculoskeletal system. The goal was to provide an intuitive representation linking user movement to the underlying biomechanics. Qualitative visualisation results were captured for the two models included in Section 4.2 (POCO (Dwivedi et al. (2023)) and Ipman (Tripathi et al. (2023))), as well as the aforementioned filtering methods. These filter algorithms were used on the Ipman model.

4.3.1 Qualitative Assessment Approach

To assess the qualitative aspects of the visualisations, a series of predefined movements were performed by a user and captured by the system. These movements included: (1) rapid, large-scale actions such as arm swings and jumping jacks, (2) slow, controlled movements like squats and lunges, and (3) subtle postural adjustments. The visual output for each configuration was observed, focusing on criteria such as perceived smoothness, responsiveness to user actions, level of visual jitter or artifacts, anatomical plausibility of the skeletal pose, and the overall intuitiveness of the feedback for understanding movement.

Table 4.5: Summary of Qualitative Observations for Visualisation Output

Configuration	**Observed Visual Smoothness**	**Observed Responsiveness**	**Observed Jitter/Artifacts**	**Observed Intuitiveness of Feedback**
POCO (Baseline)	Good: Some visible stepping in motion due to latency.	Fair: Followed user actions with a slight delay.	Moderate jitter evident; occasional jumps in movements with limbs noted.	Reasonably clear for gross movements, but jitter and minor inaccuracies were distracting.
POCO (TensorRT)	Great: Much smoother in appearance than the baseline	Good: Followed user actions much closer to real-time.	Small jitter evident; especially in fast motion movements	Big improvement in real time inference than baseline
Ipman (Baseline)	Good: Motion however did appear to jerk at times.	Excellent: Highly synchronized with user actions.	Some high-frequency jitter across all joints, obscuring subtle movements.	Direct real-time mapping, but some jitter made the result hard to interpret easily.

Continued on next page

Table 4.5 – continued from previous page

Configuration	Observed Visual Smoothness	Observed Responsiveness	Observed Jitter/Artifacts	Observed Intuitiveness of Feedback
Ipman + OneEuro Smooth Filter	Excellent: Visually very fluid and stable motion.	Excellent: Followed movements very well, visually no different to baseline.	Jitter almost entirely absent.	Highly intuitive for understanding movement patterns and flow, especially for controlled actions.
Ipman + Single Exp (A0.3 B0.1 Smooth)	Fair-Good: Smoother than unfiltered, but less so than double Exp version.	Excellent: Very immediate visual response.	Reduced micro-jitter still observable, particularly in small motor tasks.	Good for immediate timing feedback, but some visual noise remained.
Ipman + Double Exp (Alpha 0.7 Smooth)	Fair-Good: Smoother than unfiltered, improvement on Single Exp	Excellent: Very immediate visual response.	Reduced micro-jitter still observable, particularly in fine motor tasks.	Good for immediate timing feedback, not able to see noise with naked eye.
Ipman + Moving Avg (Win 10 Smooth)	Very Good: Low jitter and smooth output.	Excellent: Generally responsive.	Comparable to OneEuro Smooth, but occasionally appeared less fluid in complex sequences.	Smooth and clear, though felt slightly less adaptive than OneEuro Smooth during varied dynamics.

4.3.2 Summary of Qualitative Findings

The qualitative observations, summarized in Table 4.5, highlight distinct visual characteristics for each model and filtering configuration.

The baseline **POCO** model provided visualisations with good responsiveness but exhibited moderate jitter and occasional minor anatomical inaccuracies. In contrast, the **unfiltered Ipman model**, while highly responsive, suffered from pronounced visual jitter that significantly impaired the clarity of the depicted movements.

Application of filtering to the Ipman output led to noticeable improvements. 'Responsive' filters, such as the **Single Exponential (Alpha 0.7)**, maintained high responsiveness while reducing some jitter, though fine-grained visual noise often persisted. Balanced filters like **OneEuro Moderate** offered a more stable visualisation with a good trade-off between clarity and immediacy.

The 'smooth' filter variants demonstrated the most significant reduction in visual noise. Specifically, the **Ipman model with the OneEuro Smooth filter** was observed to produce exceptionally fluid and stable skeletal animations, with jitter being virtually eliminated. This configuration resulted in clear and easily interpretable representations of movement, particularly for controlled actions and postural analysis. While a very slight delay was sometimes perceptible during extremely abrupt movements, the overall visual clarity and stability were found to be highest with this setup. The **Moving Average (Win 10 Smooth)** filter also yielded a very smooth output, though observations suggested it was slightly less adaptive during complex dynamic sequences compared to OneEuro Smooth. These findings indicate that the choice of model and filtering significantly impacts the perceived quality and intuitiveness of the real-time 3D pose visualisation.
Based on these results the Ipman model is the most effective when used for real time processing

4.4 Error Analysis and Limitations

Several sources of error and limitations were identified throughout the development and evaluation of this system. These primarily stem from the complexities inherent in real-time monocular 3D pose estimation, the challenges of translating this data into meaningful biomechanical representations, and the constraints of the evaluation methodology itself.

