VILNIUS UNIVERSITY

FACULTY OF MATHEMATICS AND INFORMATICS

DATA SCIENCE STUDY PROGRAMME

Master's Thesis

Development of a tool for assessing hip abduction during hip-spica casting in patients with developmental dysplasia of the hip

Įrankio, skirto klubo sąnario displazija sergančių pacientų klubo sąnario atvedimo įvertinimui gipsavimo metu, kūrimas

Inga Garšvaitė

Supervisor : Asist. Dr Julius Venskus

Scientific advisor : Md. Julija Ravinskienė

Vilnius
2026

Acknowledgements

I am grateful to my supervisor, Julius Venskus, for proposing the thesis topic and for his continuous guidance throughout this work. I especially appreciate his trust in my decisions and his support throughout the process. The opinion and advice he gave significantly improved the quality of the final thesis.

For clinical guidance, I am thankful to Julia Ravinskiene for generously sharing her knowledge and for helping me deepen my understanding of the topic. Her trust, insights, and shared experience allowed me to view the problem from a clinical perspective and explore it more thoroughly.

Special thanks go to all the people who participated in the experiment —without their input, this work would not have been possible.

I would also like to thank Liutauras Baltrūnas for the idea of ensuring a robust ground-truth angle during data collection.

Finally, I sincerely thank my coworkers, friends, and family for their constant support and encouragement throughout this journey.

Contents

Summary 6

Santrauka 7

List of symbols 8

List of abbreviations 10

1 Introduction 13

2 Literature Review 15

2.1 Developmental Dysplasia of the Hip 15

2.1.1 Diagnosis and treatment of DDH 15

2.1.2 Hip abduction angle measurement during DDH 17

2.2 Pose-estimation models: a novelty to include in clinical workflows 18

2.2.1 High-level review of the modern pose estimation models 19

2.2.2 MB: framework 20

2.2.3 MB: performance overview and limitations 21

3 Methodology 23

3.1 System overview 23

3.1.1 System's evaluation pipeline 24

3.2 Data collection 24

3.2.1 Participants 24

3.2.2 Experimental variables: TRUE, `CAM` and FLEX 25

3.2.3 Experimental setup 26

3.3 System architecture 26

3.3.1 Video input acquisition and processing 26

3.3.2 Pose estimation with MB 26

3.3.3 Angle estimation from the landmarks coordinates 28

3.4 Statistical analysis 29

3.4.1 Single-trial metrics 30

3.4.2 Participant-condition metrics 31

3.4.3 Configuration-level evaluation across participants 31

3.4.4 Evaluation of the MAE 31

3.4.5 Evaluation of Noise 32

3.4.6 Temporal stabilisation - applying OEF 32

3.4.7 Prediction tasks and evaluation protocol 33

4 Results 37

4.1 Exploratory data analysis . . . 37
4.1.1 Dataset structure and notation . . . 37
4.1.2 Error distribution across configurations . . . 37
4.1.3 Agreement analysis (Bland–Altman) . . . 38
4.1.4 Zone-based classification and confusion matrix . . . 38
4.2 Mixed-effects modelling . . . 40
4.2.1 Outcome transformation and model set . . . 40
4.2.2 Baseline technical model (m0) . . . 40
4.2.3 Extended covariate model (m3) . . . 41
4.2.4 Selected model (m6): best overall fit and effect interpretation . . . 41
4.3 Global noise reduction analysis . . . 42
4.3.1 Global parameter tuning . . . 42
4.3.2 Global effect of filtering on frame-to-frame jitter . . . 42
4.4 Classification performance . . . 43
4.4.1 All features . . . 43
4.4.2 CAM–FLEX . . . 44
4.4.3 Estimation only . . . 44
4.4.4 Estimation + vis . . . 44
4.4.5 Summary across feature sets . . . 45
4.4.6 Confusion matrix (best model) . . . 45
4.5 Regression performance . . . 45
4.5.1 All features . . . 45
4.5.2 CAM–FLEX . . . 46
4.5.3 Estimation only . . . 46
4.5.4 Estimation + vis . . . 47
4.5.5 Effect of OEF (CAM–FLEX, LOPO) . . . 47
4.5.6 Key findings (LOPO) . . . 48

5 Discussion . . . **49**
5.1 Systematic error in HAA estimation . . . 49
5.2 Interpretation of secondary predictors and configuration effects . . . 50
5.3 Comparing the performances of classification and regression performances . . . 50
5.4 Clinical interpretation of `CAM` and `FLEX` effects and implications for machine learning . . . 51
5.5 Effect of temporal smoothing: improved stability but limited accuracy gain . . . 52

6 Limitations and Future Work . . . **53**
6.1 Limitations . . . 53
6.2 Future work . . . 54

7 Conclusions . . . **55**

8 References and sources . . . **56**

9 Appendices 66

A Heatmaps for grid search 66

B ANOVA / mixed-effects model outputs 67

C Python code 71

D R code 84

Summary

Developmental Dysplasia of the Hip (DDH) is a spectrum of hip abnormalities in infants ranging from joint instability to complete hip dislocation. In the surgical management of DDH, the correct hip abduction angle is essential for treatment success: insufficient abduction increases the risk of hip redislocation, while excessive abduction increases the risk of avascular necrosis. Despite the clinical importance of maintaining a “safe zone,” casting decisions are often based on subjective visual estimation. This thesis presents the development and validation of a Technological Readiness Level (TRL) 4 prototype for estimating hip abduction angle from monocular video and pose estimation. The prototype was evaluated using data collected from eleven healthy adults under a standardized protocol. Results showed a systematic underestimation of the angle, and the mean absolute error increased at higher abduction values, suggesting range compression. One Euro filtering reduced frame-to-frame jitter by approximately 83%, but did not improve abduction angle estimation. Zone-based decision support demonstrated high performance, with a Support Vector Machine (SVM) classification model providing the best Leave-one-patient-out generalization accuracy (0.909) in the reported experiments. The Support Vector Regressor performed best among the regression models, with R^2 = 0.713, suggesting that the current prototype may be more useful for zone-based decision support than for precise angle estimation.

Keywords: hip abduction angle, MediaPipe, BlazePose, One Euro filter, pose estimation, pediatric orthopedics, Developmental Dysplasia of the Hip (DDH), avascular necrosis

Santrauka

Klubo sąnario displazija (angl. *Developmental Dysplasia of the Hip*, DDH) yra vadinama vaikams pasireiškiantys klubo sąnario pakitimai, varijuojantys nuo sąnario nestabilumo iki visiško išnirimo. Atlikus operaciją, tinkamas klubo atvedimo kampas gipsuojant yra pagrindinis veiksnys, lemiantis kokybišką sąnario gijimą: per mažas atvedimo kampas didina klubo išnirimo pasikartojimo riziką, o per didelis - klubo osteonekrozės riziką. Nepaisant to, gipsavimo metodikos dažnai grindžiamos subjektyviu vizualiu klubo atvedimo įvertinimu. Šiame darbe pristatomas ketvirtos technologinės parengties prototipo kūrimas, skirtas įvertinti klubo atvedimo kampą naudojant kompiuterinės regos metodais pagrįstas kūno pozos nustatymo technologijas. Prototipas buvo vertinamas remiantis eksperimento metu, kuriame dalyvavo vienuolika sveikų suaugusiųjų, surinktais duomenimis. Rezultatai parodė, jog sukurtas prototipinis įrankis sistemingai nuvertina atvedimo kampo dydį, o didėjant tikrąjai šio kampo vertei, vidutinė absoliuti paklaida taip pat didėja. Signalo filtravimo metodas „One Euro“ sumažino tarpusavio kadrų virpesius apie 83%, tačiau nesumažino vidutinės absoliučios paklaidos įverčio. Klinikinių atvedimo kampo zonų nustatymu pagrįsta sistema pasižymėjo sąlyginai aukštu veikimo efektyvumu, o atraminių vektorių metodas pasiekė geriausią generalizacijos tikslumą (0.909), testuojant ant vis skirtingų pacientų. Atraminių vektorių metodas taip pat pasižymėjo aukščiausiais įverčiais tarp regresijos modelių, pasiekdamas R^2 = 0.713. Šie rezultatai rodo, kad dabartinė prototipo versija labiau tinkama kaip pagalbinis įrankis atvedimo kampo zonų nustatytimui, nei tiksliam kampo įvertinimui.

Raktiniai žodžiai: klubo sąnario atvedimo kampas, MediaPipe, BlazePose, One Euro filtras, skaitmeninis pozos nustatymas, pediatrinė ortopedija, klubo sąnario displazija, klubo osteonekrozė.

List of symbols

- t – time index (frame).
- LH – left hip landmark coordinates in the image plane.
- RH – right hip landmark coordinates in the image plane.
- LK – left knee landmark coordinates in the image plane.
- RK – right knee landmark coordinates in the image plane.
- H – pelvic midpoint, $H = \frac{LH+RH}{2}$.
- $\vec{v}_L$ – left thigh vector, $\vec{v}_L = H - LK$.
- $\vec{v}_R$ – right thigh vector, $\vec{v}_R = H - RK$.
- $|\vec{a}|$ – Euclidean norm of vector $\vec{a}$.
- $\vec{a} \cdot \vec{b}$ – dot product of vectors $\vec{a}$ and $\vec{b}$.
- θ – total (inter-thigh) hip abduction angle.
- $\hat{\theta}(t)$ – per-frame estimated abduction angle at frame t.
- $\{\theta_{r,t}\}_{t=1}^{T_r}$ – abduction-angle time series for trial r.
- T_r – number of retained (valid) frames in trial r.
- θ_{true} – ground-truth (target) total abduction angle (degrees).
- $\bar{\theta}_r$ – trial-level mean estimated abduction angle for trial r.
- $\bar{\theta}_{\text{p}}$ – participant–condition mean estimated abduction angle.
- $\bar{v}_r$ – trial-level mean visibility for trial r.
- $\bar{v}_{\text{p}}$ – participant–condition mean visibility.
- R – number of repeated trials within a participant–condition.
- $v_J(t)$ – visibility score for landmark J at frame t, $v_J(t) \in [0,1]$.
- MAE$_r$ – within-trial mean absolute error for trial r (degrees).
- RMSE$_r$ – within-trial root mean squared error for trial r (degrees).
- MAE$_{\text{p}}$ – participant–condition mean absolute error (degrees).
- Noise$_r$ – within-trial mean absolute first difference (deg/frame).

- Δ_r – paired noise difference, $\Delta_r = \text{Noise}_{r,\text{filtered}} - \text{Noise}_{r,\text{raw}}$.
- r_{rb} – rank-biserial correlation (Wilcoxon effect size).
- x_t – raw angle at frame t.
- $\hat{x}_t$ – OEF-filtered angle at frame t.
- f_{min} – minimum (baseline) cutoff frequency (Hz).
- β – OEF adaptation parameter.
- $f_c(t)$ – adaptive cutoff frequency at frame t (Hz).
- $\widehat{\dot{x}}_t$ – filtered estimate of signal speed (derivative magnitude).
- $J(f_{\text{min}},\beta)$ – OEF tuning objective function.
- α – significance level (Type I error rate), e.g., $\alpha = 0.05$.
- `log1p_mae` – transformed MAE outcome, $\log(1 + \text{MAE})$.
- df – degrees of freedom (Satterthwaite approximation).
- t – t-statistic for a fixed-effect term.
- F – F-statistic (Type III ANOVA) for a fixed-effect term.
- p – p-value associated with a test statistic.
- b_i – participant-specific random intercept for participant i.
- σ_b^2 – variance of random intercepts across participants.
- ε – residual error term in the mixed-effects model.
- $m0, m1, \ldots, m9$ – candidate mixed-effects model specifications compared by AIC/BIC.
- R^2 – coefficient of determination for regression.

List of abbreviations

- **AIC** — Akaike Information Criterion
- **ANOVA** — Analysis of Variance
- **API** — Application Programming Interface
- **AVN** — Avascular Necrosis
- **BA** — Bland–Altman (analysis/plot)
- **BGR** — Blue–Green–Red (image channel order)
- **BIC** — Bayesian Information Criterion
- **BMI** — Body Mass Index
- **CAM** — Camera configuration / camera angle condition (e.g., $0°$ or $45°$)
- **CNN** — Convolutional Neural Network
- **COCO** — Common Objects in Context (dataset / keypoint topology reference)
- **CR** — Closed Reduction
- **CT** — Computed Tomography
- **df** — Degrees of freedom
- **DDH** — Developmental Dysplasia of the Hip
- **FLEX** — Hip flexion configuration in the experimental setup (e.g., $70°$ or $90°$)
- **FDR** — False Discovery Rate
- **FPS** — Frames per second
- **GPT** — Generative Pre-trained Transformer
- **GPU** — Graphics Processing Unit
- **HAA** — Hip Abduction Angle
- **HSC** — Hip-spica casting
- **HRNet** — High-Resolution Network (pose estimation architecture)
- **ICC** — Intraclass Correlation Coefficient
- **IQR** — Interquartile Range

- **JB** — Jarque–Bera (test)
- **JSON** — JavaScript Object Notation
- **KNN** — k-Nearest Neighbors
- **LM** — Linear Model (ordinary least squares regression)
- **LOPO** — Leave-One-Participant-Out (cross-validation)
- **MAE** — Mean Absolute Error
- **MB** — MediaPipe BlazePose
- **ML** — Machine Learning
- **MLR** — Multinomial Logistic Regression
- **MRI** — Magnetic Resonance Imaging
- **OEF** — One Euro Filter (1€ filter)
- **OpenCV** — Open Source Computer Vision Library
- **OR** — Open Reduction
- **RAW** — Unfiltered (original) signal / dataset
- **RBF** — Radial Basis Function (kernel)
- **RGB** — Red–Green–Blue (image format)
- **RMSE** — Root Mean Square Error
- **ROC-AUC** — Receiver Operating Characteristic – Area Under the Curve
- **ROI** — Region of Interest
- **RF** — Random Forest
- **SD** — Standard Deviation
- **SE** — Standard Error
- **SVR** — Support Vector Regression
- **SVM** — Support Vector Machine
- **TRL** — Technology Readiness Level
- **TRL4** — Technology Readiness Level 4
- **TRUE** — Ground-truth / target abduction angle condition (e.g., 70°, 100°, 130°)

- **XGB** — Extreme Gradient Boosting (XGBoost)
- **Q1** — First quartile
- **Q3** — Third quartile
- **ZeroR** — Majority-class baseline classifier

1 Introduction

Developmental Dysplasia of the Hip (DDH) comprises a spectrum of hip abnormalities, ranging from instability and acetabular dysplasia to subluxation and complete dislocation [2, 62]. Early diagnosis and treatment are strongly associated with favourable outcomes, as many cases can be managed using safe, cost-effective, and non-invasive methods [85]. Clinical management typically follows a stepwise strategy in which the least invasive options are attempted first. When bracing fails or diagnosis is late, open reduction (OR) or closed reduction (CR) followed by hip-spica casting (HSC) is commonly used [2, 68]. However, treatment success is influenced by the hip abduction angle (HAA) during HSC, with excessive abduction associated with a higher avascular necrosis (AVN) of the femoral head risk [47].

To avoid hip redislocation or increased risk of AVN, clinically defined "safe zone" of HAA during HSC was established [65]. In practice, casting decisions are often guided by subjective visual estimation, as a standardised method for real-time hip position assessment is lacking. While imaging-based approaches can provide additional information, they may be impractical due to workflow constraints, resource requirements, and limited real-time feedback [42].

These limitations motivate the development of a fast, reliable, and standardised tool that supports evidence-informed decision-making and reduces procedural variability. Recent advances in computer vision and deep learning have enabled scalable, markerless approaches that estimate body keypoints directly from monocular Red-Green-Blue (RGB) video [20, 79]. In particular, Mediapipe's BlazePose (MB) real-time full-body pose estimation approach enables mobile deployment and is compatible with a sterile, no-contact clinical environment, which would be difficult to achieve with other imaging techniques [8].

Therefore, **the aim of this thesis is to develop and validate a Technology Readiness Level 4 (TRL4) prototype tool for assessing hip abduction angles using pose-estimation techniques.** The system's evaluation corresponds to a TRL4 prototype and was carried out under controlled conditions with representative users to assess feasibility before clinical validation. To achieve this aim, the work followed a structured methodology: (1) collecting experimental data under conditions simulating HSC, (2) implementing pose estimation technology and utilizing it for HAA computation, (3) quantifying systematic error against ground-truth measurements, and (4) investigating signal processing and machine learning methods to improve robustness and reduce measurement noise. A key challenge addressed in this work is the geometrical limitation of estimating 3D HAA from 2D monocular projections, particularly in flexed postures.

Objectives:

1. Review scientific literature on pediatric hip abduction measurement and pose-estimation technologies to identify and select clinically appropriate data-processing methods.
2. Design and develop the system architecture and algorithm, by selecting the most suitable method, and perform theoretical calculations to assess the system's functionality and feasibility.

3. Develop a prototype model of the TRL4 using the selected framework, designed for integration into clinical casting decision-making workflows.

4. Validate the model in a relevant environment by comparing its calculated hip abduction angles with ground-truth measurements.

5. Analyze the validation results and propose signal processing and machine learning improvements to increase accuracy, reduce noise, and improve reliability under varied recording conditions.

The thesis is structured as follows: Chapter 2 presents the literature review. Chapter 3 describes the methodology, including the system framework and the statistical analysis. Chapter 4 reports the experimental results and the development of the final model used in the prototype. Chapter 5 discusses the findings in the context of prior work and clinical relevance. Chapter 6 outlines the limitations of the study and directions for future work, including potential extensions of the proposed tool. Finally, Chapter 7 concludes the thesis and summarises the main contributions.

2 Literature Review

2.1 Developmental Dysplasia of the Hip

DDH refers to a wide spectrum of abnormalities ranging from hip instability and acetabular dysplasia to subluxation and complete dislocation (**see Figure 1**) [2, 62]. While DDH could be caused by outer conditions affecting the structural parts of the hip (e.g., trauma), it is usually congenital and varies between 1.5 to 20 per 1 000 live births, depending on population and screening method [9]. The exact etiology of DDH remains unknown and is likely multifactorial, with possible risk factors including breech presentation, family history, and genetics, with females more frequently affected [77]. DDH is a major cause of childhood disability and accounts for up to 29% of total hip replacements in patients younger than 60 years [77] and is considered a main reason for arthritis in women under 40 [71].

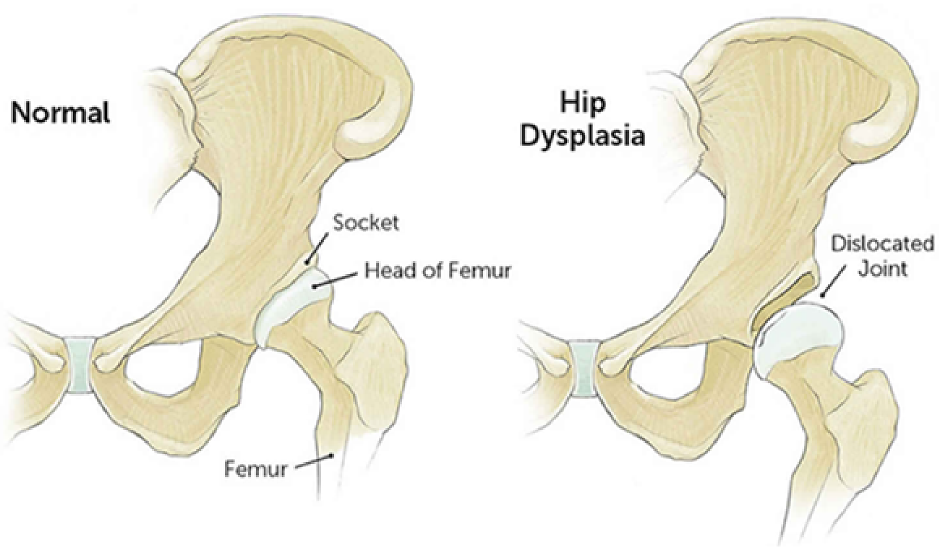


Figure 1 *Normal positioned hip (left) and dislocated hip (right) [52].*

2.1.1 Diagnosis and treatment of DDH

Early diagnosis and treatment is strongly associated with successful outcomes, as most cases can be treated with safe, cost-effective, and non-invasive methods [85]. In contrast, delayed or absent treatment increases the risk of chronic pain, gait abnormalities (e.g., limping), and early-onset osteoarthritis [56]. Diagnosis is typically established within the first year of life, most commonly through physical examination screening during routine checkups in early infancy. After the first few months, clinicians rely more on limited hip abduction and other clinical signs, as instability maneuvers become less reliable [3, 56]. If the examination is abnormal or if risk factors are present, imaging techniques, such as Magnetic Resonance Imaging (MRI), are obtained. Ultrasound is generally used up to about 4 months of age, and pelvic radiographs are preferred from about 4 months onward through childhood [3, 27]. In adolescents and adults, residual dysplasia (undiagnosed or untreated DDH) is usually diagnosed with plain radiographs, with MRI used to evaluate hip pathology in patients, often presenting with hip or groin pain, dysfunction, or even hip osteoarthritis [84].

Management of DDH follows a stepwise approach, using the least invasive option first and escalating to more invasive treatments. When detected early (under 6 months), the first-line treatment is dynamic abduction bracing, most commonly with the Pavlik harness, which positions the hips in flexion and controlled abduction to stimulate acetabular development [2]. If bracing fails or the diagnosis is delayed, CR under anesthesia is considered. CR involves gentle manipulation to relocate the femoral head, followed by hip-spica casting (HSC) to maintain reduction [89]. If concentric stability cannot be achieved or maintained, OR is required, involving surgical clearance of intra-articular obstacles. Similary, OR is followed by HSC, which is kept on a patient for usually from 3 to 12 months [77].

Although CR and OR are reported to be successful in up to 95% of cases, they carry important risks [26]. The most feared complications are redislocation of the hip and AVN of the femoral head, which is thought to result from compromised vascular supply [64]. Reported AVN rates after DDH treatment vary widely (approximately 6%–48%) [64] and is likely multifactorial (e.g., patient age, dislocation severity, reduction stability, immobilization protocol). Importantly, one of the prominent factors discussed in the literature is the in-cast hip position, particularly the HAA during HSC (**see Figure 2**) [68, 77, 89]. Postoperative CT/MRI enables accurate measurement of HAA, thus several MRI studies after CR and HSC have measured HAA (and other cast-position parameters) and assessed the link of HAA and mid-term hip development or later complications [21, 32, 37, 47, 66].