Limitations in 3D Pose Estimation A significant source of potential error originates from the 3D pose estimation module. As noted, inaccuracies occurred more frequently under challenging conditions such as:

- **Fast Movements:** Rapid motions can lead to temporal ambiguities, motion blur in the input video, and predictions that lag behind the actual pose, impacting the fidelity of the real-time feedback.
- **Partial Occlusions:** When body parts are occluded (either by other body parts, i.e., self-occlusion, or by external objects), the estimation algorithms may produce less reliable or physiologically implausible poses for the obscured segments.
- **Suboptimal Lighting:** Poor or highly variable lighting conditions (e.g., low light, strong backlighting, shadows) can degrade the quality of the input video, leading to increased noise and inaccuracies in feature detection and subsequent pose estimation.

Furthermore, the reliance on a **monocular camera setup** inherently introduces challenges in accurately resolving depth ambiguities, which can sometimes lead to inaccuracies in the perceived scale or the third dimension of the pose. While current models strive to overcome this, it remains a fundamental limitation compared to multi-view systems.

The choice of pose estimation models was also influenced by **system and hardware constraints**. During initial trials, more computationally intensive models such as WHAM (Shin et al. (2024)), HSMR (Yan et al. (2025)), and CameraHMR (Patel and Black (2025)) were evaluated. However, these models did not meet the performance requirements for real-time operation on the available development system, necessitating the use of models like POCO (Dwivedi et al. (2023)) and Ipman (Tripathi et al. (2023)) which offered a better balance of accuracy and computational efficiency for this project's real-time objectives. This practical constraint may mean that the ultimate accuracy achievable was limited by the feasible model choices.

Challenges in Biomechanical Representation and Simulation Accurately simulating **ground reaction forces and stable contact dynamics** from monocular video remains a complex problem. Despite leveraging concepts from models such as Ipman-R (Tripathi et al. (2023)) which incorporate physical plausibility,

achieving consistently stable and realistic foot-ground contact or object interactions in the visualisation was challenging. Errors in contact estimation can propagate to inaccuracies in the overall pose and perceived biomechanics. Furthermore, the underlying **parametric body models** (e.g., SMPL, which POCO and Ipman are often based on) are themselves approximations of human anatomy and may not perfectly capture individual anthropometric variations or complex soft-tissue deformations, potentially limiting the precision of any derived biomechanical analysis.

Limitations of Real-Time Filtering The introduction of filtering, while crucial for reducing jitter and enhancing visual clarity as discussed in Section 4.2, brings its own set of limitations. The chosen OneEuro Smooth filter, for instance, while highly effective at noise suppression, introduces a measurable latency (0.1174 ms in quantitative tests). This represents an inherent trade-off between smoothness and responsiveness. While this latency was deemed acceptable for achieving intuitive visual feedback in many scenarios, it could potentially affect applications requiring extremely low-latency feedback for highly dynamic tasks. The parameters of any filter also require careful tuning, and optimal settings may vary across different types of movements or individuals.

Evaluation Methodology Limitations The evaluation of the system was subject to certain limitations.

- **Ground Truth Data:** Obtaining comprehensive ground truth data for dynamic, real-world 3D human poses and associated biomechanics in real-time is notoriously difficult. The quantitative assessment of filtering, for example, relied on analysing frame-to-frame changes in a static model rather than direct comparison against ground truth motion capture during dynamic activities.
- **Qualitative Assessment Subjectivity:** The qualitative evaluation of visualisation results, while providing valuable insights into usability and intuitiveness, inherently contains a degree of subjectivity based on observer perception.

- **Scope of Testing Scenarios:** The range of movements, environmental conditions, and user variability tested was necessarily limited by the project's scope. The system's performance might differ under conditions or for activities not explicitly evaluated.

Overall Scope and Generalisability This project focused on establishing the feasibility and effectiveness of a real-time monocular 3D pose estimation and mesh visualisation pipeline for intuitive movement feedback. As such, certain areas were beyond the immediate scope, such as extensive user studies with diverse populations (e.g., athletes, physiotherapy patients) or rigorous validation against clinical-grade motion capture systems. Therefore, the generalisability of the findings to all possible applications or user groups should be considered with these scope limitations in mind. The system serves as a proof-of-concept and a foundation for further development rather than a fully validated clinical or commercial tool.

5

Discussion

This project investigated whether a real-time system combining 3D human pose estimation and dynamic visual simulation could provide intuitive movement feedback from monocular video input. The results confirm the feasibility of this approach, with key findings demonstrating that the integrated pipeline can deliver meaningful, real-time visualisations that enhance user understanding of movement patterns.