The position-dependent trade-off was first proposed by Ramsey et al. as the "safe zone," defined as the arc between the minimum abduction at which the hip remains reduced and the maximum comfortable abduction, thereby balancing insufficient abduction against excessive abduction [65]. The Ramsey safe-zone limits are commonly cited at roughly 35–40° to 55–60° of abduction [5, 47, 65]. Several studies are consistent with this concept and reported increasing AVN rates with larger in-spica abduction, with some data suggesting higher risk when HAA exceeds $\sim$55–60° [1, 22, 43, 50, 69]. Some authors recommend even lower targets in selected subgroups (e.g., limiting abduction to $< 50^\circ$ in very young infants) [69]. However, other cohorts have reported mixed or subgroup-dependent associations between HAA and AVN, highlighting that optimal angles may depend on age, stability requirements, and outcome definitions [31, 47]. Overall, the literature supports careful intraoperative assessment of the safe zone and deliberate control of HAA during HSC, with later studies suggesting target ranges for optimal HAA closer to 30–45° in certain settings [21, 32, 37].

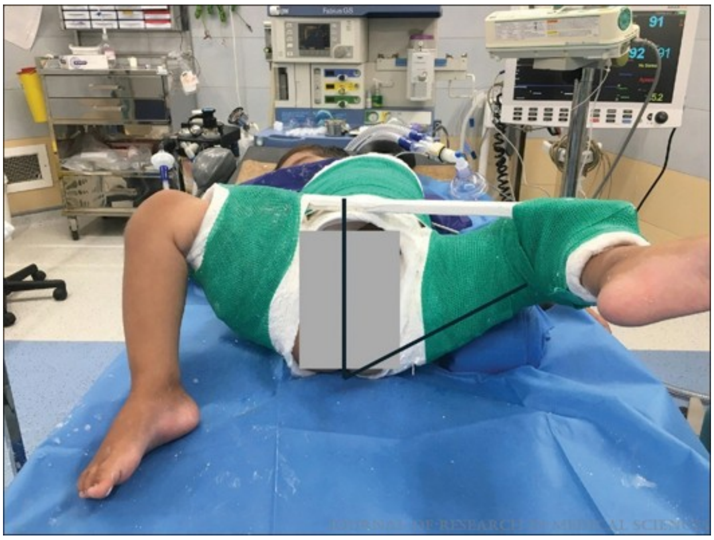


***Figure 2** HAA schematic presentation in the process of casting [43].*

2.1.2 Hip abduction angle measurement during DDH

Historically, HAA assessment has relied on two main approaches: mechanical goniometry and imaging techniques, such as MRI or CT. For example, Kheiri et. al documented quantifying HAA during HSC in the operating room using a handheld goniometer, noting that measurement is difficult for untrained assessors and is influenced by the amount of hip flexion [43]. Other studies also mention hip flexion as the critical factor for successful measurement of HAA. For positioning HSC, a 90–100° of hip flexion with neutral rotation is considered to be an optimal positioning strategy. Axial abduction estimates are considered valid primarily at 90° of flexion because at lower angles, illusion occurs in orthogonal imaging planes, therefore it could systematically overestimate abduction (**see Figure 3**) [4]. This geometric distortion inherent in 2D imaging suggests that any computer vision approach relying on 2D keypoints will likely suffer from similar projection biases unless corrected. This sets up Machine Learning (ML) correction layer as a necessity.

Gather et al. described a postoperative imaging protocol in which MRI is obtained immediately after CR and HSC application, and HAA is measured as part of the assessment [32]. While HAA is quantified objectively using cross-sectional imaging after casting, the literature around DDH reduction in a spica cast focuses primarily on confirming femoral head position (i.e., whether the hip is concentrically reduced) rather than providing real-time guidance for cast positioning or HAA adjustment [11, 25, 49]. In summary, postoperative MRI/CT/ultrasound methods are used to verify the femoral head location after the cast has been applied and imaging is obtained outside the operative workflow.

While both mechanical goniometry and advanced imaging techniques are considered as the established standards for reliable measurement, for the hip casting these tools cannot be utilized due to them being episodic snapshots rather than dynamic assessments [59]. The challenge to accurately assess HAA during HSC highlight the need for reliable, fast, and standardised evaluation tools. Existing clinical practice lacks a validated method to estimate hip position during casting, which motivates the development of a technological solution that supports timely, evidence-informed decision-making in a fast-paced casting environment and helps reduce procedural variability [39, 51].

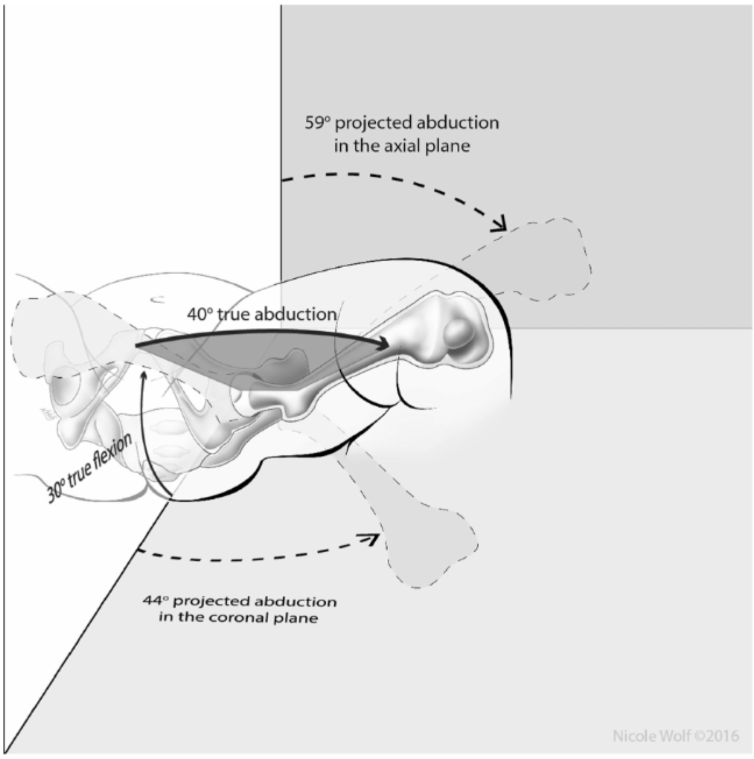


Figure 3 *Example of abduction measurement error due to imaging geometry. Despite the leg being in only 40° of true abduction, the angular deviation of the femoral axis from midline as seen on axial imaging is 59°. For this example, the traditional method for estimating HAA therefore overshoots the true value by 19° [4].*

2.2 Pose-estimation models: a novelty to include in clinical workflows

In research and specialized clinical settings, marker-based motion capture systems (e.g., Vicon, OptiTrack) are the established gold standard for capturing human movement in biomechanics, sports science, and clinical gait analysis [24, 55]. These systems measure 3D body motion by tracking reflective or active markers attached to specific anatomical landmarks on the body [24]. However, these systems require controlled laboratory conditions, specialized equipment, and technical expertise, limiting their accessibility and scalability [24, 55]. Consequently, a gap remains between high-accuracy laboratory methods and practical, point-of-care assessment tools that can be conveniently deployed in clinical environments [41, 79].

Recent advances in computer vision and deep learning have created scalable, markerless alternatives capable of estimating joint angles directly from monocular RGB video. These models detect anatomical keypoints, offering clinicians a portable, low-cost, and automated method for kinematic assessment[20, 79]. Particularly relevant are convolutional neural network (CNN)-based pose estimation models such as OpenPose [14], MB [7, 8], MoveNet [78], and HRNet [73]. These models estimate 2D keypoints and, in some pipelines, reconstruct approximate 3D kinematics from video using trained neural networks [8, 76]. Because they can be deployed with relatively simple hardware and support real-time analysis, they are increasingly explored for clinical, sports, and rehabilitation use. However, ongoing validation to determine whether performance matches or falls short of marker-based systems, continues [20, 79].

2.2.1 High-level review of the modern pose estimation models

Early computer vision in human motion analysis relied on traditional 2D image processing methods, which required manual feature extraction and were highly sensitive to lighting and occlusion [54, 63]. With the advancement of ML and in particular convolutional neural networks (CNNs) pose estimation has become significantly more robust, accurate, and accessible [57, 83]. Modern pose estimation automatically detect and track anatomical landmarks from images or video frames, producing a digital skeletal representation of the human body [14]. These models typically employ CNNs trained on large, annotated datasets to learn spatial relationships between body parts [57, 83]. A 2025 systematic review of vision-based gait analysis found that markerless pose algorithms can achieve good validity in healthy gait, though accuracy drops for certain joints and planes [20]. For example, OpenPose-based 2D gait analysis showed excellent agreement in temporal parameters (ICC $\sim$0.9–0.99) and reasonably low error in hip and knee flexion angles ($< 5^\circ$ MAE in sagittal plane) when compared to marker-based data [20].

Traditional pose estimation frameworks often rely on heatmap-based representations, where each joint is localized using a 2D probability distribution [74, 83]. In these heatmaps, pixel intensity reflects the likelihood of a joint being at that position [83]. Joint coordinates are then extracted by identifying the peak value or by applying soft-argmax techniques [48, 74]. While large CNN-based models like OpenPose offer high reliability and accuracy, they generate individual heatmaps for each joint and limb connection, which are then combined to reconstruct the full body skeleton, even in multi-person scenes [14]. However, this approach is computationally intensive and typically requires a graphics processing unit (GPU) to run efficiently [14]. In contrast, coordinate regression methods, like MoveNet, predict joint coordinates directly as numeric outputs without generating intermediate heatmaps [34]. While these models are faster and more efficient, they may sacrifice spatial accuracy, especially in the presence of occlusion or atypical poses, since they lack the rich spatial structure provided by the heatmaps [20, 76].

To balance accuracy and efficiency, hybrid approaches have emerged as a compelling solution [8]. These models use heatmaps during training to guide the network in learning spatial relationships, but switch to direct coordinate regression during inference for faster runtime [8]. This strategy, known as heatmap-guided regression, is implemented in MB [8]. MB achieves accuracy comparable to larger models (e.g. OpenPOse, AlphaPose) while maintaining a significantly lower computational footprint, making it suitable for deployment on mobile and web platforms [8, 36]. Additionally, its ability to operate fully on-device eliminates the need for cloud computation, improving privacy, reducing latency, and simplifying deployment [8, 36]. In contrast, cloud-based alternatives (e.g. Azure Vision) typically require more computational resources and raise concerns around data privacy [79]. As a result, lightweight models like MB are increasingly recognized as a practical and scalable solution for real-time, markerless motion analysis in healthcare, sports, and rehabilitation contexts [8, 79].

2.2.2 MB: framework

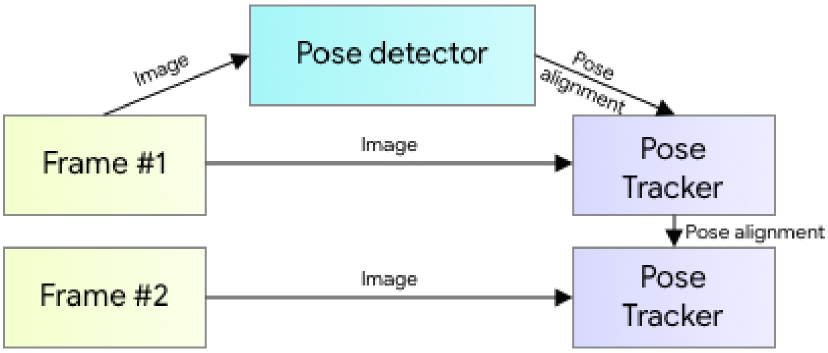


Figure 4

Google's MediaPipe framework provides an open-source, cross-platform pipeline for real-time multimedia processing. Within this framework, BlazePose is a pose estimation solution that employs a two-stage ML pipeline consisting of a detector followed by a tracker. In the first stage, a pose detector locates the person's region-of-interest (ROI) within the frame [8, 36]. In the second stage, a pose landmark tracker processes the cropped ROI to infer the full pose. For video input, this pipeline runs the detector on the first frame and then uses the previous frame's landmarks to update the ROI for subsequent frames, greatly improving efficiency (**see Figure 4**).

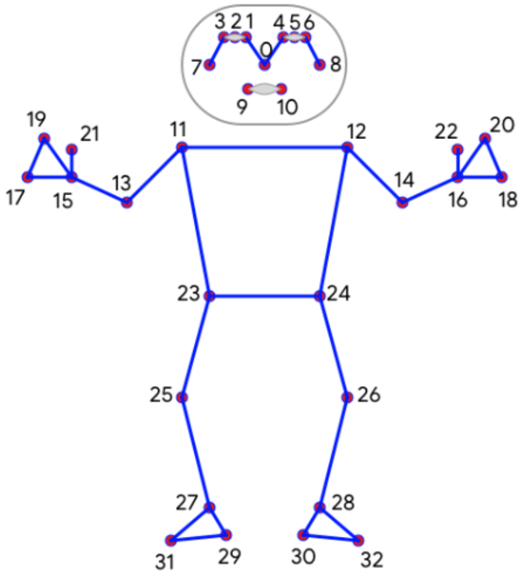


Figure 5 *MB 33 topological body and face landmarks [8].*

In the second stage, a pose-tracking network processes the cropped ROI and predicts the coordinates of 33 skeletal landmarks (**see Figure 5**), comprising the standard 17 body joints (COCO-style) plus additional facial and distal limb landmarks (e.g., eyes/ears/mouth and hand/foot extremities) [35, 36]. Each landmark is reported as (x,y,z,v), where (x,y) are image/ROI coordinates, z is a relative depth value, and v is a confidence measure (visibility) [35, 36]. The z axis is perpendicular to the camera plane and passes approximately through the midpoint between the hips; negative z values indicate points closer to the camera, while positive values indicate points farther away [33, 35]. Thus, from a single RGB camera the model produces a pseudo-3D pose representation (2D landmarks with relative depth), which can support downstream tasks such as pose analytics and augmented reality [35]. Importantly, the model directly regresses landmark coordinates in a single forward pass, enabling real-time or near–real-time operation on resource-constrained devices compared with heavier heatmap-based pipelines [8, 36].

2.2.3 MB: performance overview and limitations

Several studies have validated MB performance against marker-based systems [79, 81]. For example, Wang et al.'s study involving 25 healthy adults assessed full-body range of motion using MB with a standard RGB webcam. Participants performed a series of standing and dynamic movements involving the hip abduction. The results demonstrated high intra-session reliability, with intraclass correlation coefficients (ICC) exceeding 0.90 for most joints compared with a marker-based OptiTrack system. Specifically, hip abduction was performed and adduction achieved ICC ≈ 0.89, and inter-rater comparisons between MB and OptiTrack yielded ICC values of approximately 0.89–0.95, indicating strong agreement. Regression analyses further confirmed a good correspondence between the two systems, with $R^2 \approx 0.84$ for hip abduction angles [81].

Other studies have extended MB application to real-time fitness and rehabilitation systems, where joint angles are computed for exercise recognition and movement evaluation. For instance, one system employing MB with a k-nearest neighbors (KNN) classifier and dynamic time warping achieved 98.3% accuracy in classifying exercise quality, while operating approximately 13 times faster than OpenPose under comparable conditions. These results highlight MB suitability for continuous, real-time performance monitoring using standard consumer hardware [8, 38]. Additionally, quantitative analyses reveal that MB accuracy varies by joint and motion type [20, 79, 81]. For lower-limb movements such as hip abduction and flexion–extension, the model typically achieves intraclass correlation coefficients (ICC) of 0.85–0.95 relative to marker-based motion capture, indicating optimal reliability [81].

While MB struggles with upper-limb occlusion (ICC 0.18-0.53), likely due to self-occlusion and overlapping body parts, its performance peaks in lower-limb sagittal and frontal plane movements (ICC > 0.85), providing strong methodological support for its use in hip abduction specifically, despite its limitations elsewhere [20, 79]. In addition to the limitation to the lower-limb movements, MB model is designed for single-person tracking. In scenes with multiple individuals, it detects and tracks only one subject, potentially omitting relevant movements. Performance also declines when the head or torso is occluded, when the subject rotates substantially from the camera's frontal view, or when the camera is positioned at extreme angles [8, 79]. Environmental factors such as lighting, motion blur, and clothing contrast can also degrade performance. Studies have shown that dark, non-reflective clothing reduces segmentation accuracy, particularly when limbs overlap, while light-colored, form-fitting clothing yields more reliable landmark detection [20, 79, 81]. Similarly, reduced frame quality or poor illumination can introduce jitter and positional noise in the estimated landmarks [20].

A further limitation involves the depth estimation. While MB outputs a relative z coordinate for each landmark, this value is not metrically scaled, meaning absolute distances cannot be interpreted without calibration. The z component therefore serves primarily as a qualitative indicator of depth ordering rather than a quantitative spatial measure [35]. Finally, BlazePose's tracking stability can be affected by detection failures or confidence drops. When the detector loses the subject or confidence falls below threshold, the system may reinitialize, producing transient errors or frame-to-frame jitter [81]. Although newer implementations mitigate this through temporal smoothing and region-of-interest tracking, minor discontinuities remain possible during rapid or complex [35, 36, 79]. In summary, BlazePose offers real-time processing, is straightforward to use and deploy on mobile devices, and can achieve accuracy that is acceptable for clinical applications, making it a promising option for point-of-care motion assessment. Although its performance is strongly influenced by factors such as camera position, lighting conditions, clothing, and the complexity of the movement, it can be a very effective tool when these factors are well controlled [79].

3 Methodology

This section presents the architecture of the computer-vision tool for measuring pediatric hip abduction, the data-collection step used to evaluate the tool, and the analysis to determine whether a real-time application based on the MB framework can estimate HAA with sufficient accuracy for integration into clinical workflows.

3.1 System overview

A prototype system was developed to estimate HAA in real-time from monocular video. A high-level overview of the architecture is shown in Figure 6. The system comprises five main components:

1. **Video acquisition:** Input is provided as a live camera stream, built to work as a mobile application and supporting phone's GPU's.
2. **Pose estimation:** MB algorithm was selected as a suitable to detect full-body landmarks, including hip and knee joints, which are used for subsequent geometric calculations.
3. **Angle computation:** HAA is computed geometrically from the detected landmarks using hip midpoints and bilateral (midpoint–knee) vectors, producing a per-frame angle estimate.
4. **Signal stabilization an correction:** Optional temporal filtering and HAA correction methods are applied to reduce frame-to-frame noise, improve signal stability and correct output, if it is needed.
5. **Visualization and zone classification:** The vectors and landmarks are displayed on the real time video, portraying abduction pose. The HAA is displayed along with a color-coded zone classification based on pediatric orthopedic criteria (White, Green, Yellow, Red).

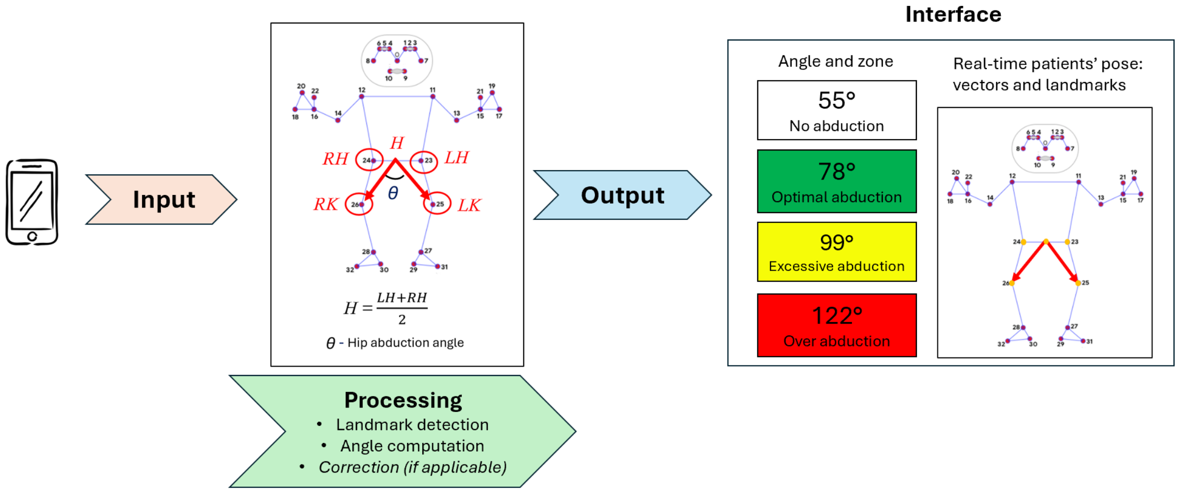
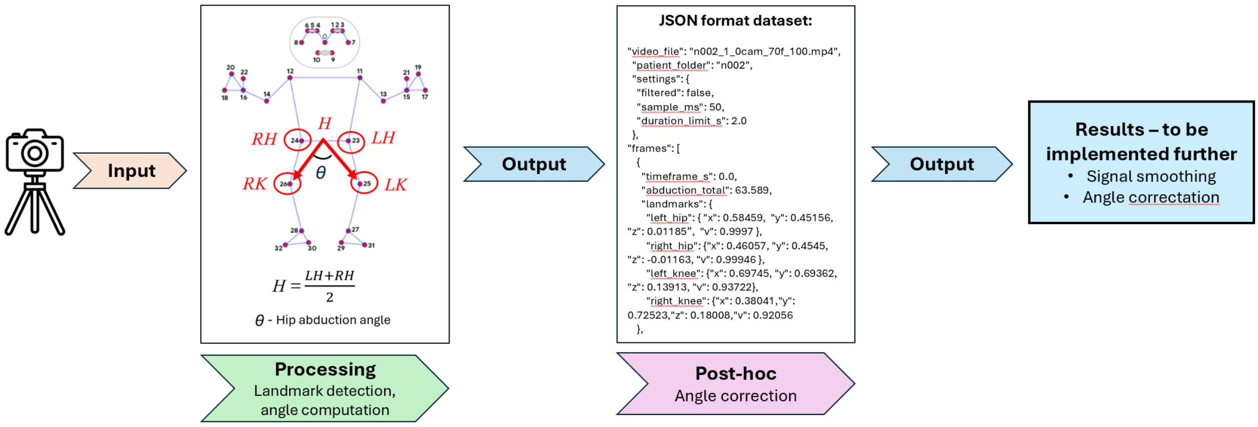


***Figure 6** Workflow of the proposed markerless HAA evaluation system.*

3.1.1 System's evaluation pipeline

The system was initially developed as a fully real-time application and deployed as a mobile-oriented web prototype (available at: https://ingagarsvaite.github.io/DDHtool/). For performance evaluation, an offline pipeline was also implemented in which pre-recorded videos were processed using the same pose-estimation and angle-computation steps as in the live system (**see Appendix C**). The resulting metrics were exported for statistical analysis and for training ML models to improve signal correction, for the resulting improvements to be subsequently integrated back into the real-time application. To collect data, a structured experimental protocol was designed to simulate HSC conditions in a controlled environment. Healthy adult volunteers held static hip-abduction poses at clinically relevant hip flexion and hip abduction angles while being recorded. Accordingly, as illustrated in Figure 7, the evaluation pipeline included pre-recorded video acquisition (1), pose estimation (MB) (2), geometric angle computation (3), data acquisition as an output for evaluation (4), and data analysis and signal's correction (5). The ML models and signal correction techniques will be implemented further in an online system's framework to acquire an appropriate signal.



```
"video_file": "n002_1_0cam_70f_100.mp4",
 "patient_folder": "n002",
 "settings": {
  "filtered": false,
  "sample_ms": 50,
  "duration_limit_s": 2.0
 },
"frames": [
  {
   "timeframe_s": 0.0,
   "abduction_total": 63.589,
   "landmarks": {
    "left_hip": { "x": 0.58459,  "y": 0.45156,
"z": 0.01185",  "v": 0.9997 },
    "right_hip": {"x": 0.46057, "y": 0.4545,
"z": -0.01163, "v": 0.99946 },
    "left_knee": {"x": 0.69745, "y": 0.69362,
"z": 0.13913, "v": 0.93722},
    "right_knee": {"x": 0.38041,"y":
0.72523,"z": 0.18008,"v": 0.92056
  },
```

Figure 7 *Offline's system architecure visualisation.*

3.2 Data collection

3.2.1 Participants

Eleven healthy adults were recruited for this pilot study. Participants reported no known musculoskeletal or neurological disorders and no hip pain or functional impairment. Close-fitting clothing was worn to ensure unobstructed visibility of anatomical landmarks during video recording.

The participants were informed that the study involved only movement assessment, without clinical intervention. All data collected were anonymized and securely stored; recorded videos were not intended for publication and only extracted numerical measures were used for subsequent analysis. Written informed consent was obtained from all volunteers prior to participation. Ethical approval was obtained from the Research Ethics Committee of the Faculty of Mathematics and Informatics at Vilnius University , and all methods were performed in accordance with Vilnius University guidelines and regulations.

3.2.2 Experimental variables: TRUE, `CAM` and FLEX

The experimental conditions were chosen to (i) span clinically meaningful abduction ranges used in pediatric HSC/abduction bracing, and (ii) stress-test robustness of the 2D angle estimation to plausible deviations in pose and viewpoint. Specifically, three total-abduction targets (TRUE$\{70, 100, 130\}^\circ$) were selected as representative values within the clinically defined zones: 70° (green/optimal), 100° (yellow/caution), and 130° (red/overabduction, elevated AVN risk). Two hip-flexion targets (FLEX $\{70, 90\}^\circ$) and two camera placements (`CAM` $\{0, 45\}^\circ$) were included to evaluate sensitivity to non-neutral flexion and oblique viewing, respectively. The nine recording configurations are explored in following subsections and summarised in Table 1.

Table 1 *Recording configurations used in the study.*

	FLEX ($^\circ$)	**`CAM` ($^\circ$)**	**TRUE ($^\circ$)**
1	90	0	70
2	90	0	100
3	90	0	130
4	70	0	70
5	70	0	100
6	70	0	130
7	90	45	70
8	90	45	100
9	90	45	130

Abduction definition and zone thresholds. To ensure consistency across studies, this thesis uses total abduction, defined as the included angle between the left and right thighs. Because much of the literature reports single-hip abduction, all published thresholds were converted to total abduction by doubling the reported values.

Using total abduction, four zones were defined:

- **White zone:** $< 60^\circ$ (insufficient abduction for HSC).
- **Green zone:** 60–90° (optimal range; corresponds to 30°–45° per hip).
- **Yellow zone:** 90–110° (cautionary interval; corresponds to 45°–55° per hip).
- **Red zone:** $> 110^\circ$ (overabduction; corresponds to $> 55^\circ$ per hip and has been associated with increased AVN risk). [65]

Hip flexion sensitivity. Hip abduction is commonly assessed with the hip flexed to approximately 90°; however, deviations from this standardized position can alter the apparent abduction angle measured in 2D, because the effective plane of motion rotates relative to the camera image plane as flexion changes (e.g., 30°–40°) has been reported to visually inflate abduction estimates [4] To evaluate the sensitivity of the proposed pipeline to non-neutral positioning, FLEX $= 70^\circ$ was selected as a clinically plausible deviation from the intended 90°. FLEX 70° and may occur unintentionally in practice, while still being sufficiently different to test whether reduced flexion changes the measured HAA and its MAE.

Camera viewpoints. Recordings were acquired from two camera placements: frontal (0°) and oblique ($\approx 45^\circ$) relative to the subject.

3.2.3 Experimental setup

A structured protocol of static hip-abduction poses was performed under controlled conditions. Participants lay supine on a flat surface with the hips flexed and the knees bent. articipants first set hip flexion (verified by goniometer), then adjusted bilateral abduction to match the template.

To standardise positioning, goniometer-calibrated cardboard templates were prepared for each abduction target (**see Figure 9**). For each recording, hip flexion was verified using a goniometer, after which participants aligned the lower limbs to the template and held the static posture for approximately 2 s while video was recorded. For each configuration, participants were instructed to:

1. Position the required hip-flexion angle (70° or 90°).

2. Adjust bilateral hip abduction to match the visual template for the target angle.

3. Hold the static posture for 2 s during recording.

Each configuration was repeated three times to ensure robustness. Videos were recorded using an iPhone 16 Pro (4K (3840×2160) at 60 FPS, 48 MP wide-angle sensor (f/1.78 aperture)) mounted on a tripod and 40 cm above the surface level to ensure stable imaging. For the frontal placement (0°), the camera was positioned 50 cm from the participant's head (along the cranio-caudal axis). For the oblique placement (45°), the angle was set using a goniometer and the camera was positioned 70 cm from the participant's head. Uniform indoor illumination was ensured using controlled artificial lighting to minimise shadows and reflections. An illustration of experimental setup, including position and tripod's position is presented in Figure 8.

3.3 System architecture

3.3.1 Video input acquisition and processing

Video acquisition was performed using OpenCV's VideoCapture [61]. For each recording, the native frame rate was extracted and used to compute a uniform sampling interval of 50 ms. Even though all recordings were captured using the same settings, this ensured that frames were processed at consistent temporal spacing regardless of the original FPS. Each sampled frame was converted from BGR to RGB format and passed directly into the pose-estimation module.

3.3.2 Pose estimation with MB

Pose estimation was performed using the MediaPipe Solutions API, which implements the MB model [8]. The pose estimator outputs 33 pseudo-3D anatomical landmarks, each represented by

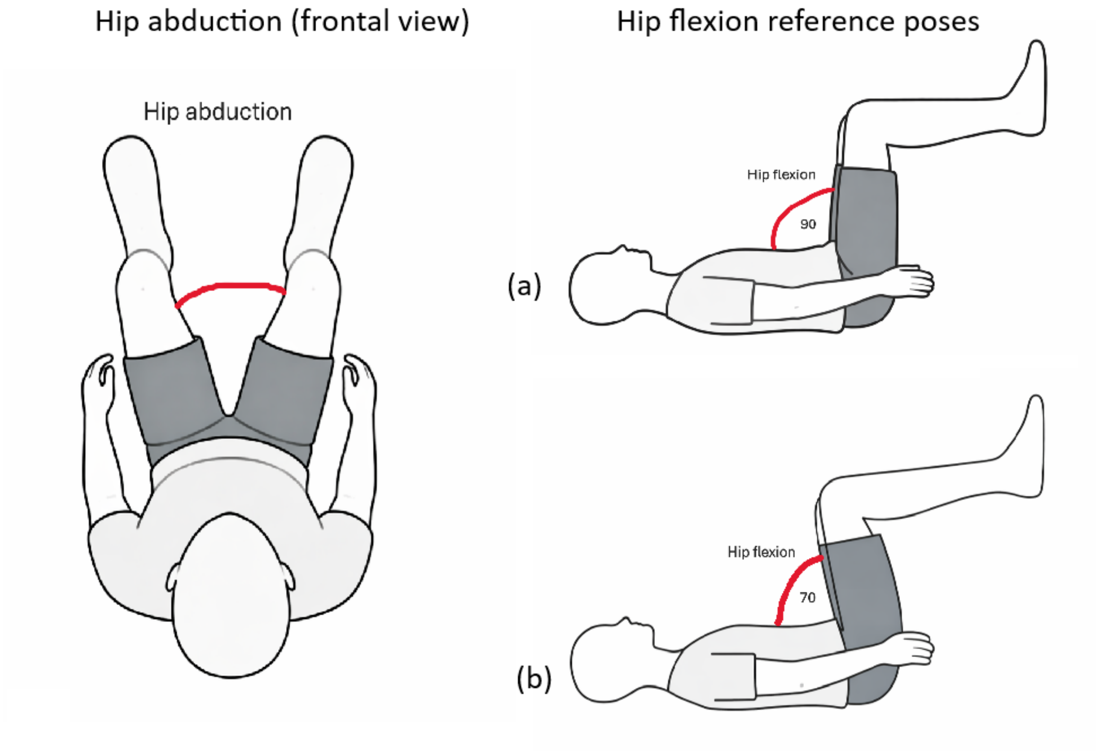


__Figure 8__ Experimental recording setup for structured hip-abduction data acquisition (supine position; hips flexed; knees bent; bilateral abduction). Left image shows the bilateral hip-abduction position used for all configurations; on the right (a) and (b) positions represents FLEX configurations (70°, 90°).

normalized coordinates (x, y, z) and a visibility score (v) indicating detection confidence. For each sampled frame, the system extracted only the landmarks relevant to hip-abduction computation, as illustrated in Figure 8:

- Left Hip (LH): 23
- Right Hip (RH): 24
- Left Knee (LK): 25
- Right Knee (RK): 26

Frames with invalid or missing landmarks were automatically discarded to ensure angles were computed only from anatomically valid detections.

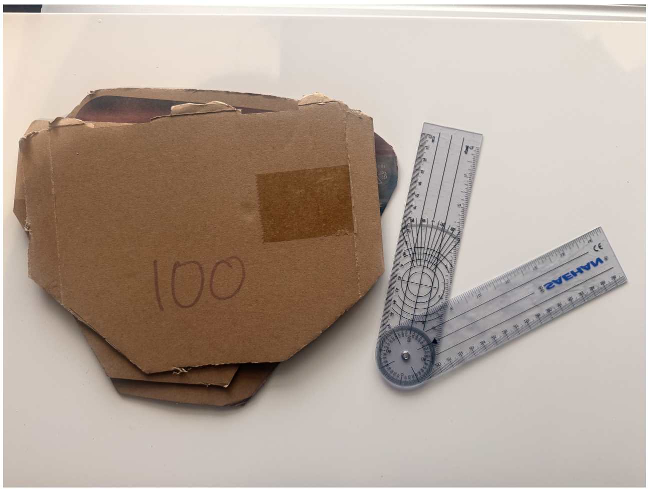
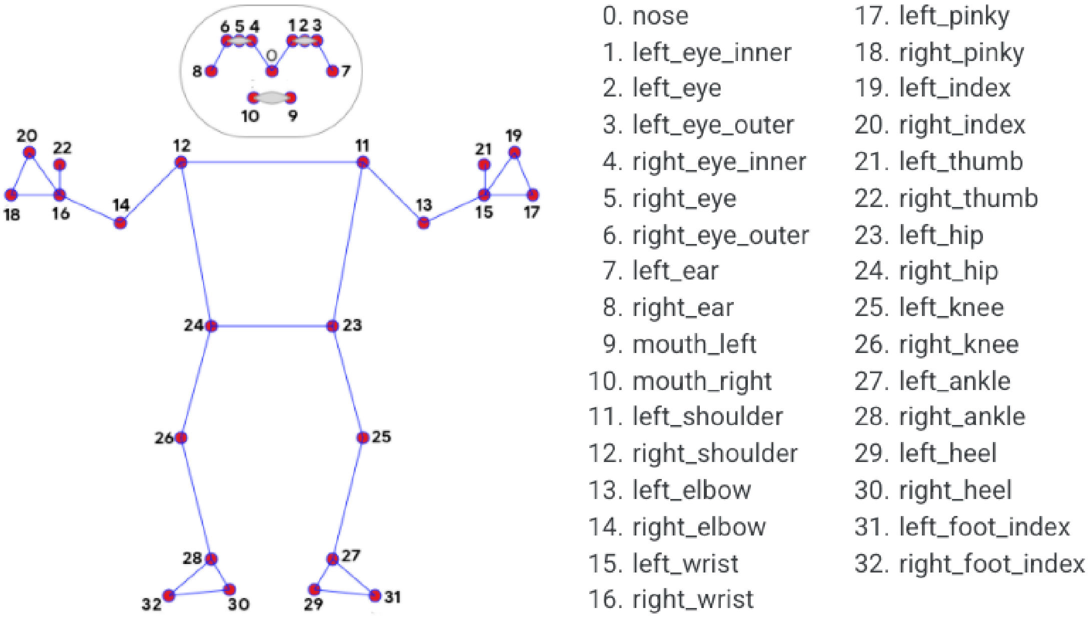


***Figure 9** Tools used in the experiment: goniometer-calibrated cardboard templates and goniometer*



0. nose
1. left_eye_inner
2. left_eye
3. left_eye_outer
4. right_eye_inner
5. right_eye
6. right_eye_outer
7. left_ear
8. right_ear
9. mouth_left
10. mouth_right
11. left_shoulder
12. right_shoulder
13. left_elbow
14. right_elbow
15. left_wrist
16. right_wrist
17. left_pinky
18. right_pinky
19. left_index
20. right_index
21. left_thumb
22. right_thumb
23. left_hip
24. right_hip
25. left_knee
26. right_knee
27. left_ankle
28. right_ankle
29. left_heel
30. right_heel
31. left_foot_index
32. right_foot_index

***Figure 10** MB architecture's 33 body landmarks [36].*

3.3.3 Angle estimation from the landmarks coordinates

Although MB outputs pseudo-3D landmark coordinates (x,y,z), monocular depth estimates (z) are typically less stable than in-plane coordinates and may introduce jitter in downstream kinematic computations [35]. Therefore, to ensure robustness and evaluate MAE across different camera conditions, hip abduction was computed using a 2D formulation based on the image-plane coordinates (x,y) across all configurations. In addition, abduction was quantified as a total (inter-thigh) abduction angle, defined as the included angle between the left and right thigh vectors in the 2D plane (**see Figure 11**).

All landmarks were expressed in 2D image coordinates as (x,y) points:

$$RH = (x_{RH}, y_{RH}), \quad LH = (x_{LH}, y_{LH}), \quad RK = (x_{RK}, y_{RK}), \quad LK = (x_{LK}, y_{LK}).$$

The pelvic midpoint H was defined as:

$$H = \frac{LH + RH}{2}. \tag{1}$$

Thigh vectors were defined from the knees to the pelvic midpoint:

$$\vec{v}_L = H - LK, \tag{2}$$

$$\vec{v}_R = H - RK. \tag{3}$$

For any 2D vector $\vec{a} = (a_x, a_y)$, the magnitude is:

$$|\vec{a}| = \sqrt{a_x^2 + a_y^2}. \tag{4}$$

The dot product of two vectors $\vec{a}$ and $\vec{b}$ is:

$$\vec{a} \cdot \vec{b} = a_x b_x + a_y b_y. \tag{5}$$

The total abduction (inter-thigh) angle was computed as:

$$\theta = \text{arccos}\left(\frac{\vec{v}_L \cdot \vec{v}_R}{|\vec{v}_L|\,|\vec{v}_R|}\right). \tag{6}$$

A schematic depiction of landmarks and vectors is provided in Figure 11.

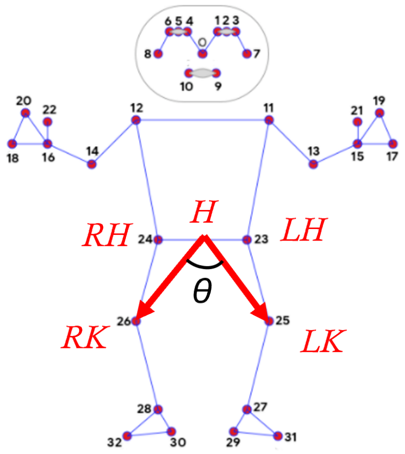


Figure 11 *Schematic presentation of the vectors in 2D plane calculated from the MB landmarks*

3.4 Statistical analysis

Python (version 3.13.5, 64-bit) [29] was used for video processing and data preparation, including landmark extraction, filtering, One Euro filter (OEF) parameter tuning, and noise reduction and analysis. All remaining statistical analyses and visualisations were performed in R (version 4.5.2, 64-bit) [75]. Video clips were trimmed to fixed-duration segments using Microsoft Clipchamp [53]. ChatGPT (models GPT-5, GPT-5.1 and GPT-5.2) [60] was used as an auxiliary tool during the study for scientific literature searching and code suggestions. The tool was treated as a helper source and was not used for information retrieval and text generation, but rather wording suggestions and overall editing. Outputs were reviewed by the author.

Continuous variables are summarised as mean$\pm$SD; when distributions were skewed, median (IQR) is additionally reported. Unless stated otherwise, statistical significance was assessed at $\alpha = 0.05$ (two-tailed).

3.4.1 Single-trial metrics

For each trial (single video), the pipeline produced a frame-wise hip abduction angle time series $\{\theta_{r,t}\}_{t=1}^{T_r}$ (degrees), where r denotes the trial and T_r is the number of retained frames after preprocessing. The target abduction angle for the corresponding configuration is denoted θ_{true}.

A single trial-level angle estimate was computed as the mean over frames,

$$\bar{\theta}_r = \frac{1}{T_r} \sum_{t=1}^{T_r} \theta_{r,t}. \tag{7}$$

Similarly, the trial-level mean visibility was computed by first averaging the visibility scores of the four landmarks used for HAA estimation (LH, RH, LK, RK) within each frame, and then averaging this frame-wise mean over the T_r retained frames of trial r. Let $v_{r,t,\ell}$ denote the visibility of landmark $\ell \in \{\text{LH,RH,LK,RK}\}$ at frame t. The frame-wise mean visibility is

$$v_{r,t} = \frac{1}{4} \sum_{\ell \in \{\text{LH,RH,LK,RK}\}} v_{r,t,\ell}, \tag{8}$$

and the trial-level mean visibility is

$$\bar{v}_r = \frac{1}{T_r} \sum_{t=1}^{T_r} v_{r,t}. \tag{9}$$

Trial-level accuracy was quantified using mean absolute error (MAE) and root mean squared error (RMSE) across frames,

$$\text{MAE}_r = \frac{1}{T_r} \sum_{t=1}^{T_r} \left| \theta_{r,t} - \theta_{\text{true}} \right|, \tag{10}$$

$$\text{RMSE}_r = \sqrt{\frac{1}{T_r} \sum_{t=1}^{T_r} \left(\theta_{r,t} - \theta_{\text{true}} \right)^2}. \tag{11}$$

MAE and RMSE are standard measures of average prediction error magnitude and error with increased sensitivity to larger deviations, respectively [18, 86].

Short-term frame-to-frame jitter (signal smoothness) was summarised using the mean absolute first difference:

$$\text{Noise}_r = \frac{1}{T_r - 1} \sum_{t=2}^{T_r} |\theta_{r,t} - \theta_{r,t-1}| \,. \tag{12}$$

.

3.4.2 Participant-condition metrics

Each participant performed repeated trials under the same participant-condition (CAM × FLEX × TRUE). To give each repeated trial equal weight, participant–condition summaries were computed as the mean across repeated trials [30, 70]. For a participant-condition with R repeated trials, following metrics were calculated:

$$\bar{\theta}_{\mathsf{p}} = \frac{1}{R}\sum_{r=1}^{R} \bar{\theta}_r, \quad \text{MAE}_{\mathsf{p}} = \frac{1}{R}\sum_{r=1}^{R} \text{MAE}_r, \quad \bar{v}_{\mathsf{p}} = \frac{1}{R}\sum_{r=1}^{R} \bar{v}_r \quad . \tag{13}$$

These participant-condition summary metrics were then used as the primary outcomes for subsequent statistical analyses, including Analysis of Variance (ANOVA), and as input/target variables in the machine-learning (ML) models.

3.4.3 Configuration-level evaluation across participants

For each experimental condition evaluation, participant-condition values were aggregated across participants n. For a participant-level metric $m_{\mathsf{p},j}$ (MAE$_{\mathsf{p}}$ or $\bar{\theta}_{\mathsf{p}}$), the configuration-level mean and sample standard deviation were computed as

$$\bar{m_{\mathsf{c}}} = \frac{1}{n}\sum_{j=1}^{n} m_{\mathsf{p},j}, \qquad s_{m,c} = \sqrt{\frac{1}{n-1}\sum_{j=1}^{n} (m_{\mathsf{p},j} - \bar{m_c})^2}. \tag{14}$$

Condition-level MAE was reported as $\overline{\text{MAE}}_c \pm s_{\text{MAE},c}$ and median $\widetilde{\text{MAE}}_c$ (together with interquartile range IQR$_c$.

3.4.4 Evaluation of the MAE

To assess the effects of experimental configuration and participant characteristics on estimation error, inferential analyses were conducted using both parametric and non-parametric methods, depending on the distributional properties of the outcomes.

Configuration effects on estimation error were analysed using linear mixed-effects models. The dependent variable was the log-transformed mean absolute error, which was used to stabilise variance and improve residual normality. Fixed effects modelling included variables presented in the Table 2.

A participant-specific random intercept was included to account for repeated measurements within participants [6, 45].