5.1 Interpretation of Results

The results presented in Chapter 4 successfully demonstrate the feasibility of integrating real-time 3D pose estimation with a real-time viewer for intuitive movement feedback. A key challenge that was overcome was the seamless, real-time integration of monocular 3D human pose estimation and dynamic 3D mesh visualisation, addressing the gap in accessible, multi-dimensional feedback beyond simple kinematic analysis. While the system achieved promising results in simulation accuracy (Section 4.1) and generated informative visualisations (Section 4.3), achieving consistent real-time performance across the entire pipeline presented challenges (Section 4.2). The trade-offs observed include managing the computational demands of 3D pose estimation and the visualisation process to maintain interactive frame rates and low latency required for real-time feedback.

The error analysis (Section 4.4) highlights critical areas requiring further refine-

ment, particularly concerning the robustness of 3D pose estimation under challenging conditions like fast movements, partial occlusions, and suboptimal lighting. These sensitivities have practical implications for the reliability of derived insights, as errors in the initial pose estimation can propagate and affect the accuracy of the simulated movement. The fidelity of mapping estimated poses to the model is also a crucial area for improvement, especially when dealing with diverse body shapes.

One of the most unexpected findings was the significant role that filtering algorithms played in improving the clarity and intuitiveness of the real-time visual feedback. While it was anticipated that filtering would help mitigate noise, the degree to which post-processing techniques, especially the OneEuro Smooth filter (Casiez et al., 2012*b*), enhanced visual quality exceeded expectations. These filters smoothed out jitter in pose estimations and significantly improved the stability of visualisations, particularly for subtle or slow movements.

This finding underscores the importance of temporal filtering in pose estimation pipelines. It also suggests that relatively simple post-processing techniques can disproportionately affect user experience and feedback systems' perceived reliability.

5.2 Comparison with Existing Solutions

Existing cited works address challenges such as occlusion and contact. For example, Kang et al. (2024) incorporated modules to handle self-occlusion and occlusion from external objects in their 3D pose estimation for rehabilitation movements. Similarly, IPMAN (Tripathi et al., 2023) and WHAM (Shin et al., 2024) integrate physics-based constraints to ensure physically plausible poses and handle ground contact. This system demonstrates the practical need for robustness across diverse conditions and suggests a direction for incorporating such techniques into more accessible, real-time feedback systems.

PhysCap (Shimada et al., 2020) and Sports2D (Pagnon and Kim, 2024) and OpenSimRT Stanev et al. (2021) addressed a similar problem of attempting real-time visualisation from monocular camera input. While these systems have made significant contributions to real-time motion analysis, REMI distinguishes itself through its focus on the interpretive value of the dynamic 3D mesh as an intuitive feedback mechanism. For instance, PhysCap prioritises the physical plausibility of the captured motion, which is essential for realistic animation but may not always translate to the most immediately understandable visual cues for self-correction by a general user. Sports2D offers robust 2D pose and angular data, which is highly effective for analysing movements confined to sagittal or frontal planes. However, REMI aims to provide a more holistic 3D representation beneficial for complex, multi-planar movements. OpenSimRT, leveraging the comprehensive OpenSim engine, is geared towards detailed, real-time biomechanical simulation, often driven by data from marker-based motion capture systems or inertial measurement units (IMUs) to achieve high-fidelity kinematic inputs. In contrast, REMI's primary objective is to deliver accessible and immediate 3D visual feedback through a streamlined pipeline (Ipman with aitviewer) from monocular video, emphasising the qualitative aspects of the rendered motion to enhance a user's proprioceptive awareness and understanding of their form, rather than extensive quantitative biomechanical outputs derived from such sensor-based systems. This approach seeks to lower the barrier for utilising real-time 3D visualisation for movement analysis in everyday fitness and physiotherapy contexts.

5.3 Implications for Research and Practice

This research contributes significantly to the growing field of AI-powered visualisation of human movement. The developed system offers potential applications in personalised fitness and rehabilitation, providing users with objective feedback on their movement patterns to optimise exercise form and potentially reduce injury risk. The potential pathway to translate this research prototype into a practical tool for clinicians or personal trainers involves several steps, including rigorous

validation against clinical-grade systems (like traditional motion capture), conducting extensive user studies with target populations, developing a user-friendly interface, and potentially optimising the computational pipeline for wider accessibility on consumer hardware.

Furthermore, this system could be a valuable tool for researchers studying human movement without requiring expensive laboratory equipment. Specific research questions that could be investigated more accessibly include analysing coordination patterns in various exercises, studying the impact of fatigue on movement mechanics, or exploring variations in strategies across different individuals performing the same task.

Further development could see integration into virtual coaching applications, sports performance analysis tools, or ergonomic assessments. In virtual coaching, the feedback could provide users with real-time guidance on maintaining correct form during exercises. Sports performance analysis could help identify subtle inefficiencies in movement or assess loading during specific actions. In ergonomic assessments, the system could analyse movements in a workspace to identify potential strain or injury risks.