***Table 2** Variables used in statistical analyses.*

Short name	Name	Role	Type	Values / scenarios
MAE	Mean absolute error	Dependent variable	Numerical	Degrees; $\log(1 + \text{MAE})$
TRUE	Target abduction angle	Primary predictor	Numerical	$70°, 100°, 130°$
CAM	Camera angle	Predictor	Factor	$0°$ (ref), $45°$
FLEX	Hip flexion angle	Predictor	Factor	$70°$ (ref), $90°$
height	Height	Covariate	Numerical	Continuous (cm)
weight	Weight	Covariate	Numerical	Continuous (kg)
age	Age	Covariate (extended models)	Numerical	Continuous (years)
gender	Gender	Covariate (extended models)	Factor	Male (ref), Female
–	Interactions	Derived predictors	Derived	$\texttt{TRUE} \times \texttt{CAM}, \texttt{TRUE} \times \texttt{FLEX}$

Models were fitted by maximum likelihood, and fixed-effect significance was assessed using Type III ANOVA with Satterthwaite-approximated degrees of freedom, using the `lme4` and `lmerTest` packages. Model selection was based on information criteria (AIC and BIC).

Normality of residuals was assessed using residual diagnostics and formal tests where appropriate. When residual distributions deviated substantially from normality, residuals transformation or non-parametric methods were used for inference.

3.4.5 Evaluation of Noise

To assess the effect of temporal filtering, noise was computed separately for the raw and filtered signals within each trial, yielding a paired noise difference with negative values indicating reduced noise after filtering. Each trial was treated as a single observational unit.

Because the distribution of paired noise differences exhibited strong deviations from normality, statistical inference was performed using a one-sided Wilcoxon signed-rank test with the alternative hypothesis $\Delta_r < 0$. To avoid reliance on asymptotic assumptions, p-values were obtained via a permutation-based implementation of the signed-rank test, in which the null distribution was generated by random sign flips of the paired differences [58, 87].

Analyses were conducted separately for each combination of target abduction angle and camera–flexion configuration. To account for multiple comparisons across conditions, p-values were adjusted using the Benjamini–Hochberg false discovery rate (FDR) correction. Effect size was quantified using the rank-biserial correlation derived from the signed-rank statistic, where more negative values indicate greater consistency of noise reduction across trials.

3.4.6 Temporal stabilisation - applying OEF

Frame-wise pose estimates from monocular video typically exhibit high-frequency jitter [88]. To reduce this noise while preserving responsiveness to genuine motion, all HAA time series were temporally smoothed using the adaptive OEF (1€ filter), a first-order low-pass filter whose effective cutoff frequency increases with signal speed, balancing jitter suppression and lag [17]. OEF has also been used in practice to stabilise noisy tracking and kinematic signals in real-time interaction and motion-control settings [15, 46].

OEF was chosen because it is designed for real-time use and is computationally lightweight. Importantly, this choice is also aligned with the pose-estimation framework: MediaPipe provides built-in landmark smoothing options, including a OEF, and it is described as a commonly used smoothing approach across MediaPipe solutions [8]. Unlike fixed low-pass filters, the OEF adapts its effective cutoff based on signal dynamics, enabling a controllable trade-off between responsiveness and smoothness.

Let x_t denote the raw angle at frame t and $\hat{x}_t$ the filtered angle. The filter performs recursive exponential smoothing,

$$\hat{x}_t = \alpha_t x_t + (1 - \alpha_t)\hat{x}_{t-1}, \tag{15}$$

where the smoothing factor α_t is derived from an adaptive cutoff frequency $f_c(t)$. Following Casiez et. al methodology, [17], the cutoff is set to a baseline value that increases with the (smoothed) magnitude of the signal velocity,

$$f_c(t) = f_{\text{min}} + \beta\,|\widehat{\dot{x}_t}|. \tag{16}$$

Here, f_{min} controls the amount of smoothing during quasi-static periods, and β controls how quickly the filter becomes more responsive during faster movements. The derivative smoothing cutoff ($d_{\_cutoff}$) was held constant.

Filter parameters were selected empirically via a grid search, following the tuning guidance of the 1€ filter [16, 17]. Candidate values $f_{\text{min}} \in (0, 4]$ Hz and $\beta \in [0, 1]$ were evaluated on the recorded trials; for each (f_{min},β) pair, the filtered signal was assessed using (i) RMSE with respect to the reference angle and (ii) a residual-jitter proxy computed as the mean absolute first difference. The parameter pair minimising the combined score,

$$J(f_{\text{min}},\beta) = 0.5\overline{\text{RMSE}} + 0.5\overline{\text{Noise}}, \tag{17}$$

was selected for further analysis. RMSE and the noise metric are defined in Section 3.4.1.

3.4.7 Prediction tasks and evaluation protocol

To correct the system output (estimated HAA), two supervised learning tasks were considered: (i) **classification** of the discrete target zone (ZONE), defined from the target abduction angle θ_{true} according to Table 3; and (ii) **regression** to predict the corrected continuous target angle (TRUE).

Table 3 *Mapping of target abduction angles (TRUE) to assessed clinical zone.*

TRUE (°)	**ZONE**
70	Green
100	Yellow
130	Red

To reflect realistic availability of inputs under different operational scenarios, four feature-set configurations were evaluated. System-derived inputs were the participant-condition estimated angle $\bar{\theta}_{\mathrm{p}}$ and the mean visibility $\bar{v}_{\mathrm{p}}$. Experimental factors CAM and FLEX were also included in selected feature sets to allow assessment of whether acquisition settings provide additional predictive value.

Table 4 *Feature-set configurations evaluated in the machine-learning experiments.*

Feature set	Variables included
All features	$\bar{\theta}_{\mathrm{p}}$, CAM, FLEX, $\bar{v}_p$
CAM–FLEX	$\bar{\theta}_{\mathrm{p}}$, CAM, FLEX
Estimation + vis	$\bar{\theta}_{\mathrm{p}}$, $\bar{v}_p$
Estimation only	$\bar{\theta}_{\mathrm{p}}$

To assess generalisation to unseen participants, all prediction models were evaluated using leave-one-participant-out (LOPO) cross-validation: in each fold, all observations from one participant were held out for testing and the remaining participants were used for training [12]. While this protocol is well-suited to small datasets, it explicitly evaluates patient-level generalization, making it sensitive to participant-specific shifts in feature distributions.

For **classification** (ZONE), the following model families were evaluated: a majority-class baseline (ZeroR) [82], multinomial logistic regression (MLR) [10], an RBF-kernel support vector machine (SVM), Random Forest (RF) [13], XGBoost (XGB) [19]. Classification performance was assessed using accuracy, macro-averaged F1 [72], and macro-averaged one-vs-rest ROC-AUC [28, 67]. The full LOPO procedures, model families, and hyperparameters are summarised in 1 algorithm. Confusion matrices were computed by pooling out-of-fold predictions across all LOPO folds, yielding a single cross-validated confusion matrix that reflects performance on unseen participants.

1 algorithm LOPO classification for ZONE

```
Input: dataset $D$ with patient_id, target ZONE, and predefined feature set $\mathbb{F}$ (presented in Table 4)
Output: mean accuracy, macro-F1, macro-AUC across LOPO folds
Set $\mathbb{F} \leftarrow \mathbb{F} \cap \text{columns}(D)$
for each held-out participant $p$ in unique(patient_id) do
    Split: $D_{\text{train}} \leftarrow D[\texttt{patient\_id} \neq p]$, $D_{\text{test}} \leftarrow D[\texttt{patient\_id} = p]$
    Keep complete cases for $\mathbb{F}$ and true_zone in each split
    Define factor levels of $y_{\text{test}}$ to match $y_{\text{train}}$; skip fold if training has $< 2$ classes
    ZeroR: predict majority class from $y_{\text{train}}$
    RF: Random Forest with ntree=500 [13]
    XGB: XGBoost [19] with objective=multi:softprob, max_depth=4, $\eta$=0.1, subsample=0.8, colsample_bytree=0.8, nrounds=200
    MLR: multinomial logistic regression (softmax); standardise numeric predictors using training mean/SD; maxit=500
    SVM: RBF-kernel SVM with cost=1 and probability estimates enabled [23]
    For XGB, one-hot encode predictors using training levels and align test columns
    For each model: compute accuracy, macro-F1 [72], and macro-AUC (one-vs-rest) [28, 67] for participant $p$
end for
Return the mean of each metric across LOPO folds
```

For **regression** models to determine TRUE, ordinary least squares linear regression (LM), RBF-kernel support vector regression (SVR) [23], Random Forest (RF) [13], XGBoost (XGB) [19] were evaluated. Regression was evaluated using MAE, RMSE, and R^2 [40]. The full LOPO procedures, model families, and hyperparameters are summarised in 2 algorithm.

2 algorithm LOPO regression for TRUE

```
**Input:** dataset $D$ with patient_id, target TRUE, and predefined feature set $\mathbb{F}$ (presented in Table 4)
**Output:** mean MAE, RMSE, $R^2$ across LOPO folds
Set $F \leftarrow F \cap$ columns$(D)$
Keep complete cases for {patient_id, angle_true} $\cup F$
**for** each held-out participant $p$ in unique(patient_id) **do**
    Split: $D_{\text{train}} \leftarrow D[\texttt{patient\_id} \neq p]$, $D_{\text{test}} \leftarrow D[\texttt{patient\_id} = p]$; **skip fold** if $|D_{\text{train}}| < 5$ or $|D_{\text{test}}| < 1$
    Extract $(X_{\text{train}}, y_{\text{train}})$ and $(X_{\text{test}}, y_{\text{test}})$ using $\mathbb{F}$
    **LM:** ordinary least squares regression
    **RF:** Random Forest with ntree=500 [13]
    **XGB:** XGBoost [19] with objective=reg:squarederror, max_depth=4, $\eta$=0.1, subsample=0.8, colsample_bytree=0.8, nrounds=200
    **SVR:** RBF-kernel $\varepsilon$-SVR with cost=1 and $\varepsilon = 0.1$ [23]
    For matrix-based models (RF/XGB/LGB), one-hot encode predictors using training levels and align test columns
    For each model: predict $\hat{y}$ on $X_{\text{test}}$ and compute MAE, RMSE, $R^2$ [40] for participant $p$
**end for**
Return the mean of each metric across LOPO folds
```

The model families were selected to compare interpretable linear baselines with flexible non-linear learners suited to low-dimensional tabular predictors. LM and MLR were included as transparent reference models. To capture potential non-linearities and interactions without manual feature engineering, tree-ensemble methods (RF, XGB) were evaluated [13, 19]. Finally, RBF-kernel SVM/SVR were included as margin-based kernel methods that can model smooth non-linear relationships in moderate-sample settings [23, 40].

4 Results

This section provides an overview of the collected data (Section 4.1), assess statistical significance of main and interaction effects (Section 4.2), analyzes noise to reduce jitter (Section 4.3), and finally applies machine learning models to calibrate values (Sections 4.4–4.5)

4.1 Exploratory data analysis

4.1.1 Dataset structure and notation

In total of 11 healthy adults (5 males and 6 females; age: 26.4 $\pm$ 3.5 years; height: 176.1 $\pm$ 9.4 cm; weight: 69.3 $\pm$ 10.4 kg; bmi: 22.4 $\pm$ 2.4 kg/m^2) participated in the experiment. Each participant performed 2 s of a static pose hold in 9 different configurations, 3 times each, yielding 297 videos in total. Videos were processed using the pipeline described in Section 3.3 and sampled every 50 ms (20 Hz), producing a time series of 40 frames per trial. For each frame t, the processing pipeline outputs the 3D coordinates (x, y, z) and visibility $v_J(t) \in [0,1]$ for each landmark J (LK, RK, LH, RH), as well as the calculated total HAA $\hat{\theta}(t)$ (see 3.3.3). The resulting outputs were exported for subsequent analyses. Following subsections 3.4.1 - 3.4.3, data were further reduced into multiple representations; each reduction level was used for a different analysis described in the corresponding subsection.

4.1.2 Error distribution across configurations

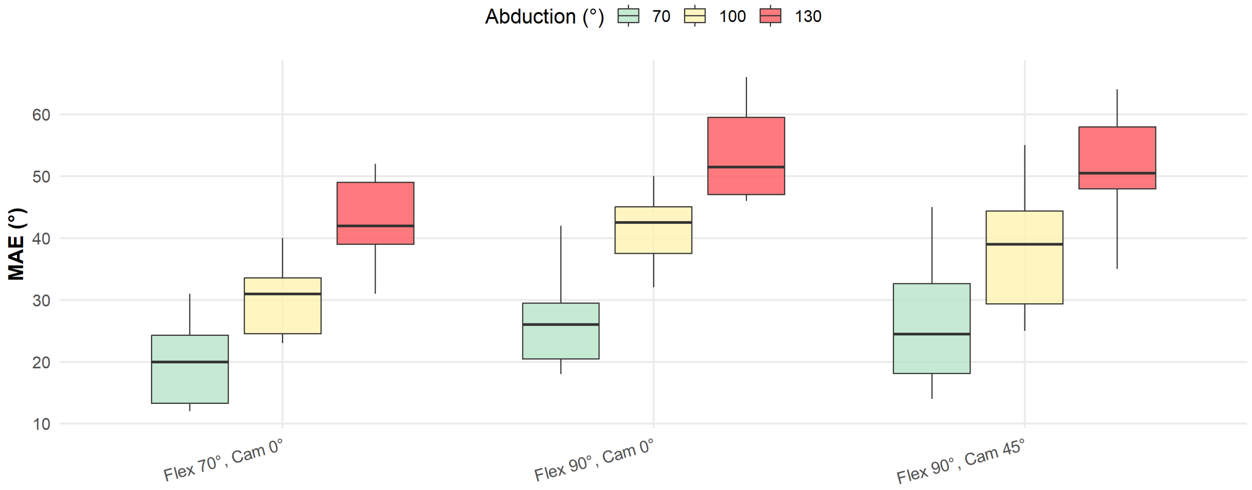


Figure 12 *MAE distributions across configurations and target abduction angles (box = IQR, centre line = median, whiskers = minimum–maximum)*

Table 5 summarises the MAE distributions by configuration (FLEX $\in$ $\{70°, 90°\}$, CAM $\in$ $\{0°, 45°\}$, and target abduction TRUE $\in \{70°, 100°, 130°\}$). Across all configurations, MAE increases with larger target angles, indicating a systematic degradation in estimation accuracy at higher abduction. Dispersion also generally increases with abduction: for example, at FLEX $= 70°$ and CAM $= 0°$, SD rises from 7.85° at TRUE $= 70^\circ$ to 13.03° at TRUE $= 130^\circ$, and the IQR widens accordingly (see Table 5). In addition, both SD and IQR tend to be higher when CAM $= 45^\circ$, suggesting less consistent performance under an oblique viewpoint.

Across all participants and configurations, the raw pose-based pipeline produced substantial absolute errors that are too large for direct clinical use. As shown in Table 5 and Figure 17, typical errors are around 20° at lower abduction (TRUE $= 70^\circ$) and exceed 50° at higher abduction (TRUE $= 130^\circ$), with consistent increases in both central tendency (mean/median) and dispersion (SD/IQR) as TRUE increases. Similar patterns across FLEX$\times$ `CAM` settings indicate that the error is not configuration-specific, but instead reflects a systematic limitation of the underlying geometric estimation approach. In Figure 17, boxes represent the IQR (Q1–Q3) with the median shown as the centre line, and whiskers denote the minimum and maximum MAE.

***Table 5** Configuration-level summaries of estimated HAA and absolute error compared to the reference. Values are reported in degrees. MAE is reported as mean$\pm$SD and as median (Q1–Q3).*

`CAM` (°)	FLEX (°)	TRUE (°)	MAE (mean$\pm$SD)	MAE (median (Q1–Q3))
0	70	70	20.21$\pm$7.85	20.02 (13.31–24.31)
0	70	100	36.54$\pm$9.51	31.12 (24.56–33.56)
0	70	130	47.57$\pm$13.03	42.65 (39.00–50.10)
0	90	70	27.16$\pm$6.89	25.79 (20.47–29.47)
0	90	100	43.20$\pm$5.87	42.54 (37.54–45.03)
0	90	130	52.12$\pm$8.01	51.57 (47.05–61.55)
45	90	70	28.42$\pm$9.26	24.70 (18.15–32.65)
45	90	100	42.18$\pm$8.89	39.08 (29.37–44.36)
45	90	130	54.13$\pm$10.50	50.53 (47.98–62.98)

Note: The reference TRUE angles are also subject to measurement error.

4.1.3 Agreement analysis (Bland–Altman)

Figure 13 shows a Bland–Altman plot at the participant–condition level ($N = 99$), where each point corresponds to one participant under one configuration. The plot shows the difference $(\bar{\theta}_{\text{p}} - \theta_{\text{true}})$ against the mean $\left(\frac{\bar{\theta}_{\text{p}}+\theta_{\text{true}}}{2}\right)$, where θ_{true} denotes the reference angle for the corresponding configuration. A clear negative bias is observed: all 99 observed points lie below zero, indicating systematic underestimation of the true abduction angle in this pipeline. In addition, the magnitude of underestimation increases with angle (proportional bias), as the cloud shifts downward for larger mean angles. Several outliers are also present, suggesting occasional failures (e.g., landmark misdetection or temporary occlusion).

4.1.4 Zone-based classification and confusion matrix

Following the methodology in Section 3.2.2, each estimated angle $\bar{\theta}_{\text{p}}$ was assigned to a corresponding clinical zone.

Figure 14 shows the resulting confusion matrix across zones. The matrix exhibits a pronounced downward shift in predicted zones, consistent with the previously presented error patterns. In par-

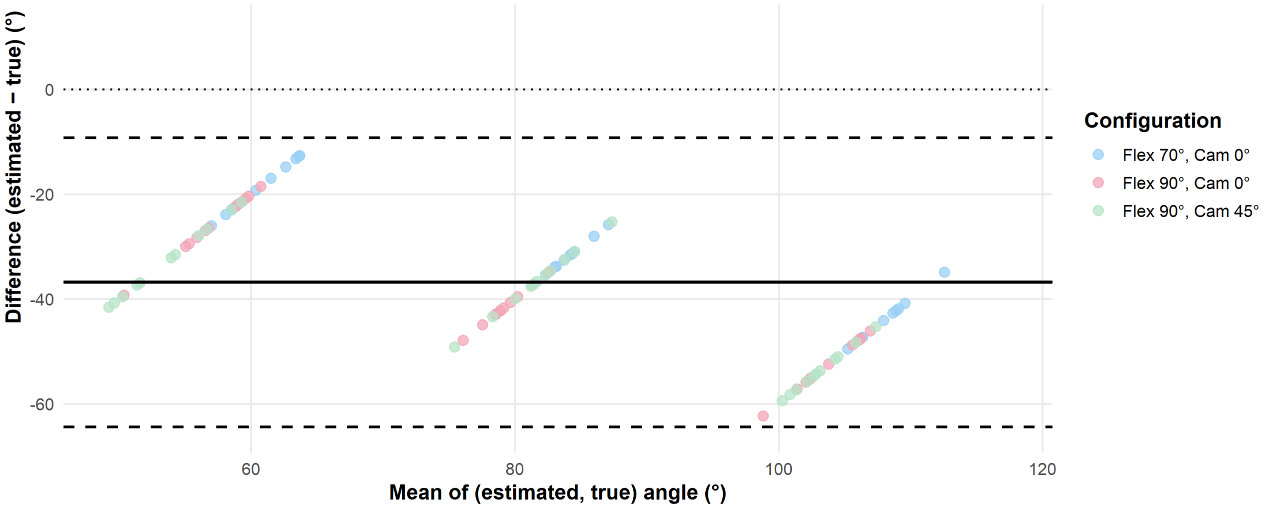
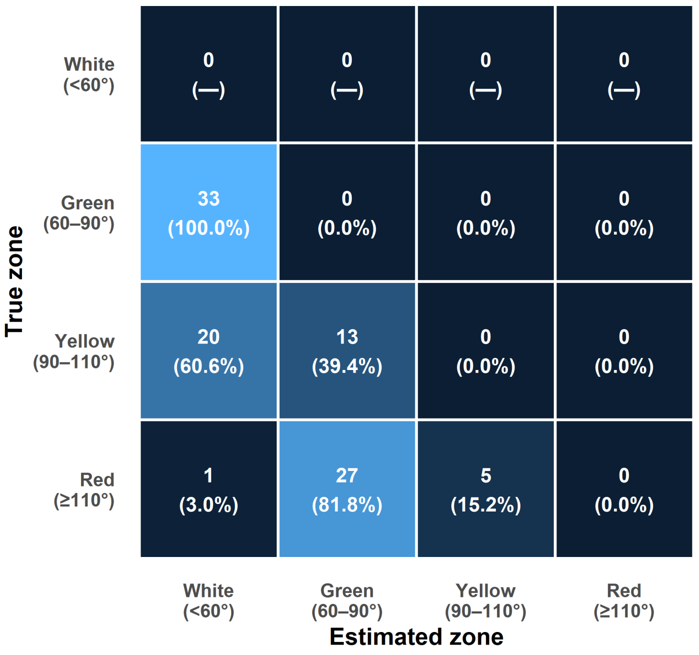


Figure 13 *Bland–Altman plot: difference* $(\bar{\theta}_{\mathrm{p}} - \theta_{\mathrm{true}})$ *against the mean* $\left(\frac{\bar{\theta}_{\mathrm{p}}+\theta_{\mathrm{true}}}{2}\right)$*. Colors denote measurement configuration*

ticular: (i) true Green cases are predicted as White; (ii) true Yellow cases are split between White and Green; (iii) true Red cases are mostly predicted as Green (and occasionally as Yellow or White); and (iv) there are no samples in the true White class in this dataset subset, and the model does not predict Red at all. Across the 99 samples, no correct classifications were observed for this dataset subset.



True zone	White (<60°)	Green (60–90°)	Yellow (90–110°)	Red (≥110°)
White (<60°)	0 (—)	0 (—)	0 (—)	0 (—)
Green (60–90°)	33 (100.0%)	0 (0.0%)	0 (0.0%)	0 (0.0%)
Yellow (90–110°)	20 (60.6%)	13 (39.4%)	0 (0.0%)	0 (0.0%)
Red (≥110°)	1 (3.0%)	27 (81.8%)	5 (15.2%)	0 (0.0%)

Figure 14 *Confusion matrix: true zone vs estimated zone counts. The dominant errors correspond to underestimation (prediction in a lower-risk zone).*

4.2 Mixed-effects modelling

4.2.1 Outcome transformation and model set

Residual diagnostics on the MAE scale indicated departures from the Gaussian error assumption and heteroscedasticity. To stabilise variance and improve residual normality, MAE was transformed using a log$(1 + \cdot)$ mapping (log1p), which is well-defined at zero:

$$\texttt{log1p\_mae} = \log(1 + \texttt{mae})\,. \tag{18}$$

All linear mixed-effects models used `log1p_mae` as the dependent variable and included a participant-specific random intercept. Models were fitted by maximum likelihood on the same dataset ($N = 99$ observations; 11 participants) and compared using AIC/BIC.

Across the candidate model set, a consistent overall tendency was observed: TRUE strongly explained increases in error on the log scale, and both `CAM` and `FLEX` contributed additional, angle-dependent effects. Demographic covariates (`age`, `gender`) did not improve fit or reach statistical significance in any explored specification, and adding the visibility metric (`v`) likewise did not improve model fit (see Annex B).

***Table 6** Model comparison for linear mixed-effects models predicting `log1p_mae`. Lower AIC/BIC indicate better fit after penalising model complexity.*

Model	Fixed effects	AIC	BIC	logLik	df.resid
m5	`TRUE`	-19.2	-9.2	13.6	86
m0	`CAM` + `FLEX`) × `TRUE`	-43.2	**-23.2**	29.6	82
m1	m0 + `gender`	-42.3	-19.8	30.1	81
m2	m1 + `age`	-40.5	-15.5	30.2	80
m4	m0 + `height`	-44.9	-22.4	31.4	81
m6	m0 + `height` + `weight`	**-47.9**	-22.9	34.0	80
m3	m2 + `height` + `weight`	-47.6	-17.6	35.8	78
m7	m6 + `v`	-46.0	-18.5	34.0	79
m9	m4 + `v`	-43.4	-18.4	31.7	80
m8	m3 + `v`	-46.3	-13.8	36.1	77

Notes: `TRUE` denotes the target abduction angle; `CAM` and `FLEX` denote camera viewpoint and hip-flexion setting, respectively. All models were fitted as linear mixed-effects models with the same random-effects structure, and differ only in their fixed-effects specification. The selected fixed-effects specification includes interactions of `TRUE` with `CAM` and `FLEX`, but does not include a `CAM` × `FLEX` interaction term. Boldface indicates the best (lowest) value in each criterion.

4.2.2 Baseline technical model (m0)

Model m0 included only the experimental (technical) predictors and served as a parsimonious reference specification. Under BIC, which penalises model complexity more strongly, m0 achieved the best score in the candidate set (BIC $= -23.2$; Table 6). Compared with the simplest baseline m5 (`TRUE` only; BIC $= -9.2$), incorporating `CAM`, `FLEX`, and their interactions with `TRUE` improved BIC by

14.0 points and increased log-likelihood from 13.6 to 29.6, indicating that the experimental factors capture substantial systematic variation in `log1p_mae` beyond the effect of `TRUE` alone.

4.2.3 Extended covariate model (m3)

Model m3 extended the technical predictors by adding demographics and anthropometrics (`gender`, `age`, `height`, `weight`). In terms of AIC, m3 improved over m0 (AIC -47.6 vs. -43.2; ΔAIC $= -4.4$) and achieved the highest log-likelihood in the candidate set (35.8). However, this added complexity was not supported under BIC ($= -17.6$, compared with m0 -23.2), and `gender` and `age` were not statistically supported in subsequent inference, suggesting limited additional explanatory value of these covariates in this dataset.

4.2.4 Selected model (m6): best overall fit and effect interpretation

Model m6 was selected as the primary mixed-effects specification because it achieved the **lowest AIC** (AIC $= -47.9$) and the **second-lowest BIC** (BIC $= -22.9$; Table 6). Relative to m0, adding `height` and `weight` improved AIC by 4.7 points (from -43.2 to -47.9) while only slightly worsening BIC by 0.3 points (from -23.2 to -22.9), indicating a favourable trade-off between improved fit and parsimony. Compared with m3, m6 achieved a marginally better AIC (by 0.3 points) with fewer fixed-effect terms, as `gender` and `age` were excluded. Adding `v` did not improve model selection: m7 (m6+`v`) increased AIC to -46.0 and BIC to -18.5, and m9 (m4+`v`) similarly worsened both criteria relative to m4. The final m6 specification is given in Equation 19:

$$\texttt{log1p\_mae} \sim (\texttt{CAM} + \texttt{FLEX}) * \texttt{TRUE} + \texttt{height} + \texttt{weight} + (1|\texttt{patient\_id}). \tag{19}$$

Table 7 *Selected model (m6): fixed-effect estimates and Type III tests for `log1p_mae` (Satterthwaite df).*

Term	**Estimate**	**SE**	**df**	t	p	**Type III** $\mathbb{F}$ **(**p**)**
Intercept	0.3240	0.5940	10.403	0.545	0.597	—
CAM_45	0.1790	0.0850	80.000	2.117	0.037*	4.48 (0.037)
FLEX_90	0.3030	0.0850	80.000	3.575	0.001***	12.78 (0.001)
TRUE	0.0125	0.0008	80.000	15.223	$< 0.001^{***}$	285.07 (< 0.001)
height	0.0156	0.0042	10.000	3.716	0.004**	13.81 (0.004)
weight	0.0110	0.0043	10.000	2.565	0.028*	6.58 (0.028)
CAM_45:TRUE	0.00170	0.00082	80.000	2.065	0.042*	4.27 (0.042)
FLEX_90:TRUE	0.00199	0.00082	80.000	2.418	0.018*	5.85 (0.018)

Notes: The response is log$(1 + \text{MAE})$. Reference levels are `CAM` $= 0°$ and `FLEX` $= 70°$. `height` is measured in cm and `weight` in kg. Type III tests are reported as F statistics with Satterthwaite degrees of freedom. Significance codes: $^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$.

4.3 Global noise reduction analysis

4.3.1 Global parameter tuning

To reduce frame-to-frame jitter while preserving estimation accuracy, a global grid search was performed to tune the OEF parameters across all available trials (see Equation 17). The search varied the smoothing coefficient $\beta \in [0{,}1]$ and the minimum cutoff $f_{\text{min}} \in [0{,}4.0]$, and selected the parameter combination that minimised a joint criterion based on signal noise and RMSE. The resulting heatmaps (Appendix A) indicated an optimum at $\beta = 0$ and $f_{\text{min}} = 0.026$. These parameters were then integrated into the processing pipeline, after which landmark coordinates and the derived abduction angles were recomputed.

4.3.2 Global effect of filtering on frame-to-frame jitter

To quantify the effect of filtering on signal stability, a per-trial noise metric was computed for both the raw and filtered abduction-angle time series (see Equation 12). For each trial, raw and filtered noise values formed a paired observation, and the paired difference was defined as where negative values indicate reduced noise after filtering. The paired noise differences were strongly non-normal (Jarque–Bera = 1012.25, $p \approx 1.56 \times 10^{-220}$). Therefore, a nonparametric one-sided Wilcoxon signed-rank permutation test was used to test whether filtering reduced noise (H_1 : Noise$_{\text{filtered}}$ $<$ Noise$_{\text{raw}}$).

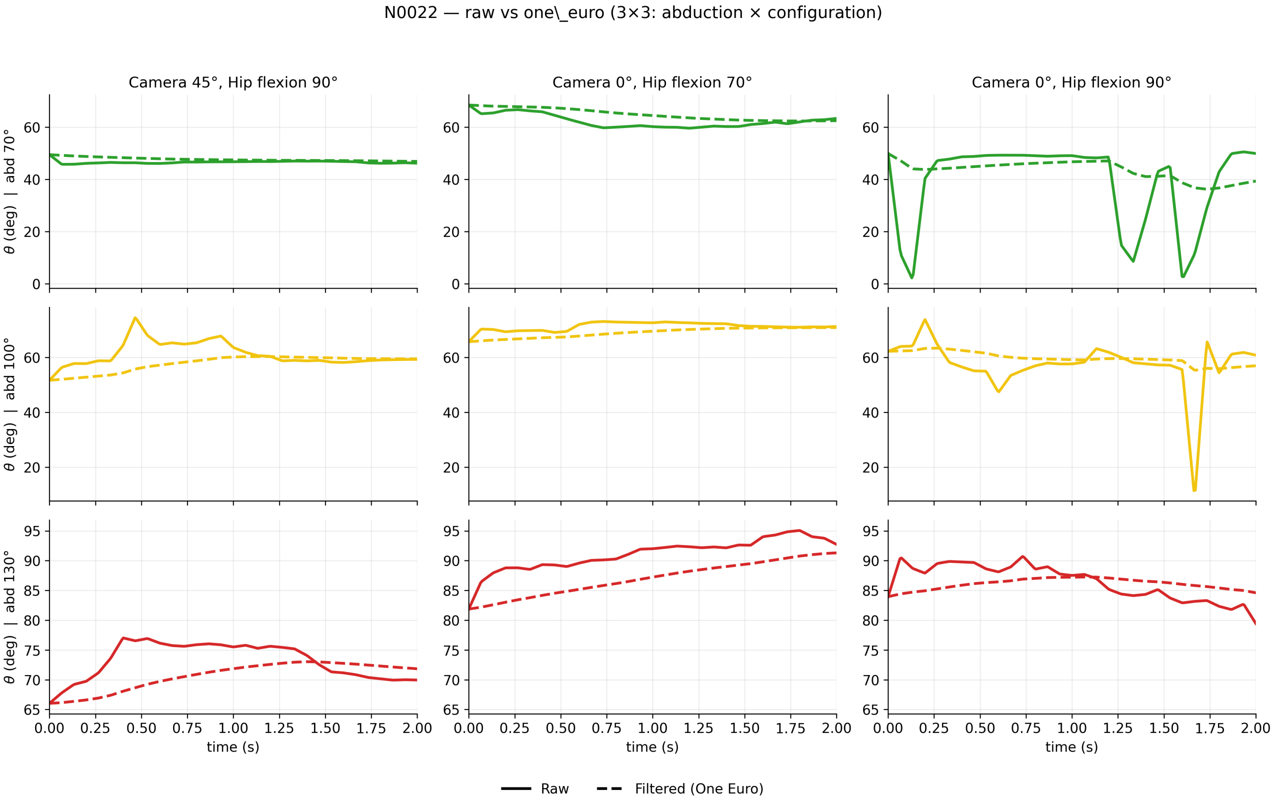


Figure 15 *Noise comparison (RAW vs OEF) across configurations and target angles per patient.*

Across all included paired trials ($n = 297$), filtering produced a statistically significant reduc-

***Table 8** Global (pooled) trial-level noise reduction across all included paired trials ($n = 297$). Noise is the mean absolute first difference of the estimated angle (deg/frame). $\Delta =$ Noise$_{\text{filtered}} -$ Noise$_{\text{raw}}$ (negative indicates reduced jitter).*

n	**Median noise (raw)**	**Median noise (filt.)**	**Median Δ**	**Median % red.**	p **(one-sided)**	r_{rb}
297	2.494	0.377	-2.118	83.46	< 0.001	-1

Note: p is from a one-sided Wilcoxon signed-rank permutation test ($H_1 : \Delta < 0$). r_{rb} denotes the rank-biserial correlation derived from the signed-rank statistic.

tion in noise (one-sided Wilcoxon; $p < 0.001$). Median noise decreased from 2.494 (raw) to 0.377 (filtered), with a median paired change of -2.118. The median relative reduction was 83.46%. The rank-biserial correlation was -1, indicating that noise decreased in 100% of trials, i.e., the improvement in temporal smoothness was consistent across all paired observations rather than being driven by a subset of trials.

For completeness, condition-wise analyses were also performed for each target abduction angle × camera–flexion configuration. In that setting, p-values from the groupwise Wilcoxon tests were adjusted using the Benjamini–Hochberg false discovery rate (FDR) procedure, yielding q-values; all groupwise comparisons remained significant after adjustment. Median percentage reductions ranged from 75.86% to 86.08%, demonstrating robust improvements in temporal smoothness across tested abduction angles and camera–flexion configurations. An example of noise reduction across configurations for one participant is shown in **??**.

4.4 Classification performance

4.4.1 All features

Using the **All features** set (see Table 9), all learning-based models substantially outperformed the majority baseline (accuracy 0.333). The best overall performance was achieved by **SVM** (accuracy 0.889; macro-F1 0.909; macro-AUC 0.947). **XGB** and **MLR** also performed strongly (accuracy 0.818 and 0.798), while **RF** was moderate (accuracy 0.737).

***Table 9** LOPO classification results using **All features**.*

Model	Mean Accuracy	Mean Macro-F1	Mean Macro-AUC
Baseline (Majority)	0.333	0.500	–
MLR	0.798	0.800	0.934
RF	0.737	0.770	0.903
XGB	0.818	0.821	0.924
SVM	0.889	0.909	0.947

4.4.2 CAM–FLEX

Using the **CAM–FLEX** set (see Table 10) yielded consistently high performance. **SVM** again achieved the best overall results (accuracy 0.909; macro-F1 0.908; macro-AUC 0.946). **XGB** achieved accuracy 0.818 (macro-F1 0.823), and **MLR** and **RF** performed similarly (accuracy 0.798 and 0.788).

Table 10 *LOPO classification results using* ***CAM–FLEX****.*

Model	Mean Accuracy	Mean Macro-F1	Mean Macro-AUC
Baseline (Majority)	0.333	0.500	–
MLR	0.798	0.789	0.939
RF	0.788	0.796	0.941
XGB	0.818	0.823	0.928
SVM	0.909	0.908	0.946

4.4.3 Estimation only

Restricting inputs to **Estimation only** (see Table 11) reduced performance relative to **CAM–FLEX** and **All features**. **SVM** remained strongest (accuracy 0.838; macro-F1 0.852). **XGB** achieved accuracy 0.768, while **MLR** and **RF** achieved 0.758.

Table 11 *LOPO classification results using* ***Estimation only****.*

Model	Mean Accuracy	Mean Macro-F1	Mean Macro-AUC
Baseline (Majority)	0.333	0.500	–
MLR	0.758	0.748	0.936
RF	0.758	0.763	0.916
XGB	0.768	0.760	0.926
SVM	0.838	0.852	0.909

4.4.4 Estimation + vis

Adding visibility (see Table 12) did not yield consistent gains under LOPO. **MLR** improved slightly in accuracy (0.778 vs. 0.758), while **SVM** and **XGB** were similar to (or slightly below) their estimation-only performance.

Table 12 *LOPO classification results using* ***Estimation + vis****.*

Model	Mean Accuracy	Mean Macro-F1	Mean Macro-AUC
Baseline (Majority)	0.333	0.500	–
MLR	0.778	0.781	0.931
RF	0.768	0.784	0.895
XGB	0.768	0.769	0.928
SVM	0.847	0.871	0.938

4.4.5 Summary across feature sets

Across all feature configurations, all learning-based models substantially exceeded the majority baseline (accuracy 0.333), confirming that the extracted features contain discriminative information for zone prediction. Performance was highest when acquisition-context variables were included (**CAM–FLEX** and **All features**). The best overall configuration was **SVM with CAM–FLEX**, achieving **0.909 mean accuracy** (**90.9%**), macro-F1 0.908, and macro-AUC 0.946 (Table 10). Using **Estimation only** reduced performance, while **Estimation + vis** did not provide consistent gains under LOPO.

4.4.6 Confusion matrix (best model)

Figure 16 shows the aggregated LOPO confusion matrix for the best-performing classifier (**SVM** with **CAM–FLEX**). The Green zone is identified perfectly (33/33, 100%). The Red zone is classified reliably (30/33, 90.9%), with the remaining errors confined to the adjacent Yellow zone (3/33, 9.1%). The Yellow zone is the most challenging: most samples are correctly classified as Yellow (27/33, 81.8%), with residual confusion primarily toward Green (4/33, 12.1%) and less frequently toward Red (2/33, 6.1%).

For comparison, the original (uncorrected) zone assignment from the raw geometric estimates (**??**) exhibits a pronounced collapse toward lower-risk predictions: all Green samples are mapped to White (33/33, 100%), Yellow samples are split between White and Green (60.6% and 39.4%), and Red samples are almost never predicted as Red (mostly mapped to Green, 81.8%, and Yellow, 15.2%). Relative to this baseline behaviour, the learned SVM CAM–FLEX model markedly reduces severe misclassification and concentrates the remaining errors near the intermediate Yellow boundary.

4.5 Regression performance

4.5.1 All features

Using the **All features** set (see Table 13), **SVM** achieved the best overall performance (MAE = 9.30°, RMSE = 12.16°, R^2 = 0.674). **XGB** was similar in MAE (9.41°) but slightly weaker by R^2 (0.654). **LM** remained competitive (R^2 = 0.645), while **RF** performed markedly worse (MAE = 15.33°, R^2 = 0.487).

Table 13 *LOPO regression results using* ***All features****.*

Model	Mean MAE	Mean RMSE	Mean R^2
LM	10.765	13.148	0.645
RF	15.333	17.373	0.487
XGB	9.410	13.000	0.654
SVM	9.295	12.161	0.674

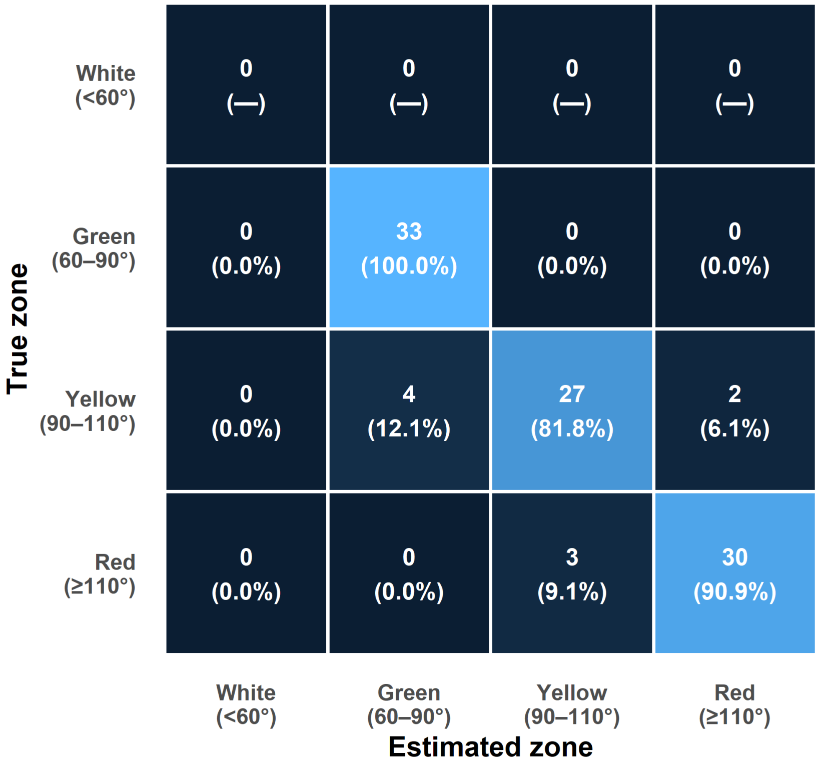


***Figure 16** Aggregated LOPO confusion matrix for **SVM** using **CAM–FLEX**.*

4.5.2 CAM–FLEX

The **CAM–FLEX** set (see Table 14) produced the strongest and most stable LOPO regression performance. **SVM** achieved the best results across metrics (MAE = 8.21°, RMSE = 11.03°, R^2 = 0.713). **XGB** achieved a comparable MAE (8.98°) but a substantially higher RMSE (14.25°), suggesting larger errors for a subset of held-out participants. **LM** remained stable (R^2 = 0.669), while **RF** again showed weak generalisation.

***Table 14** LOPO regression results using **CAM–FLEX**.*

Model	Mean MAE	Mean RMSE	Mean R^2
LM	10.271	12.751	0.669
RF	15.499	17.508	0.482
XGB	8.983	14.251	0.586
SVM	8.209	11.029	0.713

4.5.3 Estimation only

Restricting inputs to **Estimation only** (see Table 15) reduced performance relative to **CAM–FLEX**. Nevertheless, **SVM** remained strong (MAE = 8.92°, R^2 = 0.675), and **LM** achieved a similar R^2 (0.668), indicating that a substantial fraction of predictive signal is already contained in $\bar{\theta}_{\mathrm{p}}$. In this low-dimensional setting, **RF** improved (R^2 = 0.597), while **XGB** showed weak fit by R^2 (0.437), suggesting LOPO sensitivity in this configuration.

Table 15 *LOPO regression results using* ***Estimation only****.*

Model	Mean MAE	Mean RMSE	Mean R^2
LM	10.614	13.079	0.668
RF	9.509	13.953	0.597
XGB	9.266	15.878	0.437
SVM	8.922	12.339	0.675

4.5.4 Estimation + vis

Adding visibility (see Table 16) did not yield consistent improvements. Performance degraded for **LM** and especially **SVM** (SVM: MAE = 10.84°, R^2 = 0.595). In contrast, **RF** improved in this setting (R^2 = 0.680), indicating that $\bar{v}_p$ may be informative primarily for tree-based models under LOPO.

Table 16 *LOPO regression results using* ***Estimation + vis****.*

Model	Mean MAE	Mean RMSE	Mean R^2
LM	11.069	13.425	0.646
RF	9.588	12.279	0.680
XGB	12.147	15.242	0.546
SVM	10.836	14.100	0.595

4.5.5 Effect of OEF (CAM–FLEX, LOPO)

As an additional analysis, the **CAM–FLEX** set was evaluated on the **OEF-filtered** dataset to test whether reducing frame-level jitter improves participant-level generalisation. As summarised in Table 17, OEF did not improve LOPO regression: all models shifted toward slightly worse metrics (higher MAE/RMSE and lower R^2). For example, **SVM** changed from MAE = 8.21°, R^2 = 0.713 (RAW) to MAE = 8.40°, R^2 = 0.672 (OEF), while **LM** degraded more strongly (R^2 from 0.669 to 0.532).

Table 17 CAM–FLEX *regression performance under LOPO, with and without OEF.*

Model	**OEF**	$\overline{\textbf{MAE}}$ (deg)	$\overline{\textbf{RMSE}}$ (deg)	$\overline{\mathbf{R^2}}$
LM	FALSE	10.27	12.75	0.669
LM	TRUE	11.37	14.47	0.532
RF	FALSE	15.50	17.51	0.482
RF	TRUE	16.78	17.94	0.474
XGB	FALSE	8.98	14.25	0.586
XGB	TRUE	9.76	16.24	0.536
SVM	FALSE	8.21	11.03	0.713
SVM	TRUE	8.40	11.53	0.672

Note: OEF indicates whether the input time series was smoothed using the One Euro filter prior to computing participant–condition summaries.