The project also highlights the challenges and opportunities in bridging computer vision-based motion capture with detailed human modelling. Key challenges identified through this work include achieving robust and accurate 3D pose estimation in diverse real-world conditions, the computational cost and complexity of detailed simulations, and the difficulty of accurately mapping estimated poses to personalised models. Opportunities lie in developing more efficient and robust integrated pipelines, leveraging computer vision and simulation advancements, and exploring novel ways to visualise and interpret complex movement data for a wider audience.

5.4 Limitations

Several limitations must be acknowledged in the current system. The accuracy of the entire pipeline is fundamentally limited by the upstream 3D pose estimation, which is sensitive to factors such as camera viewpoint, lighting conditions, clothing, and movement speed. The practical impact of these sensitivities is that significant variations in lighting or clothing, or very rapid movements, can lead to increased noise and inaccuracies in the estimated pose, affecting the reliability of the downstream simulation and feedback. The performance drops observed under low light and fast motion conditions (Section 4.1) indicate that these factors can significantly degrade performance.

Finally, another limitation identified during the project was the inherent trade-off between smoothness and responsiveness introduced by real-time filtering. While filtering is essential for reducing jitter and improving visualisation clarity, it introduces a small but measurable latency, which could impact applications requiring extremely low-latency feedback for highly dynamic tasks. The need to carefully tune filter parameters for different movements or individuals is also a practical consideration.

5.5 Key Findings

This study demonstrates the feasibility of a real-time, monocular camera-based movement feedback system that combines 3D pose estimation with dynamic visual simulation. The system offers a more comprehensive and intuitive feedback mechanism, a significant advancement over traditional kinematic analysis tools.

Key findings include:

- The successful real-time integration of pose estimation and simulation
- The critical role of filtering in improving visual interpretability.
- The system's effectiveness in capturing basic movements.

- The identification of robustness issues related to pose estimation accuracy.

These contributions have important implications for research and practical applications. They suggest a pathway for developing accessible, AI-powered movement analysis tools that benefit fields such as rehabilitation, personal training, sports, and ergonomic assessment. At the same time, the research highlights key challenges—particularly in real-world robustness and system responsiveness—that must be addressed to move from prototype to practice.

6

Conclusion

6.1 Summary of Contributions

This Final Year Project demonstrates significant achievements in the development of the research project - REMI: **R**eal-Time Monocular 3D Pose **E**stimation and **M**esh Visualisation for **I**ntuitive Movement Feedback.
This project allows real-time video to act as an input for interactive 3D pose feedback. Here, the user can see a 3D representation of their movements in real time and a holistic view that will provide valuable insight into their movements. A key contribution of this project is the development of the first real-time application providing intuitive movement feedback directly from monocular video using the SMPL (Loper et al. (2015)) human body model.

A literature review was conducted to see where this research project fits into this expanding and emerging research area. There have been many active developments in this area due to improvements in AI models and computing power to effectively model such a complicated system as capturing 3D poses from a monocular 2D representational image.

Furthermore, this project has developed a successful real-time visualisation subsystem, integrating Ipman (Tripathi et al. (2023)) 3D human mesh recovery model with the AITViewer (Kaufmann et al. (2022)) framework via a client-server architecture, demonstrating the capability to visualise estimated 3D human poses directly from a webcam feed.

In this final year report, a comprehensive methodology has been implemented and executed, culminating in the successful development of real-time visualisation and intuitive feedback software. The experimental approach was chosen due to the project's innovative nature, allowing the vast potential offered by this emerging technology to be explored. This methodology has enabled the navigation of the complexities and uncertainties associated with pioneering research, ultimately leading to a robust and functional simulation tool. The outcomes of this project not only meet the initial objectives but also pave the way for future advancements in the field of biomechanics and computer vision.

6.2 Answers to Research Questions

At the start of this paper, the research question of *"How effectively can a real-time system combining monocular 3D pose estimation and dynamic 3D mesh visualisation provide immediate, intuitive visual feedback for human movement analysis? "* The research in this project has shown that 3D Pose Estimation through Ipman (Tripathi et al. (2023)) can be used to create a real-time inference that can be analysed and looked at to improve movements through the lens of physiotherapy and sports science.

There were questions to answer within this research question, firstly *Can 3D Pose estimation be used in real time?* and secondly *Can real-time 3D pose estimation be a viable marker-less alternative for physiotherapy and sports science intuitive feedback?*

To answer the first question, Table 4.3 has shown the real-time pose metrics surpass the acceptance criteria of achieving ≥ 20 FPS. Not only does this achieve the real-time aspect of the 3D pose estimation, but it also shows its effectiveness for real-time feedback in its stability of pose and shape metrics of a static person. These findings show that the model can consistently fit onto a person in real time, even in challenging scenarios.