4.5.6 Key findings (LOPO)

Across feature sets, meaningful continuous-angle prediction was achievable: the best-performing models reached MAE in the $\sim$8–9° range with $R^2 > 0.70$, representing a major improvement over the raw geometric pipeline. In the raw estimates, configuration-level mean MAE ranged from 20.21° at $($CAM $= 0°,$ FLEX $= 70°,$ TRUE $= 70°)$ up to 54.13° at $($CAM $= 45°,$ FLEX $= 90°,$ TRUE $= 130°)$ (see Table 5). By comparison, the best LOPO regressor (**SVM** with **CAM–FLEX**) achieved MAE = 8.21°, corresponding to an approximate 60% reduction relative to the lowest raw MAE (20.21° $\rightarrow$ 8.21°) and an $\sim$85% reduction relative to the highest raw MAE (54.13° $\rightarrow$ 8.21°).

Consistent with the classification results, **CAM–FLEX** yielded the strongest and most stable LOPO performance (see Table 14). Within this setting, the **SVM** regressor achieved the best overall metrics (MAE = 8.21°, RMSE = 11.03°, R^2 = 0.713).

Model behaviour differed across metrics. **LM** performed competitively ($R^2 \approx$ 0.65–0.67) across several settings, suggesting that once configuration variables are included, much of the correction is approximately linear. **XGB** occasionally achieved low MAE (e.g., 8.98° with **CAM–FLEX**), but with higher RMSE and lower R^2, indicating larger errors for some held-out participants. **RF** showed the weakest LOPO generalisation in higher-dimensional feature sets, but improved when inputs were restricted (e.g., **Estimation only**). Finally, adding visibility (**Estimation + vis**) did not provide consistent benefits for regression: it tended to reduce performance for **LM** and **SVM**, while **RF** improved in that specific setting (see Table 16), indicating that $\bar{v}_p$ is a model-dependent and LOPO-sensitive signal.

5 Discussion

This thesis evaluated the feasibility of estimating HAA from monocular video using a lightweight, real-time, markerless pipeline based on MB. The prototype system extracted a set of 2D landmarks and computed a bilateral HAA using a midpoint-anchored geometric formulation. Prior work has established that markerless pose-estimation frameworks (e.g., MediaPipe, OpenPose) can recover joint kinematics such as abduction, but most studies have focused on upper-limb abduction or gait-related hip motion, typically varying camera viewpoint and using limited ML correction [44, 81].

To the author's knowledge, this is the first study to evaluate HAA estimation in the context of HSC.Given the small, relatively homogeneous sample and the absence of pediatric participants and intraoperative conditions, the findings should be interpreted as proof of concept rather than as validation of a clinical measurement system. In particular, effect sizes and model performance estimates may change in larger and more diverse cohorts and under additional sources of variability (e.g., occlusion, motion, casting materials, and lighting). Moreover, the reference (ground-truth) angle measurements (TRUE) may contain error, which should be considered when interpreting discrepancies between estimated and measured angles.

5.1 Systematic error in HAA estimation

The posture constraints of the experiment primarily explain the large MAE. In this pipeline, total abduction is defined as the angle between two vectors originating at the estimated hip midpoint and pointing to the left and right knee landmarks. Under the supine–flexed posture, the estimated hip midpoint is displaced lower compared to its anatomically expected location. This inferior shift reduces the angle between them; consequently, abduction is systematically underestimated. This dominant bias is further compounded by variability in knee landmark localisation: small inconsistencies in knee detection alter the midpoint-to-knee vectors and add additional angular error on top of the midpoint's displacement. Finally, the camera's framing was constrained to include the head to ensure stable subject detection, but this reduced the lower-body context. Taken together, these effects produce persistently biased measurements rather than occasional failures, which explains why MAE is large even when landmark detection appears qualitatively stable.

A second key observation is that MAE increases with the true abduction target. This behaviour is consistent with range compression induced by the same geometric mechanism: as TRUE increases, much of the motion is expressed at the knees while the misplaced hip midpoint remains comparatively stable in image space. Because the inferiorly displaced midpoint injects a common vertical component into both midpoint-to-knee vectors, their angular separation grows more slowly than the true separation, so estimated abduction increases with θ_{true} but with a reduced slope. The mixed-effects results support this interpretation, with TRUE being the primary predictor, and significant TRUE× `CAM` and TRUE×FLEX interactions indicate that camera viewpoint and hip flexion have influence the rate at which underestimation increases. These findings motivate modelling approaches that explicitly learn and correct systematic, angle-dependent bias.

5.2 Interpretation of secondary predictors and configuration effects

Height and weight remained significant after controlling for configuration, which is plausible because the estimated angle depends directly on the relative geometry of body landmarks, such as body size, limb proportions. Hip flexion was also a significant factor, with lower MAE observed at 70° compared with 90°. Although all configurations exhibited systematic underestimation, smaller MAE at 70° indicates that the estimated angle was numerically closer to the target. A plausible explanation is that flexion below 90° alters body proportions and projected geometry in a way that inflates the abduction angle in the 2D pose representation, as described in the literature (e.g., [4]). In this case, the flexion-related distortion can oppose the underestimation bias and incidentally reduce the MAE, without implying improved biomechanical validity. This interpretation is consistent with prior observations that non-neutral flexion postures distort midline-based angle estimates and should therefore be treated cautiously in clinical contexts. Finally, camera angle significantly affected both MAE and variability: an oblique 45° view introduces perspective distortion and asymmetric landmark visibility, which biases hip-midpoint and knee localisation and increases between-subject heterogeneity relative to the orthogonal 0° configuration. This is in line with Wang et al., who reported that camera viewing angles (and camera distance) have a statistically significant impact on markerless measurement accuracy and recommended careful camera placement for improved performance [80].

5.3 Comparing the performances of classification and regression performances

The difficulty of Yellow-zone classification likely reflects both data limitations and a conceptual ambiguity: because zone boundaries vary across studies and depend on patient-specific factors such as age and anatomy [43], treating Yellow as a strict, fixed class may be suboptimal in the context of HSC.

Best performing model (SVM with CAM-FLEX features) classifies the Green zone with perfect accuracy, indicating that clearly safe abduction configurations are well separated in feature space. This is expected, as smaller abduction angles are associated with lower estimation error and less geometric ambiguity.

The Yellow zone was the most difficult to classify accurately. Yellow represents a boundary region between safe and potentially dangerous abduction, where measurement uncertainty, inter-individual anatomical variation, and literature-defined thresholds overlap. Many Yellow samples were misclassified as Green (12.1%), reflecting partial overlap in estimated angle distributions. Clinically, this type of error may be acceptable if higher thresholds (e.g., up to 54–55°) are still considered safe [43]. More critical are cases where Yellow is misclassified as Red, which represent unnecessary overestimation of risk. Although such errors were relatively infrequent, they highlight the sensitivity of zone-based classification to threshold definition.

While the model did not misclassify any Red-zone cases as Green, it did misclassify some Red targets as Yellow (9.1%). This is clinically important because Yellow may not be interpreted as a danger category; instead, it can be considered an acceptable or even preferred positioning range, as reported by Kheiri et al. [43]. In that situation, a Red→Yellow error could lead to a hazardous configuration being treated as satisfactory, with potentially serious consequences. Therefore, even with good overall accuracy, the model should not be used as a standalone decision-maker; rather, it is more appropriate as a supportive tool that augments clinical judgement and prompts closer review of borderline cases.

An important limitation of the regression task is that the TRUE was recorded at a small number of discrete target values. As a result, the regression problem partially resembles prediction toward a set of anchor points rather than learning a fully continuous mapping. Collecting data across a denser and more continuous range of abduction angles would likely improve both model training and the validity of regression-based generalisation claims.

Overall, the regression results mirror the classification findings: `CAM` and `FLEX` are key contextual variables for correcting pose-derived abduction estimates, and the `CAM--FLEX` feature set yielded the strongest and most stable LOPO performance. In this setting, the SVM regressor performed best (MAE $= 8.21^\circ$, RMSE $= 11.03^\circ$, $R^2 = 0.713$), indicating the most reliable patient-level generalisation among the tested methods.

Despite this improvement over the geometric estimates the remaining error (MAE of $\sim 8^\circ$ and RMSE $> 11^\circ$) could not be applicable in the clinical use. The moderate R^2 also suggests that a meaningful portion of variability remains unexplained, likely due to residual subject-specific effects.

5.4 Clinical interpretation of `CAM` and `FLEX` effects and implications for machine learning

The `CAM-FLEX` feature set consistently outperforms the other variants because camera angle and hip flexion capture configuration-specific effects that are not contained in the estimated angle alone, allowing the model to correct systematic, setup-dependent bias. Clinically, however, the availability of these variables differs. The oblique camera placements used experimentally (e.g., 45°) are not typical in routine practice and would not usually be recorded, limiting generalisation if `CAM` is required as an explicit input.`FLEX` is more clinically relevant: intermediate flexion (e.g., $\sim 70^\circ$) can occur unintentionally and varies across procedures, so the significant `FLEX` effects likely reflect realistic sources of systematic error. If `CAM` and `FLEX` are not available at deployment, a practical approach is to calibrate the system to a reference configuration (e.g., `CAM` $= 0^\circ$, `FLEX` $= 90^\circ$) and treat deviations as added uncertainty.

To address the substantial systematic error in the raw geometric estimates, both regression-based bias correction and zone-based classification were explored. Regression models substantially reduced MAE relative to the baseline, but did not fully remove the dominant range-compression behaviour, leaving residual error that is likely too large for precise continuous-angle quantification in a clinical setting. In contrast, reframing the task as zone classification produced more clinically aligned performance, since classification is inherently tolerant to moderate continuous-angle error when the goal is to decide whether a posture lies within a broad risk interval (e.g., safe vs. risky) rather than to recover an exact angle in degrees. From a clinical standpoint, avoiding critical misclassification between low- and high-risk postures is often more important than achieving goniometer-level precision; the classification results (particularly SVM with the CAM-FLEX feature set) demonstrate strong separation between categories, supporting the system's current role as a real-time screening and decision-support tool rather than a replacement for quantitative goniometry.

5.5 Effect of temporal smoothing: improved stability but limited accuracy gain

A global grid search selected $\beta = 0$ and $f_{\text{min}} = 0.026$. With $\beta = 0$, speed-based adaptation is disabled and the filter behaves as a conservative low-pass filter governed by f_{min}, prioritising stability over responsiveness. This is appropriate for the current protocol of static pose holds, where suppressing high-frequency landmark noise is more important than minimising latency. Consistent with this objective, filtering reduced jitter in 100% of trials (median reduction 83.46%), substantially improving readability of the displayed angle and making smoothing a practical requirement for any user-facing tool.

Although RMSE was included in the joint objective during parameter tuning, filtering did not change MAE significantly. This is expected because temporal smoothing attenuates high-frequency variability but does not correct systematic error sources such as projection effects, biased hip-midpoint estimation, or persistent landmark mislocalisation. Moreover, smoothing did not improve—and sometimes slightly degraded—downstream machine-learning performance (most notably for RF and XGB, while SVM remained stable). A plausible explanation is that small framewise fluctuations in the raw signal may carry discriminative cues (e.g., subtle adjustments or re-detections) that are informative once aggregated into participant-level features; filtering reduces variance but can also compress dynamic range, decreasing separability near zone boundaries. Under LOPO evaluation this effect may be amplified because filtering interacts with participant-specific noise structure, introducing distribution shifts between training and held-out subjects. Overall, the results suggest that OEF is best used as a front-end stabilisation step for human interpretation, applied cautiously when the filtered signal is used as direct input to predictive models.

6 Limitations and Future Work

This feasibility study has several limitations that should be considered when interpreting the results and judging clinical readiness.

6.1 Limitations

1. **Population and setting (domain shift).** Although the intended application is pediatric hip positioning, validation was performed on healthy adults under controlled, laboratory-like conditions. Differences in body size, limb proportions, soft-tissue distribution, and typical occlusion patterns in real use may introduce substantial domain shift. The cohort was small ($n = 11$) and relatively homogeneous; therefore, the reported performance should not be interpreted as evidence of performance in infants or in intraoperative/post-reduction environments.

2. **Restricted angle sampling and trial dynamics.** The experiment used a discrete and relatively coarse set of target abduction angles (70°, 100°, 130°) with short static holds. While clinically motivated, this provides limited angular granularity and does not characterise behaviour across the full clinically relevant range, particularly near decision boundaries. The absence of longer recordings and transitions between angles further limits conclusions about time-dependent instability (e.g., jitter during movement) and the behaviour of temporal filtering under realistic use.

3. **Limited viewpoint exploration.** Only two camera geometries were tested (frontal 0° and oblique 45°). This restricts inference about viewpoint robustness and about the optimal acquisition geometry. In practice, camera height, distance, and left/right placement may vary and may be constrained by space, staff workflow, and equipment positioning, which could meaningfully affect performance.

4. **Dependence on monocular landmark visibility.** The pipeline relies on monocular markerless pose estimation and therefore depends on visibility of relevant segments and consistent landmark localisation. Self-occlusion, partial out-of-frame capture, and external occlusions (clothing, blankets, staff hands, casting material, or equipment) can degrade landmark placement. Environmental factors such as variable lighting, motion blur, and low contrast between clothing and background can further increase landmark jitter and downstream angle error; these factors are likely to be more common in real clinical scenes than in the controlled acquisition used here.

5. **Structural error and reference-geometry bias.** The target posture (supine with hips and knees flexed and legs abducted) is atypical for general-purpose pose-estimation models. In this configuration, hip landmarks can be difficult to place consistently, the trunk–pelvis midline can shift, and knee landmarks may be biased because the knee occupies a large portion of the image and can obscure the thigh axis. These effects can accumulate into systematic error rather than random noise. Consistent with this, the current pipeline exhibited an angle-dependent

bias (range compression), where higher true abduction angles tended to be underestimated. Temporal smoothing improved visual stability but did not remove this bias, suggesting that a substantial portion of error is structural (projection effects and landmark bias) and unlikely to be resolved by filtering or post-hoc regression alone without revisiting the geometry and/or the observation model.

6. **Constraints on clinical interpretation and label coverage.** Because continuous-angle accuracy remains insufficient under several conditions, conclusions should be restricted to the specific participants, postures, and viewpoints examined. Zone-based classification outperformed continuous regression in this pilot, which may partly reflect the discreteness of the ground-truth angles (70/100/130) rather than true continuous-angle capability. In addition, since White zone was absent, evaluation limitates claims about full-spectrum screening performance.

6.2 Future work

Future work should prioritize (i) broader validation and (ii) methodological changes that directly target systematic bias. Validation should include a larger and more diverse cohort (including clinically representative pediatric data where feasible), broader variation in anthropometry and positioning, and more realistic occlusion patterns. Data collection should cover finer angle increments (e.g., 5°–10° steps), denser sampling across the full clinically relevant range, longer trials, and transitions between angles to assess temporal stability under movement. Viewpoint robustness should be tested systematically across camera height, distance, and left/right placement reflecting realistic clinical constraints, and the next step should include prospective evaluation in a real casting/post-reduction workflow.

Methodologically, reducing range compression and other structural errors will likely require revisiting the reference geometry. Promising directions include: (1) defining more stable trunk/pelvis reference frames using multiple landmarks (rather than a single indirectly computed midline), (2) enforcing bilateral consistency by estimating both sides and verifying midline stability, and (3) applying camera-aware corrections through explicit calibration. Where feasible, multi-view capture or 3D pose estimation could further mitigate projection effects caused by depth changes (e.g., knees moving closer to the camera). Alternatively, depth proxies (e.g., z-coordinates from 3D models or learned depth cues), together with simple biomechanical constraints and landmark refinement, may improve robustness. Future work could also train ML models on filtered signals and compare their performance against the current pipeline. Classification may be improved by collecting additional samples near zone boundaries; alternatively, the Yellow zone could be modelled as a supervised decision region rather than a hard class, enabling clinicians to combine model output with expert judgment. Finally, given current error levels, presenting results primarily as zone-based decision support (inside vs. outside safe ranges) appears more realistic in the near term, while continuous-angle estimation remains a longer-term objective contingent on reducing systematic bias.

7 Conclusions

1. **Literature-driven method selection.** The literature review confirmed that pose-estimation–based measurement is a viable direction for objective hip abduction assessment in pediatric DDH workflows, and supported the selection of clinically appropriate data-processing principles. Among the reviewed options, MB was identified as the most suitable pose-estimation framework for this application due to its accuracy, real-time performance and simplified deployment for the mobile devices.

2. **Feasible architecture and algorithm design.** A complete system architecture and algorithm were **designed and developed**, integrating MB into a unified pipeline, where **coordinate-based vector calculations** were implemented to compute hip abduction angles from extracted landmarks.

3. **TRL4 prototype implementation.** A **TRL4 prototype** was successfully implemented using the MB. The prototype consistently extracted 4 lower-limb landmarks and generated a continuous hip abduction signal in **real-time**, showing landmarks in real-time, as well as total hip abduction, supporting potential integration into clinical surgical decision-making workflows.

4. **Validation shows stability with a systematic limitation.** Validation against ground-truth measurements demonstrated that the system can produce a stable real-time signal; however, accuracy was constrained by a **systematic, angle-dependent bias** (range compression), with underestimation increasing at higher abduction targets. This trend was supported by mixed-effects modelling on the log-transformed MAE, where the true target angle was the strongest predictor of error. Hip flexion and participant demographics improved fit and were significant, suggesting that structural factors (projection geometry and landmark mislocalisation) are significant in this dataset.

5. **Practical decision-support performance and improvement directions.** OEF, using fmin=0.026, β=0.0 as parameters, reduced frame-to-frame jitter around 83% and improved usability, but did not eliminate systematic error. Importantly, **classification using SVM into clinically meaningful abduction zones** performed strongly (LOPO method achieved mean accuracy 0.909), suggesting that the current solution is most promising as a **decision-support tool** for zone-based assessment rather than a high-precision goniometric replacement.

8 References and sources

[1] H. Aguş, H. Omeroğlu, H. Uçar, A. Biçimoğlu, Y. Türmer. “Evaluation of the Risk Factors of Avascular Necrosis of the Femoral Head in Developmental Dysplasia of the Hip in Infants Younger Than 18 Months of Age.” In: *Journal of Pediatric Orthopaedics B* (2002). https://doi.org/10.1097/01202412-200201000-00007.

[2] American Academy of Orthopaedic Surgeons. *Detection and Nonoperative Management of Pediatric Developmental Dysplasia of the Hip in Infants up to Six Months of Age: Evidence-Based Clinical Practice Guideline*. American Academy of Orthopaedic Surgeons, 2022. URL: https://www.aaos.org/globalassets/quality-and-practice-resources/pddh/pddhcpg.pdf.

[3] American Academy of Pediatrics, Committee on Quality Improvement, Subcommittee on Developmental Dysplasia of the Hip. “Clinical practice guideline: early detection of developmental dysplasia of the hip. Committee on Quality Improvement, Subcommittee on Developmental Dysplasia of the Hip. American Academy of Pediatrics.” In: *Pediatrics* 105.4 Pt 1 (2000), pages 896–905. https://doi.org/10.1542/peds.105.4.896.

[4] *An improved method for measuring hip abduction in spica after surgical reduction for developmental dysplasia of the hip*. 2017. URL: https://www.researchgate.net/publication/318122920_An_improved_method_for_measuring_hip_abduction_in_spica_after_surgical_reduction_for_developmental_dysplasia_of_the_hip.

[5] A. S. Barakat, A. Zein, A. Arafa, et al. “Closed reduction with or without adductor tenotomy for developmental dysplasia of the hip: does it affect the safe zone?” In: *Orthopaedic Practice* (2017). URL: https://journals.lww.com/c-orthopaedicpractice/fulltext/2017/03000/closed_reduction_with_or_without_adductor_tenotomy.15.aspx.

[6] D. Bates, M. Mächler, B. Bolker, S. Walker. “Fitting Linear Mixed-Effects Models Using lme4.” In: *Journal of Statistical Software* 67.1 (2015), pages 1–48. https://doi.org/10.18637/jss.v067.i01. URL: https://www.jstatsoft.org/v67/i01/.

[7] V. Bazarevsky, I. Grishchenko. *On-device, Real-time Body Pose Tracking with MediaPipe BlazePose*. Google Research Blog post (posted August 13, 2020). 2020. URL: https://research.google/blog/on-device-real-time-body-pose-tracking-with-mediapipe-blazepose/.

[8] V. Bazarevsky, I. Grishchenko, K. Raveendran, T. Zhu, F. Zhang, M. Grundmann. “BlazePose: On-device Real-time Body Pose Tracking.” In: *arXiv* (2020). URL: https://arxiv.org/abs/2006.10204.

[9] V. Bialik, G. M. Bialik, S. Blazer, P. Sujov, F. Wiener, M. Berant. “Developmental dysplasia of the hip: a new approach to incidence.” In: *Pediatrics* 103.1 (1999), pages 93–99. https://doi.org/10.1542/peds.103.1.93. URL: https://pubmed.ncbi.nlm.nih.gov/9917445/.

[10] D. Böhning. "Multinomial Logistic Regression Algorithm." In: *Annals of the Institute of Statistical Mathematics* 44.1 (1992), pages 197–200. https://doi.org/10.1007/BF00048682.

[11] C. F. Bos, J. L. Bloem, W. R. Obermann, P. M. Rozing. "Magnetic resonance imaging in congenital dislocation of the hip." In: *The Journal of Bone and Joint Surgery. British Volume* 70-B.2 (1988), pages 174–178. https://doi.org/10.1302/0301-620X.70B2.3346282. URL: https://pubmed.ncbi.nlm.nih.gov/3346282/.

[12] H. Bragança, A. Colonna, P. Lopes, J. Fernandes, H. Gamboa, A. Fred. "How Validation Methodology Influences Human Activity Recognition." In: *Sensors* 22.6 (2022), page 2360. https://doi.org/10.3390/s22062360. URL: https://www.mdpi.com/1424-8220/22/6/2360.

[13] L. Breiman. "Random Forests." In: *Machine Learning* 45.1 (2001), pages 5–32. https://doi.org/10.1023/A:1010933404324. URL: https://link.springer.com/article/10.1023/A:1010933404324.

[14] Z. Cao, T. Simon, S.-E. Wei, Y. Sheikh. "Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields." In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pages 7291–7299. https://doi.org/10.1109/CVPR.2017.143. URL: https://openaccess.thecvf.com/content_cvpr_2017/html/Cao_Realtime_Multi-Person_2D_CVPR_2017_paper.html.

[15] Z. Cao, T. Bao, Z. Ren, Y. Fan, K. Deng, W. Jia. "Real-Time Stylized Humanoid Behavior Control through Interaction and Synchronization." In: *Sensors* 22.4 (2022), page 1457. https://doi.org/10.3390/s22041457. URL: https://www.mdpi.com/1424-8220/22/4/1457.