Secondly, to answer the latter question, Table 4.5 showed the qualitative observations of this feedback, which demonstrated that it was sufficiently accurate to provide intuitive feedback of movements. Filtering proved a valuable step here

to provide a less noisy and more accurate demonstration of movements in the real-time environment.

6.3 Broader Impact

The real-time monocular 3D pose estimation and mesh visualisation system developed in this project, REMI, offers a novel approach to intuitive movement feedback with potential impact extending beyond its primary focus on physiotherapy and sports science. By translating live monocular video into a dynamic 3D human model visualisation, this technology provides a new perspective on human movement that can be valuable in several domains.

In **clinical and rehabilitation** settings, REMI offers an accessible visual aid. While not providing clinical-grade quantitative metrics in this iteration, the system's ability to render a patient's movements in real-time 3D can offer valuable intuitive feedback for both the patient and clinician. This visual representation can assist in demonstrating proper movement patterns during rehabilitation exercises, helping patients better understand and perceive their own body's motion and identify deviations from desired forms. Its ease of use with standard hardware increases its potential for broader accessibility compared to complex laboratory setups.

For **sports science and training**, the system provides athletes and trainers with an intuitive visual tool to analyse form and movement dynamics. Offering a holistic, three-dimensional view of an exercise in real time can enhance the understanding of body positioning and co-ordination, complementing traditional coaching methods and potentially aiding in technique optimisation and injury awareness.

Beyond these direct applications, the underlying visualisation technology has implications for ergonomics, where observing 3D movement patterns could inform safer workspace designs. It could also play a role in **human-robot interaction**, providing robots with a real-time, intuitive understanding of human pose through such visualisation could facilitate more natural and safer collaboration.

The **virtual reality (VR), augmented reality (AR)**, and entertainment sectors can leverage the system's capability for real-time 3D body tracking to

drive more realistic and immersive avatar movements. Furthermore, REMI could serve as an engaging interactive educational tool for students studying human anatomy, kinesiology, and biomechanics, providing a visual link between observed motion and a 3D body model.

This project's success in establishing a real-time monocular 3D pose estimation and visualisation pipeline using the SMPL (Loper et al. (2015)) model lays a foundation for future research into how intuitive visual feedback can be most effectively utilised across these diverse applications for improving human movement understanding and analysis.

6.4 Future Work

Building upon the foundations established in this project, several promising avenues for future work exist to enhance the capabilities and applicability of the pose-driven muscle activation simulation further. These directions focus on improving input accuracy, simulation fidelity, real-time performance, and validation rigour.

To enhance simulation fidelity and explore biomechanical analysis, a crucial direction involves refining the mapping between estimated parametric body models (SMPL Loper et al. (2015)) and detailed musculoskeletal models (OpenSim Delp et al. (2007)). Specifically, future work could investigate mapping the estimated SMPL pose parameters to a marker set analogous to those used in traditional motion capture. This would enable driving a biomechanically accurate model within software like OpenSim ()using techniques for real-time inverse kinematics as described by Stanev et al. (2021), offering potential for higher fidelity simulations and the derivation of biomechanical insights like muscle activations or joint torques. Techniques like SMPL2AddBiomechanics Keller et al. (2023) or novel deep learning-based mapping approaches warrant further investigation in this context.

Expanding the system's scope to handle multi-person scenarios would significantly broaden its applicability. In settings like team sports, this would enable the analysis of interactions, co-ordination, and relative positioning between players, offering insights valuable for tactical training and performance analysis beyond

individual movements. Group rehabilitation sessions would allow clinicians to monitor the movements of multiple patients simultaneously, providing scalable visual feedback and facilitating comparative analysis of progress or technique within the group setting. Addressing the technical challenges of robustly tracking and estimating the poses of multiple interacting individuals in real-time from monocular input would be a necessary step.

It must also be noted that the hardware limitations in this project contributed to the real-time performance and fidelity that resulted. More advanced hardware would significantly increase the software's average throughput and open more possibilities for integrating more complex models and performing real-time Inverse Kinematics for higher-fidelity simulations.