[16] G. Casiez, N. Roussel, D. Vogel. *1€ Filter*. Official project page with implementations and parameter-tuning guidance (fcmin/mincutoff and beta). 2012. URL: https://gery.casiez.net/1euro/.

[17] G. Casiez, N. Roussel, D. Vogel. "1€ Filter: A Simple Speed-Based Low-Pass Filter for Noisy Input in Interactive Systems." In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. Austin, TX, USA: ACM, 2012, pages 2527–2530. https://doi.org/10.1145/2207676.2208639. URL: https://gery.casiez.net/publications/CHI2012-casiez.pdf.

[18] T. Chai, R. R. Draxler. "Root mean square error (RMSE) or mean absolute error (MAE)? Arguments against avoiding RMSE in the literature." In: *Geoscientific Model Development* 7.3 (2014), pages 1247–1250. https://doi.org/10.5194/gmd-7-1247-2014. URL: https://gmd.copernicus.org/articles/7/1247/2014/.

[19] T. Chen, C. Guestrin. "XGBoost: A Scalable Tree Boosting System." In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16)*. 2016, pages 785–794. https://doi.org/10.1145/2939672.2939785. URL: https://arxiv.org/abs/1603.02754.

[20] X. Cheng, Y. Jiao. “Reliability and validity of current computer vision based motion capture systems in gait analysis: A systematic review.” In: *Gait & Posture* 120 (2025), pages 150–160. https://doi.org/10.1016/j.gaitpost.2025.04.016. URL: https://www.sciencedirect.com/science/article/pii/S0966636225001791.

[21] J. E. Cheon et al. “MRI Risk Factors for Development of Avascular Necrosis after Closed Reduction of Developmental Dysplasia of the Hip.” In: *PLOS ONE* (2021). https://doi.org/10.1371/journal.pone.0248701. URL: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0248701.

[22] A. Çitlak et al. “The incidence of avascular necrosis of the femoral head changes with the hip abduction angle in the hip spica in treatment of developmental dislocation of the hip.” In: *Journal of Pediatric Orthopaedics B* (2013). URL: https://journals.lww.com/jpo-b/fulltext/2013/11000/the_incidence_of_avascular_necrosis_of_the_femoral.19.aspx.

[23] C. Cortes, V. Vapnik. “Support-Vector Networks.” In: *Machine Learning* 20 (1995), pages 273–297. https://doi.org/10.1007/BF00994018. URL: https://link.springer.com/article/10.1007/BF00994018.

[24] K. Das et al. “Comparison of markerless and marker-based motion capture systems using functional limits of agreement in a linear mixed-effects modelling framework.” In: *Scientific Reports* 13 (2023), page 23049. https://doi.org/10.1038/s41598-023-49360-2. URL: https://www.nature.com/articles/s41598-023-49360-2.

[25] O. Eberhardt, M. Zieger, M. Langendoerfer, T. Wirth, F. F. Fernandez. “Determination of hip reduction in spica cast treatment for DDH: A comparison of radiography and ultrasound.” In: *Journal of Children’s Orthopaedics* 3.4 (2009). PMC full text accessed 2026-01-03., pages 313–318. https://doi.org/10.1007/s11832-009-0194-5. URL: https://pmc.ncbi.nlm.nih.gov/articles/PMC2726876/.

[26] R. Esteve. “Congenital dislocation of the hip: A review and assessment of results of treatment with special reference to frame reduction as compared with manipulative reduction.” In: *The Journal of Bone and Joint Surgery. British Volume* 42-B (1960), pages 253–263. https://doi.org/10.1302/0301-620X.42B2.253. URL: https://pubmed.ncbi.nlm.nih.gov/13849736/.

[27] Expert Panel on Pediatric Imaging, J. C. Nguyen, S. R. Dorfman, C. K. Rigsby, et al. “ACR Appropriateness Criteria® Developmental Dysplasia of the Hip-Child.” In: *Journal of the American College of Radiology* 16.5S (2019), S94–S103. https://doi.org/10.1016/j.jacr.2019.02.014.

[28] T. Fawcett. “An Introduction to ROC Analysis.” In: *Pattern Recognition Letters* 27.8 (2006), pages 861–874. https://doi.org/10.1016/j.patrec.2005.10.010.

[29] P. S. Foundation. *Python 3.13.5*. Software release page (used Python 3.13.5, 64-bit). 2025. URL: https://www.python.org/downloads/release/python-3135/.

[30] L. Frison, S. J. Pocock. "Repeated measures in clinical trials: analysis using mean summary statistics and its implications for design." In: *Statistics in Medicine* 11.13 (1992), pages 1685–1704. https://doi.org/10.1002/sim.4780111304. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/sim.4780111304.

[31] Z. Fu et al. "Risk factors for avascular necrosis after closed reduction for developmental dysplasia of the hip: the abduction angle in spica cast was not significantly related." In: *Orthopaedics & Traumatology: Surgery & Research / or related indexing record* (2021). Indexed record; verify journal metadata if you prefer to cite the publisher version. URL: https://pesquisa.bvsalud.org/portal/resource/pt/wpr-910682.

[32] K. S. Gather, I. Mavrev, S. Gantz, T. Dreher, S. Hagmann, N. A. Beckmann. "Outcome Prognostic Factors in MRI during Spica Cast Therapy Treating Developmental Hip Dysplasia with Midterm Follow-Up." In: *Children* 9.7 (2022), page 1010. https://doi.org/10.3390/children9071010. URL: https://www.mdpi.com/2227-9067/9/7/1010.

[33] Google. *Model Card: BlazePose GHUM 3D*. ML Kit Pose Detection model card (June 2021). 2021. URL: https://developers.google.com/static/ml-kit/images/vision/pose-detection/pose_model_card.pdf.

[34] Google. *MoveNet on TensorFlow Hub*. Official TensorFlow tutorial describing MoveNet coordinate regression models. 2022. URL: https://www.tensorflow.org/hub/tutorials/movenet.

[35] Google. *Pose detection | ML Kit*. 2024. URL: https://ai.google.dev/edge/mediapipe/solutions/vision/pose_landmarker.

[36] Google. *MediaPipe Pose solution documentation*. MediaPipe Pose overview and usage details. 2025. URL: https://github.com/google-ai-edge/mediapipe/blob/master/docs/solutions/pose.md.

[37] A. L. Gornitzky, A. G. Georgiadis, M. A. Seeley, et al. "Does Perfusion MRI After Closed Reduction of Developmental Dysplasia of the Hip Decrease the Risk of Osteonecrosis?" In: *Clinical Orthopaedics and Related Research* (2016). https://doi.org/10.1007/s11999-015-4387-6. URL: https://link.springer.com/article/10.1007/s11999-015-4387-6.

[38] T. Guo, Q. Yin, X. Liu, Y. Sun, Z. Qin, Y. Han, G. Lu. "Fitness exercise evaluation system based on improved DTW algorithm." In: *Scientific Reports* 15 (2025). Published: 06 June 2025. Article number: 19961., page 19961. https://doi.org/10.1038/s41598-025-02535-5. URL: https://www.nature.com/articles/s41598-025-02535-5.

[39] M. Haenen et al. "Automatic analysis of carpal angles using dynamic CT imaging: Reference values in healthy wrists and assessment of scapholunate ligament injuries." In: *Skeletal Radiology* (2025). Dynamic CT example illustrating movement-capable imaging protocols. URL: https://pubmed.ncbi.nlm.nih.gov/40896875/.

[40] T. Hastie, R. Tibshirani, J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. 2nd edition. Springer, 2009. https://doi.org/10.1007/978-0-387-84858-7.

[41] A. A. Hulleck et al. "Present and future of gait assessment in clinical practice: Towards the application of novel trends and technologies." In: *Frontiers in Medical Technology* (2022). https://doi.org/10.3389/fmedt.2022.901331. URL: https://pmc.ncbi.nlm.nih.gov/articles/PMC9800936/.

[42] B. K. Karmazyn et al. "ACR Appropriateness Criteria® on Developmental Dysplasia of the Hip." In: *Journal of the American College of Radiology* (2009). https://doi.org/10.1016/j.jacr.2009.04.010.

[43] S. Kheiri, M. A. Tahririan, S. Shahnaser, M. Piri Ardakani. "Avascular necrosis predictive factors after closed reduction in patients with developmental dysplasia of the hip." In: *Journal of Research in Medical Sciences* 28, 81 (2023). https://doi.org/10.4103/jrms.jrms_288_23. URL: https://journals.lww.com/jrms/fulltext/2023/11300/avascular_necrosis_predictive_factors_after_closed.81.aspx.

[44] J.-S. Kim, Y.-W. Kim, Y.-K. Woo, K.-N. Park. "Validity of an Artificial Intelligence-Assisted Motion-Analysis System Using a Smartphone for Evaluating Weight-Bearing Activities in Individuals with Patellofemoral Pain Syndrome." In: *Journal of Musculoskeletal Science and Technology* 5.1 (2021), pages 34–40. https://doi.org/10.29273/jmst.2021.5.1.34.

[45] A. Kuznetsova, P. B. Brockhoff, R. H. B. Christensen. "lmerTest Package: Tests in Linear Mixed Effects Models." In: *Journal of Statistical Software* 82.13 (2017), pages 1–26. https://doi.org/10.18637/jss.v082.i13. URL: https://www.jstatsoft.org/v82/i13/.

[46] M. N. Lystbæk, R. R. Krüger, K. Hornbæk, P. O. B. Kristensson. "Hands-on, Hands-off: Gaze-Assisted Bimanual 3D Interaction." In: *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology (UIST '24)*. Pittsburgh, PA, USA: ACM, 2024. https://doi.org/10.1145/3654777.3676331. URL: https://www.researchgate.net/publication/384882367_Hands-on_Hands-off_Gaze-Assisted_Bimanual_3D_Interaction.

[47] Y.-H. Liu, H. K. W. Kim, et al. "Effect of abduction on avascular necrosis of the femoral epiphysis in patients with late-detected developmental dysplasia of the hip treated by closed reduction: a MRI study of 59 hips." In: *The Bone & Joint Journal* (2019). PMC full text. Discusses accepted abduction limits and the Ramsey "safe zone". URL: https://pmc.ncbi.nlm.nih.gov/articles/PMC6808074/.

[48] D. C. Luvizon, D. Picard, H. Tabia. "Human pose regression by combining indirect part detection and contextual information." In: *Computer Vision and Image Understanding* 188 (2019), page 102812. https://doi.org/10.48550/arXiv.1710.02322. URL: https://arxiv.org/abs/1710.02322.

[49] M. Mehdizadeh, M. Dehnavi, A. Tahmasebi, S. A. Mahlisha Kazemi Shishvan, N. Babakhan Kondori, R. Shahnazari. “Transgluteal ultrasonography in spica cast in postreduction assessment of developmental dysplasia of the hip.” In: *Journal of Ultrasound* 23.4 (2020). Published online 2019; journal issue 2020. PMC full text accessed 2026-01-03., pages 509–514. https://doi.org/10.1007/s40477-019-00408-y. URL: https://pmc.ncbi.nlm.nih.gov/articles/PMC7588562/.

[50] X. Meng, J. Yang, Z. Wang. “Magnetic resonance imaging follow-up can screen for soft tissue changes and evaluate the short-term prognosis of patients with developmental dysplasia of the hip after closed reduction.” In: *BMC Pediatrics* 21.1 (2021). PMCID: PMC7938578; open access article on developmental dysplasia of the hip after closed reduction., page 115. https://doi.org/10.1186/s12887-021-02587-2. URL: https://link.springer.com/article/10.1186/s12887-021-02587-2 (viewed 2026-01-04).

[51] M. Mercurio et al. “Artificial Intelligence for the Diagnosis and Management of Musculoskeletal Disorders.” In: *Diagnostics* 15.22 (2025), page 2918. https://doi.org/10.3390/diagnostics15222918.

[52] Michael J. Bellino, MD. *Developmental hip dysplasia (illustration)*. Image used in Figure; retrieved from the Hip Dysplasia page. 2023. URL: https://static.wixstatic.com/media/bc1888_96c39bceefc34bc59d239c7561c6dd3b~mv2.jpeg/v1/crop/x_160%2Cy_0%2Cw_740%2Ch_513/fill/w_320%2Ch_222%2Cal_c%2Cq_80%2Cusm_0.66_1.00_0.01%2Cenc_avif%2Cquality_auto/developmental_hip_dysplasia.jpeg.

[53] Microsoft. *How to trim videos, images, or audio assets*. Clipchamp help documentation for trimming/cutting media to a desired duration. 2025. URL: https://support.microsoft.com/en-us/topic/how-to-trim-videos-images-or-audio-assets-ebe15340-668e-4c31-a8f3-285a659d7fb3.

[54] T. B. Moeslund, E. Granum. “A survey of computer vision-based human motion capture.” In: *Computer Vision and Image Understanding* 104.2–3 (2006), pages 90–126. https://doi.org/10.1016/j.cviu.2006.08.002.

[55] M. Moro et al. “Markerless vs. Marker-Based Gait Analysis: A Proof of Concept Study.” In: *Sensors* 22.5 (2022), page 2011. https://doi.org/10.3390/s22052011. URL: https://www.mdpi.com/1424-8220/22/5/2011.

[56] T. Nandhagopal, F. L. De Cicco. *Developmental Dysplasia of the Hip*. StatPearls [Internet]. Treasure Island (FL): StatPearls Publishing. 2024. URL: https://www.ncbi.nlm.nih.gov/books/NBK563157/.

[57] A. Newell, K. Yang, J. Deng. “Stacked Hourglass Networks for Human Pose Estimation.” In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2016, pages 483–499. https://doi.org/10.1007/978-3-319-46484-8_29. URL: https://arxiv.org/abs/1603.06937.

[58] T. E. Nichols, A. P. Holmes. "Nonparametric permutation tests for functional neuroimaging: a primer with examples." In: *Human Brain Mapping* 15.1 (2002), pages 1–25. https://doi.org/10.1002/hbm.1058.

[59] S. Nussbaumer, M. Leunig, J. F. Glatthorn, S. Stauffacher, H. Gerber, A. Mündermann. "Validity and test-retest reliability of manual goniometers for measuring passive hip range of motion in femoroacetabular impingement patients." In: *BMC Musculoskeletal Disorders* 11 (2010), page 194. https://doi.org/10.1186/1471-2474-11-194.

[60] OpenAI. *ChatGPT. Language models 5.0, 5.1, 5.2*. 2025. URL: https://chatgpt.com/.

[61] OpenCV. *cv::VideoCapture Class Reference*. Official OpenCV documentation (4.x). 2025. URL: https://docs.opencv.org/4.x/d8/dfe/classcv_1_1VideoCapture.html.

[62] H. Patel. "Preventive health care, 2001 update: screening and management of developmental dysplasia of the hip in newborns." In: *CMAJ* (2001). URL: https://pmc.ncbi.nlm.nih.gov/articles/PMC81153/.

[63] R. Poppe. "Vision-based human motion analysis: An overview." In: *Computer Vision and Image Understanding* 108.1–2 (2007), pages 4–18. https://doi.org/10.1016/j.cviu.2006.10.016. URL: https://www.sciencedirect.com/science/article/pii/S1077314206001541.

[64] R. Pospischill, J. Weninger, R. Ganger, J. Altenhuber, F. Grill. "Does open reduction of the developmental dislocated hip increase the risk of osteonecrosis?" In: *Clinical Orthopaedics and Related Research* 470.1 (2012), pages 250–260. https://doi.org/10.1007/s11999-011-1929-4. URL: https://pubmed.ncbi.nlm.nih.gov/21643924/.

[65] P. L. Ramsey, S. Lasser, G. D. MacEwen. "Congenital dislocation of the hip. Use of the Pavlik harness in the child during the first six months of life." In: *J Bone Joint Surg Am* 58.7 (1976), pages 1000–1004. https://doi.org/10.2106/00004623-197658070-00017.

[66] M. J. Rivlin, D. J. Sucato, A. Diaz, et al. "Spica magnetic resonance imaging for determination of abduction angle: initial results and reproducibility assessment." In: *Journal of Pediatric Orthopaedics* (2016). PMC full text. Accessed 2026-01-03. URL: https://pmc.ncbi.nlm.nih.gov/articles/PMC5033784/.

[67] X. Robin, N. Turck, A. Hainard, N. Tiberti, F. Lisacek, J.-C. Sanchez, M. Müller. "pROC: An Open-Source Package for R and S+ to Analyze and Compare ROC Curves." In: *BMC Bioinformatics* 12 (2011), page 77. https://doi.org/10.1186/1471-2105-12-77. URL: https://pmc.ncbi.nlm.nih.gov/articles/PMC3068975/.

[68] W. N. Sankar, A. L. Gornitzky, N. M. P. Clarke, et al. "Closed Reduction for Developmental Dysplasia of the Hip: Early-Term Results From a Prospective, Multicenter Cohort." In: *The Journal of Bone and Joint Surgery. American Volume* (2016). URL: https://pmc.ncbi.nlm.nih.gov/articles/PMC6416015/.

[69] M. D. Schur, C. Lee, A. Arkader, et al. “Risk factors for avascular necrosis after closed reduction and spica casting for developmental dysplasia of the hip.” In: *Journal of Children’s Orthopaedics* (2016). URL: https://pubmed.ncbi.nlm.nih.gov/27177477/.

[70] S. Senn, L. Stevens, N. Chaturvedi. “Repeated measures in clinical trials: simple strategies for analysis using summary measures.” In: *Statistics in Medicine* 19.6 (2000), pages 861–877. https://doi.org/10.1002/(SICI)1097-0258(20000330)19:6<861::AID-SIM407>3.0.CO;2-F. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/%28SICI%291097-0258%2820000330%2919%3A6%3C861%3A%3AAID-SIM407%3E3.0.CO%3B2-F.

[71] B. A. Shaw, L. S. Segal, Section on Orthopaedics. “Evaluation and Referral for Developmental Dysplasia of the Hip in Infants.” In: *Pediatrics* 138.6 (2016). Epub 2016 Nov 21, e20163107. https://doi.org/10.1542/peds.2016-3107.

[72] M. Sokolova, G. Lapalme. “A Systematic Analysis of Performance Measures for Classification Tasks.” In: *Information Processing & Management* 45.4 (2009), pages 427–437. https://doi.org/10.1016/j.ipm.2009.03.002.

[73] K. Sun, B. Xiao, D. Liu, J. Wang. “Deep High-Resolution Representation Learning for Human Pose Estimation.” In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. Open access version provided by the Computer Vision Foundation (CVF). 2019, pages 5693–5703. URL: https://openaccess.thecvf.com/content_CVPR_2019/html/Sun_Deep_High-Resolution_Representation_Learning_for_Human_Pose_Estimation_CVPR_2019_paper.html.

[74] X. Sun, B. Xiao, F. Wei, S. Liang, Y. Wei. “Integral Human Pose Regression.” In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018, pages 529–545. https://doi.org/10.1007/978-3-030-01231-1_33. URL: https://arxiv.org/abs/1711.08229.

[75] R. C. Team. *NEWS for R version 4.5.2 (2025-10-31)*. Release notes (used R 4.5.2, 64-bit). 2025. URL: https://cran.r-project.org/doc/manuals/r-release/NEWS.pdf.

[76] A. Tharatipyakul, T. Srikaewsiew, S. Pongnumkul. “Deep learning-based human body pose estimation in providing feedback for physical movement: A review.” In: *Heliyon* 10.17 (2024). PMCID: PMC11401083; PMID: 39281455. eCollection 2024-09-15., e36589. https://doi.org/10.1016/j.heliyon.2024.e36589. URL: https://pmc.ncbi.nlm.nih.gov/articles/PMC11401083/ (viewed 2026-01-04).

[77] A. Vaquero-Picado, G. González-Morán, E. Gil Garay, L. Moraleda. “Developmental dysplasia of the hip: update of management.” In: *EFORT Open Reviews* 4.9 (2019), pages 548–556. https://doi.org/10.1302/2058-5241.4.180019. URL: https://pubmed.ncbi.nlm.nih.gov/31598333/.

[78] R. Votel, N. Li. *Next-Generation Pose Detection with MoveNet and TensorFlow.js*. TensorFlow Blog post (posted May 17, 2021). 2021. URL: https://blog.tensorflow.org/2021/05/next-generation-pose-detection-with-movenet-and-tensorflowjs.html.

[79] L. Wade, L. Needham, P. McGuigan, J. Bilzon. “Applications and limitations of current markerless motion capture methods for clinical gait biomechanics.” In: *PeerJ* 10 (2022). Open-access full text via PubMed Central., e12995. https://doi.org/10.7717/peerj.12995. URL: https://pmc.ncbi.nlm.nih.gov/articles/PMC8884063/.

[80] H. Wang, B. Su, L. Lu, S. Jung, L. Qing, Z. Xie, X. Xu. “Markerless gait analysis through a single camera and computer vision.” In: *Journal of Biomechanics* 165 (2024), page 112027. https://doi.org/10.1016/j.jbiomech.2024.112027. URL: https://www.sciencedirect.com/science/article/pii/S0021929024001040.

[81] X. M. Wang, D. T. Smith, Q. Zhu. “A webcam-based machine learning approach for three-dimensional range of motion evaluation.” In: *PLOS ONE* 18.10 (2023), e0293178. https://doi.org/10.1371/journal.pone.0293178. URL: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0293178.

[82] M. Ware, E. Frank, G. Holmes, M. Hall, I. H. Witten. “Interactive machine learning: letting users build classifiers.” In: *International Journal of Human-Computer Studies* 55.3 (2001), pages 281–292. https://doi.org/10.1006/ijhc.2001.0499.

[83] S.-E. Wei, V. Ramakrishna, T. Kanade, Y. Sheikh. “Convolutional Pose Machines.” In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pages 4724–4732. https://doi.org/10.1109/CVPR.2016.511. URL: https://openaccess.thecvf.com/content_cvpr_2016/html/Wei_Convolutional_Pose_Machines_CVPR_2016_paper.html.

[84] K. L. Welton, M. J. Kraeutler, T. Garabekyan, O. Mei-Dan. “Radiographic Parameters of Adult Hip Dysplasia.” In: *Orthopaedic Journal of Sports Medicine* 11.2 (2023), page 23259671231152868. https://doi.org/10.1177/23259671231152868.

[85] N. Williams. “Improving early detection of developmental dysplasia of the hip through general practitioner assessment and surveillance.” In: *Australian Journal of General Practice* 47.9 (2018), pages 615–619. https://doi.org/10.31128/AJGP-03-18-4524. URL: https://www1.racgp.org.au/ajgp/2018/september/improving-early-detection-of-developmental-dysplas.