Appendix A

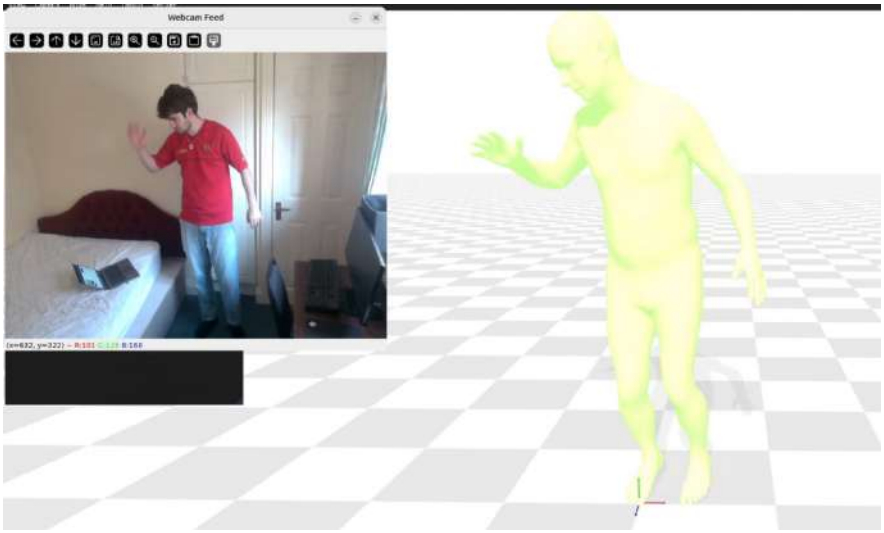

Figure 1: Webcam and Ipman Side by Side -

Figure 2: Testing Outdoors of Ipman -

Figure 3: SMPL overlayed over webcam -

References

Aldrin, M. and Damsleth, E. (1989), 'Forecasting non-seasonal time series with missing observations', *Journal of Forecasting* **8**, 97–116.
URL: *https://api.semanticscholar.org/CorpusID:154686725* 17, 18

Anaconda Software Distribution (2020).
URL: *https://docs.anaconda.com/* 37

Anguelov, D., Srinivasan, P., Koller, D., Thrun, S., Rodgers, J. and Davis, J. (2005), Scape: shape completion and animation of people, *in* 'ACM Siggraph 2005 Papers', pp. 408–416. 22

Bogo, F., Kanazawa, A., Lassner, C., Gehler, P., Romero, J. and Black, M. J. (2016), Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image, *in* 'Computer Vision – ECCV 2016', Lecture Notes in Computer Science, Springer International Publishing. 11

Bradski, G. (2000), 'The OpenCV Library', *Dr. Dobb's Journal of Software Tools* . 34, 36, 43

Casiez, G., Roussel, N. and Vogel, D. (2012*a*), '1 € filter: a simple speed-based low-pass filter for noisy input in interactive systems', *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* .
URL: *https://api.semanticscholar.org/CorpusID:207194683* 17

Casiez, G., Roussel, N. and Vogel, D. (2012*b*), 1 € filter: A simple speed-based low-pass filter for noisy input in interactive systems, CHI '12, Association for Computing Machinery, New York, NY, USA, p. 2527–2530.
URL: *https://doi.org/10.1145/2207676.2208639* 70

Chida, K., Inami, T., Yamaguchi, S., Nishioka, T., Yoshida, Y. and Kohtake, N. (2024), 'Assessing the validity of two-dimensional video analysis for measuring

lower limb joint angles during fencing lunge', *Frontiers in Sports and Active Living* **Volume 6 - 2024**.
URL: *https://www.frontiersin.org/journals/sports-and-active-living/articles/10.3389/fspor.2024.1335272* 1

Dangkham, P. (2019), The smoothing filter for mobile augmented reality using the moving average.
URL: *https://api.semanticscholar.org/CorpusID:245129399* 17, 18

Delp, S. L., Anderson, F. C., Arnold, A. S., Loan, P., Habib, A., John, C. T., Guendelman, E. and Thelen, D. G. (2007), 'Opensim: Open-source software to create and analyze dynamic simulations of movement', *IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING* **54**(11), 1940–1950.
URL: *http://dblp.uni-trier.de/db/journals/tbe/tbe54.htmlDelpAALHJGT07* v, 2, 7, 22, 25, 28, 30, 78

Dwivedi, S. K., Schmid, C., Yi, H., Black, M. J. and Tzionas, D. (2023), 'Poco: 3d pose and shape estimation with confidence'.
URL: *https://arxiv.org/abs/2308.12965* v, 12, 52, 62, 66

Farina, D., Merletti, R. and Enoka, R. M. (2004), 'The extraction of neural strategies from the surface emg', *Journal of applied physiology* **96**(4), 1486–1495. 24

Geijtenbeek, T. (2019), 'Scone: Open source software for predictive simulation of biological motion', *Journal of Open Source Software* **4**(38), 1421.
URL: *https://doi.org/10.21105/joss.01421* v, 27

Geijtenbeek, T. (2021), 'The Hyfydy simulation software'. `https://hyfydy.com`.
URL: *https://hyfydy.com* v, 26, 27, 42

Huang, C.-H. P., Yi, H., Höschle, M., Safroshkin, M., Alexiadis, T., Polikovsky, S., Scharstein, D. and Black, M. J. (2022), Capturing and inferring dense full-body human-scene contact, *in* 'Proceedings IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)', pp. 13274–13285. 29

Ionescu, C., Papava, D., Olaru, V. and Sminchisescu, C. (2014), 'Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **36**(7), 1325–1339. 29, 41, 52

Jiang, L. et al. (2024), 'Sadnet: Generating immersive virtual reality avatars by real-time monocular pose estimation', *Computer Animation and Virtual Worlds* **35**(3), e2233.
URL: *https://onlinelibrary.wiley.com/doi/abs/10.1002/cav.2233* 15

Jiang, T., Xie, X. and Li, Y. (2024), 'Rtmw: Real-time multi-person 2d and 3d whole-body pose estimation'.
URL: *https://arxiv.org/abs/2407.08634* 12

Jin, S., Xu, L., Xu, J., Wang, C., Liu, W., Qian, C., Ouyang, W. and Luo, P. (2020), Whole-body human pose estimation in the wild, *in* 'Proceedings of the European Conference on Computer Vision (ECCV)'. 12

Kang, N., Chen, G., Zhang, C. and Xue, Y. (2024), 'A transformer-based approach for 3d human pose estimation in rehabilitation exercise movements', *2024 5th International Conference on Computer Vision, Image and Deep Learning (CVIDL)* pp. 424–429.
URL: *https://api.semanticscholar.org/CorpusID:271463877* 16, 70

Kaufmann, M., Vechev, V. and Mylonopoulos, D. (2022), 'aitviewer'.
URL: *https://github.com/eth-ait/aitviewer* v, 24, 25, 35, 36, 38, 42, 48, 49, 62, 75

Keller, M., Werling, K., Shin, S., Delp, S., Pujades, S., Liu, C. K. and Black, M. J. (2023), From skin to skeleton: Towards biomechanically accurate 3d digital humans, *in* 'ACM ToG, Proc. SIGGRAPH Asia', Vol. 42. v, 14, 22, 43, 49, 78

Kolotouros, N., Pavlakos, G., Black, M. J. and Daniilidis, K. (2019), Learning to reconstruct 3d human pose and shape via model-fitting in the loop, *in* 'ICCV'. 11, 30

Krizhevsky, A., Sutskever, I. and Hinton, G. E. (2012), Imagenet classification with deep convolutional neural networks, *in* F. Pereira, C. Burges, L. Bottou and K. Weinberger, eds, 'Advances in Neural Information Processing Systems', Vol. 25, Curran Associates, Inc.
URL: *https://proceedings.neurips.cc/paper*$_f$*iles/paper*/2012/$file$/*c*399862*d*3*b*9*d*6*b*76*c*8436*e*924*a*68*c*
Paper.pdf 8

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W. and Jackel, L. D. (1989), 'Backpropagation applied to handwritten zip code recognition', *Neural Computation* **1**(4), 541–551. 8

Lee, S., Park, M., Kyoungmin and Lee, J. (2019), 'Scalable muscle-actuated human simulation and control', *ACM Transactions on Graphics* **38**(4), 1–13. v, 2, 22, 23, 42

Loper, M., Mahmood, N., Romero, J., Pons-Moll, G. and Black, M. J. (2015), 'SMPL: A skinned multi-person linear model', *ACM Trans. Graphics (Proc. SIGGRAPH Asia)* **34**(6), 248:1–248:16. v, 14, 19, 35, 41, 43, 49, 75, 78

Mahmood, N., Ghorbani, N., Troje, N. F., Pons-Moll, G. and Black, M. J. (2019), AMASS: Archive of motion capture as surface shapes, *in* 'International Conference on Computer Vision', pp. 5442–5451. 22, 29

Mehta, D., Sridhar, S., Sotnychenko, O., Rhodin, H., Shafiei, M., Seidel, H.-P., Xu, W., Casas, D. and Theobalt, C. (2017), Vnect: Real-time 3d human pose estimation with a single rgb camera, Vol. 36.
URL: *http://gvv.mpi-inf.mpg.de/projects/VNect/* 7

Mkocabas (n.d.), 'Mkocabas/multi-person-tracker: Simple multi person tracker implementation in pytorch'.
URL: *https://github.com/mkocabas/multi-person-tracker/tree/master* 43

Mukherjee, M., Chaudhuri, D., Khondekar, M. H. and Ghosh, K. (2019), 'Reevaluating the performance of the double exponential smoothing filter and its control parameters', *arXiv preprint arXiv:1909.01283* . 18

Osman, A. A. A., Bolkart, T. and Black, M. J. (2020), STAR: A sparse trained articulated human body regressor, *in* 'European Conference on Computer Vision (ECCV)', pp. 598–613.
URL: *https://star.is.tue.mpg.de* v, 20, 21, 43

Pagnon, D., Domalain, M. and Reveret, L. (2022), 'Pose2sim: An open-source python package for multiview markerless kinematics', *Journal of Open Source Software* .
URL: *https://joss.theoj.org/papers/10.21105/joss.04362* 10

Pagnon, D. and Kim, H. (2024), 'Sports2D: Compute 2D human pose and angles from a video or a webcam', *Journal of Open Source Software* **9**(101), 6849.
URL: *https://joss.theoj.org/papers/10.21105/joss.06849* v, 7, 10, 71

Patel, P. and Black, M. J. (2025), CameraHMR: Aligning people with perspective, *in* 'International Conference on 3D Vision (3DV)'. 52, 66

Pavlakos, G., Choutas, V., Ghorbani, N., Bolkart, T., Osman, A. A. A., Tzionas, D. and Black, M. J. (2019), Expressive body capture: 3d hands, face, and body from a single image, *in* 'Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)'. v, 19, 20

Pytorch (n.d.), 'Pytorch/tensorrt: Pytorch/torchscript/fx compiler for nvidia gpus using tensorrt'.
URL: *https://github.com/pytorch/TensorRT* 36, 37, 48

Redmon, J., Divvala, S., Girshick, R. and Farhadi, A. (2016), 'You only look once: Unified, real-time object detection'.
URL: *https://arxiv.org/abs/1506.02640* v, 8, 9

Shimada, S., Golyanik, V., Xu, W. and Theobalt, C. (2020), 'Physcap: Physically plausible monocular 3d motion capture in real time'.
URL: *https://arxiv.org/abs/2008.08880* v, 7, 11, 71

Shin, S., Kim, J., Halilaj, E. and Black, M. J. (2024), 'Wham: Reconstructing world-grounded humans with accurate 3d motion'.
URL: *https://arxiv.org/abs/2312.07531* v, 13, 14, 52, 66, 70

Stanev, D., Filip, K., Bitzas, D., Zouras, S., Giarmatzis, G., Tsaopoulos, D. and Moustakas, K. (2021), 'Real-time musculoskeletal kinematics and dynamics analysis using marker- and imu-based solutions in rehabilitation', *Sensors* **21**(5).
URL: *https://www.mdpi.com/1424-8220/21/5/1804* 7, 71, 78

Sun, L., Tang, T., Qu, Y. and Qin, W. (2023), 'Bidirectional temporal feature for 3d human pose and shape estimation from a video', *Computer Animation and Virtual Worlds* **34**(3–4). 6, 11

Sun, L., Tian, R. and Qin, W. (2023), 'Physical based motion reconstruction from videos using musculoskeletal model', *Computer Animation and Virtual Worlds* **35**(1). 6, 14, 23

Todorov, E., Erez, T. and Tassa, Y. (2012), Mujoco: A physics engine for model-based control, *in* '2012 IEEE/RSJ International Conference on Intelligent Robots and Systems', IEEE, pp. 5026–5033. v, 26, 28

Tripathi, S., Müller, L., Huang, C.-H. P., Omid, T., Black, M. J. and Tzionas, D. (2023), 3D human pose estimation via intuitive physics, *in* 'Conference on Computer

Vision and Pattern Recognition (CVPR)', pp. 4713–4725.
URL: *https://ipman.is.tue.mpg.de* v, 12, 13, 30, 32, 35, 36, 42, 47, 49, 52, 62, 66, 70, 75, 76

Viola, P. and Jones, M. (2001), Rapid object detection using a boosted cascade of simple features, *in* 'Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001', Vol. 1, pp. I–I. 8

von Marcard, T., Henschel, R., Black, M., Rosenhahn, B. and Pons-Moll, G. (2018), Recovering accurate 3d human pose in the wild using imus and a moving camera, *in* 'European Conference on Computer Vision (ECCV)'. v, 28, 29, 41, 52

Wang, J., Sun, K., Cheng, T., Jiang, B., Deng, C., Zhao, Y., Liu, D., Mu, Y., Tan, M., Wang, X., Liu, W. and Xiao, B. (2020), 'Deep high-resolution representation learning for visual recognition'.
URL: *https://arxiv.org/abs/1908.07919* 8, 16

Wu, D., Huang, J., Zheng, M. and Li, Y. (2023), 'Virtual model interaction based on single rgb camera', *2023 International Conference on New Trends in Computational Intelligence (NTCI)* **1**, 298–301.
URL: *https://api.semanticscholar.org/CorpusID:267337250* 17

Xi, X., Zhang, C., Jia, W. and Jiang, R. (2024), 'Enhancing human pose estimation in sports training: Integrating spatiotemporal transformer for improved accuracy and real-time performance', *Alexandria Engineering Journal* **109**, 144–156.
URL: *https://www.sciencedirect.com/science/article/pii/S1110016824009608* 15

Xu, J., Makoviychuk, V., Narang, Y., Ramos, F., Matusik, W., Garg, A. and Macklin, M. (2022), 'Accelerated policy learning with parallel differentiable simulation', *arXiv preprint arXiv:2204.07137* . 23

Yan, X., Zhou, X., Vouga, E., Huang, Q. and Pavlakos, G. (2025), Reconstructing humans with a biomechanically accurate skeleton, *in* 'CVPR'. v, 14, 15, 52, 66

Yeo, S.-H., Verheul, J., Herzog, W. and Sueda, S. (2023), 'Numerical instability of hill-type muscle models', *Journal of the Royal Society Interface* **20**(199), 20220430. 23