[86] C. J. Willmott, K. Matsuura. “Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance.” In: *Climate Research* 30 (2005), pages 79–82. https://doi.org/10.3354/cr030079. URL: https://www.int-res.com/abstracts/cr/v30/n1/p79-82/.

[87] A. M. Winkler, G. R. Ridgway, M. A. Webster, S. M. Smith, T. E. Nichols. “Permutation inference for the general linear model.” In: *NeuroImage* 92 (2014), pages 381–397. https://doi.org/10.1016/j.neuroimage.2014.01.060.

[88] A. Zeng, L. Yang, X. Ju, J. Li, J. Wang, Q. Xu. "SmoothNet: A Plug-and-Play Network for Refining Human Poses in Videos." In: *Computer Vision – ECCV 2022 (Proceedings, Part V)*. Volume 13665. Lecture Notes in Computer Science. Springer, 2022, pages 625–642. https://doi.org/10.1007/978-3-031-20065-6_36. URL: https://dblp.org/rec/conf/eccv/ZengYJLWX22.

[89] G. Zhang, M. Li, X. Qu, Y. Cao, X. Liu, C. Luo, Y. Zhang. "Efficacy of closed reduction for developmental dysplasia of the hip: midterm outcomes and risk factors associated with treatment failure and avascular necrosis." In: *Journal of Orthopaedic Surgery and Research* 15.1 (2020), page 579. https://doi.org/10.1186/s13018-020-02098-3. URL: https://pubmed.ncbi.nlm.nih.gov/33267908/.

9 Appendices

A Heatmaps for grid search

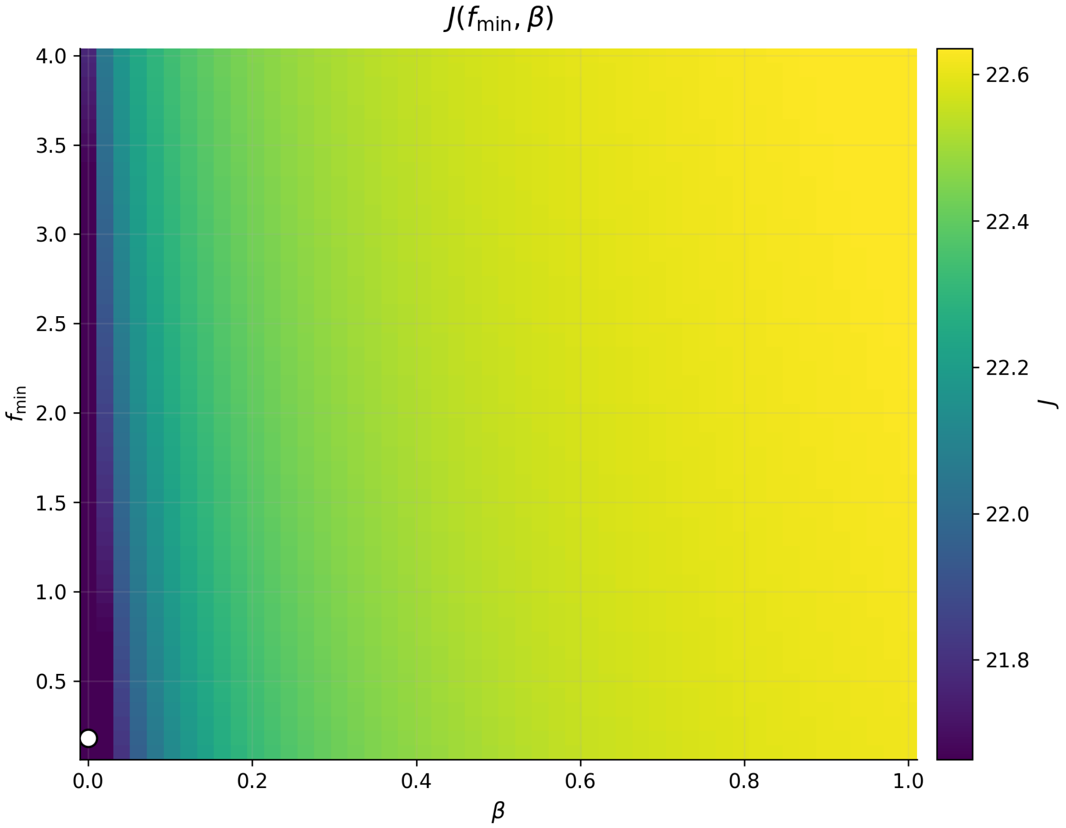


Figure 17 *Heatmap of grid search*

B ANOVA / mixed-effects model outputs

Table 18 *Fixed-effect estimates and Type III tests (Satterthwaite df) for mixed models with outcome* `log1p_mae` *(Eq. 18). All models include a participant random intercept* `(1|patient_id)` *and were fitted by maximum likelihood.*

Model	Term	Est.	SE	df	t	p	Sig.	$\mathbb{F}$	p_{ANOVA}
m0	Intercept	2.3112710	0.0939953	80.6363	24.589	$< 2 \times 10^{-16}$	***	–	–
	`CAM_45`	0.1793349	0.0847239	80.0000	2.117	0.037393	*	4.4804	0.0373933
	`FLEX_90`	-0.3028604	0.0847239	80.0000	-3.575	0.000598	***	12.7783	0.0005981
	`TRUE`	0.0125273	0.0008229	80.0000	15.223	$< 2 \times 10^{-16}$	***	285.0662	$< 2.2 \times 10^{-16}$
	`CAM_45:TRUE`	-0.0016997	0.0008229	80.0000	-2.065	0.042114	*	4.2663	0.0421139
	`FLEX_90:TRUE`	0.0019900	0.0008229	80.0000	2.418	0.017869	*	5.8478	0.0178687
m1	Intercept	2.3202717	0.0933430	82.3000	24.857	$< 2 \times 10^{-16}$	***	–	–
	`CAM_45`	0.1793349	0.0847239	80.0000	2.117	0.037393	*	4.4804	0.0373933
	`FLEX_90`	-0.3028604	0.0847239	80.0000	-3.575	0.000598	***	12.7783	0.0005981
	`TRUE`	0.0125273	0.0008229	80.0000	15.223	$< 2 \times 10^{-16}$	***	285.0662	$< 2.2 \times 10^{-16}$
	`gender`	-0.0450035	0.0424930	10.0000	-1.059	0.314474		1.1217	0.3144743
	`CAM_45:TRUE`	-0.0016997	0.0008229	80.0000	-2.065	0.042114	*	4.2663	0.0421139
	`FLEX_90:TRUE`	0.0019900	0.0008229	80.0000	2.418	0.017869	*	5.8478	0.0178687
m2	Intercept	2.4615635	0.3552586	11.1868	6.929	0.0000228	***	–	–
	`CAM_45`	0.1793349	0.0847239	80.0000	2.117	0.037393	*	4.4804	0.0373933
	`FLEX_90`	-0.3028604	0.0847239	80.0000	-3.575	0.000598	***	12.7783	0.0005981
	`TRUE`	0.0125273	0.0008229	80.0000	15.223	$< 2 \times 10^{-16}$	***	285.0662	$< 2.2 \times 10^{-16}$

Continues on next page

Model	Term	Est.	SE	df	t	p	Sig.	$\mathbb{F}$	p_{ANOVA}
	gender	-0.0496037	0.0435899	10.0000	-1.138	0.281665		1.2950	0.2816653
	age	-0.0052574	0.0127561	10.0000	-0.412	0.688929		0.1699	0.6889289
	CAM_45:TRUE	-0.0016997	0.0008229	80.0000	-2.065	0.042114	*	4.2663	0.0421139
	FLEX_90:TRUE	0.0019900	0.0008229	80.0000	2.418	0.017869	*	5.8478	0.0178687
m3	Intercept	-1.1055841	0.9923523	10.1417	-1.114	0.290941		–	–
	CAM_45	0.1793349	0.0847239	80.0000	2.117	0.037393	*	4.4804	0.0373933
	FLEX_90	-0.3028604	0.0847239	80.0000	-3.575	0.000598	***	12.7783	0.0005981
	TRUE	0.0125273	0.0008229	80.0000	15.223	$< 2 \times 10^{-16}$	***	285.0662	$< 2.2 \times 10^{-16}$
	gender	0.0868808	0.0460760	10.0000	1.886	0.088703	.	3.5555	0.0887028
	age	-0.0003327	0.0081108	10.0000	-0.041	0.968086		0.0017	0.9680861
	height	0.0245142	0.0056841	10.0000	4.313	0.001531	**	18.5996	0.0015305
	weight	-0.0131101	0.0037429	10.0000	-3.503	0.005702	**	12.2683	0.0057015
	CAM_45:TRUE	-0.0016997	0.0008229	80.0000	-2.065	0.042114	*	4.2663	0.0421139
	FLEX_90:TRUE	0.0019900	0.0008229	80.0000	2.418	0.017869	*	5.8478	0.0178687
m4	Intercept	0.8160683	0.7208372	10.2710	1.132	0.283326		–	–
	CAM_45	0.1793349	0.0847239	80.0000	2.117	0.037393	*	4.4804	0.0373933
	FLEX_90	-0.3028604	0.0847239	80.0000	-3.575	0.000598	***	12.7783	0.0005981
	TRUE	0.0125273	0.0008229	80.0000	15.223	$< 2 \times 10^{-16}$	***	285.0662	$< 2.2 \times 10^{-16}$
	height	0.0084522	0.0040424	10.0000	2.091	0.063031	.	4.3720	0.0630314
	CAM_45:TRUE	-0.0016997	0.0008229	80.0000	-2.065	0.042114	*	4.2663	0.0421139
	FLEX_90:TRUE	0.0019900	0.0008229	80.0000	2.418	0.017869	*	5.8478	0.0178687
m5	Intercept	2.4720	0.09307	79.94	26.56	$< 2 \times 10^{-16}$	***	–	–

Continues on next page

Model	Term	Est.	SE	df	t	p	Sig.	$\mathbb{F}$	p_{ANOVA}
	TRUE	0.01130	0.0008206	80.00	13.77	$< 2 \times 10^{-16}$	***	189.55	$< 2.2 \times 10^{-16}$
m6	Intercept	0.3239090	0.5941253	10.4026	0.545	0.597120		–	–
	CAM_45	0.1793349	0.0847239	80.0000	2.117	0.037393	*	4.4804	0.0373933
	FLEX_90	-0.3028604	0.0847239	80.0000	-3.575	0.000598	***	12.7783	0.0005981
	TRUE	0.0125273	0.0008229	80.0000	15.223	$< 2 \times 10^{-16}$	***	285.0662	$< 2.2 \times 10^{-16}$
	height	0.0156235	0.0042041	10.0000	3.716	0.004000	**	13.8108	0.0039996
	weight	-0.0109666	0.0042758	10.0000	-2.565	0.028139	*	6.5784	0.0281386
	CAM_45:TRUE	-0.0016997	0.0008229	80.0000	-2.065	0.042114	*	4.2663	0.0421139
	FLEX_90:TRUE	0.0019900	0.0008229	80.0000	2.418	0.017869	*	5.8478	0.0178687
m7	Intercept	0.4092622	0.6481760	11.2041	0.631	0.540443		–	–
	CAM_45	0.1789239	0.0844533	79.2259	2.119	0.037257	*	4.4885	0.0372566
	FLEX_90	-0.3008807	0.0846151	79.4955	-3.556	0.000638	***	12.6442	0.0006381
	TRUE	0.0124912	0.0008260	80.1920	15.122	$< 2 \times 10^{-16}$	***	279.8464	$< 2.2 \times 10^{-16}$
	height	0.0156516	0.0042892	9.2677	3.649	0.005070	**	13.3154	0.0050703
	v	-0.1035858	0.2797597	44.6427	-0.370	0.712935		0.1371	0.7129345
	weight	-0.0108532	0.0043725	9.3183	-2.482	0.034027	*	6.1612	0.0340269
	CAM_45:TRUE	-0.0017044	0.0008203	79.2303	-2.078	0.040974	*	4.3169	0.0409742
	FLEX_90:TRUE	0.0019828	0.0008204	79.2537	2.417	0.017961	*	5.8404	0.0179610
m8	Intercept	-1.3347343	1.0445970	10.9689	-1.278	0.227717		–	–
	CAM_45	0.1783148	0.0841977	79.8123	2.118	0.037303	*	4.4851	0.0373034
	FLEX_90	-0.2979461	0.0843924	80.0314	-3.530	0.000691	***	12.4643	0.0006912
	TRUE	0.0124375	0.0008247	80.5969	15.081	$< 2 \times 10^{-16}$	***	277.9458	$< 2.2 \times 10^{-16}$
	height	0.0264566	0.0062229	12.0366	4.252	0.001117	**	18.0754	0.0011168

Continues on next page

Model	Term	Est.	SE	df	t	p	Sig.	$\mathbb{F}$	p_{ANOVA}
	v	-0.2571267	0.3064700	67.0035	-0.839	0.404456		0.7039	0.4044557
	gender	0.1080494	0.0531872	13.8176	2.031	0.061899	.	4.1270	0.0618990
	age	0.0036085	0.0094867	14.2851	0.380	0.709264		0.1447	0.7092645
	weight	-0.0128249	0.0038187	9.9167	-3.358	0.007345	**	11.2792	0.0073453
	CAM_45:TRUE	-0.0017113	0.0008178	79.8158	-2.092	0.039577	*	4.3783	0.0395770
	FLEX_90:TRUE	0.0019720	0.0008180	79.8349	2.411	0.018214	*	5.8121	0.0182141
m9	Intercept	0.9862255	0.7730183	11.3638	1.276	0.227484		–	–
	CAM_45	0.1784637	0.0842433	79.6870	2.118	0.037254	*	4.4878	0.0372542
	FLEX_90	-0.2986633	0.0844362	79.9115	-3.537	0.000677	***	12.5114	0.0006768
	TRUE	0.0124506	0.0008251	80.4911	15.090	$< 2 \times 10^{-16}$	***	278.3210	$< 2.2 \times 10^{-16}$
	height	0.0086690	0.0041397	9.7581	2.094	0.063373	.	4.3853	0.0633735
	v	-0.2196039	0.3051031	65.4330	-0.720	0.474230		0.5181	0.4742300
	CAM_45:TRUE	-0.0017096	0.0008183	79.6906	-2.089	0.039876	*	4.3650	0.0398762
	FLEX_90:TRUE	0.0019747	0.0008184	79.7102	2.413	0.018130	*	5.8212	0.0181297

C Python code

```python
# -*- coding: utf-8 -*-

# ################-------------------------------------------------------------------------- code block 0
## minimal single-file pipeline: video -> raw json -> grid search -> plots -> filtered json
# this file does four things:
# 1) extracts hip/knee/shoulder landmarks from videos and computes total abduction angle, saving raw json
# 2) runs a 2d grid search to find best one euro filter parameters (min_cutoff, beta)
# 3) saves heatmaps and group-average plots for raw vs filtered signals
# 4) writes filtered json files by applying the best parameters to raw json signals

import os
import re
import json
import math
from pathlib import Path

import cv2
import numpy as np
import matplotlib.pyplot as plt
import mediapipe as mp


# ################-------------------------------------------------------------------------- code block 0a
## configuration
# defines input/output folders and a few processing constants

video_root = Path("video") # input videos: video/<patient>/*.mp4 or *.mov
raw_root = Path("data") # output raw json: data/<patient>/pose_data_raw_<video>.json
flt_root = Path("data_one_euro") # output filtered json: data_one_euro/<patient>/pose_data_one_euro_<video>.json

results_dir = raw_root / "global_grid_results"
plots_dir = results_dir / "plots"
heatmaps_dir = results_dir / "heatmaps"
```

```

sample_ms = 50 # sample one frame every sample_ms milliseconds
duration_limit_s = 2.0 # process only first duration_limit_s seconds of each video
show_preview = False # set true only for manual visual checking during debugging

# grid search setup (min_cutoff, beta)
n_min_cutoff = 25
n_beta = 20
min_cutoff_min = 0.1
min_cutoff_max = 4.0
beta_min = 0.0
beta_max = 1.0

d_cutoff = 1.0 # derivative cutoff kept fixed for simplicity

# create folders
for p in [raw_root, flt_root, results_dir, plots_dir, heatmaps_dir]:
    p.mkdir(parents=True, exist_ok=True)


# ################-------------------------------------------------------------------------- code block 1
## one euro filter implementation
# adaptive low-pass filter for smoothing angle signals with less lag during fast motion

class OneEuro:
    def __init__(self, min_cutoff=1.0, beta=0.0, d_cutoff=1.0):
        self.min_cutoff = float(min_cutoff)
        self.beta = float(beta)
        self.d_cutoff = float(d_cutoff)
        self.x_prev = None
        self.dx_prev = 0.0
        self.t_prev = None

    @staticmethod
    def _alpha(dt, cutoff):
        tau = 1.0 / (2.0 * math.pi * cutoff)
        return 1.0 / (1.0 + tau / dt)

    def filter(self, t, x):
        if self.t_prev is None:
            self.t_prev, self.x_prev, self.dx_prev = t, x, 0.0
            return x

        dt = max(1e-6, t - self.t_prev)
        dx = (x - self.x_prev) / dt

        a_d = self._alpha(dt, self.d_cutoff)
        dx_hat = a_d * dx + (1.0 - a_d) * self.dx_prev

        cutoff = max(1e-6, self.min_cutoff + self.beta * abs(dx_hat))
        a_x = self._alpha(dt, cutoff)
        x_hat = a_x * x + (1.0 - a_x) * self.x_prev

        self.t_prev, self.x_prev, self.dx_prev = t, x_hat, dx_hat
        return x_hat


# ################-------------------------------------------------------------------------- code block 2
## geometry helpers + total abduction angle
# computes the angle between vectors (hip-midpoint -> left knee) and (hip-midpoint -> right knee)

def _unit(v):
    n = math.hypot(v["x"], v["y"])
    return {"x": v["x"] / n, "y": v["y"] / n} if n > 1e-6 else {"x": 0.0, "y": 0.0}

def _sub(a, b):
    return {"x": a["x"] - b["x"], "y": a["y"] - b["y"]}

def _dot(a, b):
    return a["x"] * b["x"] + a["y"] * b["y"]

def _angle_between(a, b):
    c = max(-1.0, min(1.0, _dot(_unit(a), _unit(b))))
    return math.degrees(math.acos(c))

def _hip_midpoint(L):
    return {
        "x": (L["left_hip"]["x"] + L["right_hip"]["x"]) / 2.0,
        "y": (L["left_hip"]["y"] + L["right_hip"]["y"]) / 2.0,
    }
```

```

def total_abduction(L):
    H_mid = _hip_midpoint(L)
    vL = _sub(L["left_knee"], H_mid)
    vR = _sub(L["right_knee"], H_mid)
    return _angle_between(vL, vR), H_mid


# ################--------------------------------------------------------------------------- code block 3
## video -> raw json extraction (mediapipe pose)
# reads each video, samples frames, runs mediapipe pose, computes abduction, and saves raw json per video

mp_pose = mp.solutions.pose
pose = mp_pose.Pose(
    model_complexity=1,
    enable_segmentation=False,
    min_detection_confidence=0.3,
    min_tracking_confidence=0.3,
)

LM = {
    "LEFT_HIP": 23,
    "RIGHT_HIP": 24,
    "LEFT_KNEE": 25,
    "RIGHT_KNEE": 26,
}

def extract_raw_from_videos(video_root: Path, raw_root: Path):
    patient_dirs = sorted([p for p in video_root.iterdir() if p.is_dir() and p.name.lower().startswith("n")])

    for patient_dir in patient_dirs:
        out_dir = raw_root / patient_dir.name
        out_dir.mkdir(parents=True, exist_ok=True)

        videos = sorted([f for f in patient_dir.iterdir() if f.suffix.lower() in (".mp4", ".mov")])
        for video_path in videos:
            cap = cv2.VideoCapture(str(video_path))
            if not cap.isOpened():
```

```
            continue

        fps = cap.get(cv2.CAP_PROP_FPS) or 30.0
        frame_interval = int(round(fps * (sample_ms / 1000.0)))
        duration_frames = int(duration_limit_s * fps)

        frames = []
        frame_count = 0

        while cap.isOpened():
            ret, frame = cap.read()
            if not ret or frame_count > duration_frames:
                break

            frame = cv2.rotate(frame, cv2.ROTATE_180)

            if frame_count % frame_interval == 0:
                rgb = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
                res = pose.process(rgb)
                timestamp = round(frame_count / fps, 3)

                if res.pose_landmarks:
                    lm = res.pose_landmarks.landmark
                    coords = {}
                    for name, idx in LM.items():
                        p = lm[idx]
                        coords[name.lower()] = {
                            "x": round(p.x, 5),
                            "y": round(p.y, 5),
                            "z": round(p.z, 5),
                            "v": round(p.visibility, 5),
                        }

                    abd_raw, H_mid = total_abduction(coords)

                    frames.append({
                        "timeframe_s": timestamp,
                        "abduction_total": round(abd_raw, 3),
                        "landmarks": coords,
                    })

                    if show_preview:
                        h, w, _ = frame.shape
                        points = {n: (int(c["x"] * w), int(c["y"] * h)) for n, c in coords.items()}
                        Hx, Hy = int(H_mid["x"] * w), int(H_mid["y"] * h)

                        for p_pt in points.values():
                            cv2.circle(frame, p_pt, 6, (0, 0, 255), -1)
                        cv2.circle(frame, (Hx, Hy), 6, (0, 0, 255), -1)

                        cv2.line(frame, (Hx, Hy), points["left_knee"], (0, 255, 0), 2)
                        cv2.line(frame, (Hx, Hy), points["right_knee"], (0, 255, 0), 2)

                        cv2.putText(frame, f"raw: {abd_raw:.1f} deg", (50, 50),
                                    cv2.FONT_HERSHEY_SIMPLEX, 0.7, (255, 255, 255), 2)

                        cv2.imshow("pose tracking (total abduction)", cv2.resize(frame, (640, 360)))
                        if cv2.waitKey(1) & 0xFF == ord("q"):
                            break

            frame_count += 1

        cap.release()
        if show_preview:
            cv2.destroyAllWindows()

        out_path = out_dir / f"pose_data_raw_{video_path.stem}.json"
        with open(out_path, "w", encoding="utf-8") as f:
            json.dump({
                "video_file": video_path.name,
                "patient_folder": patient_dir.name,
                "settings": {
                    "filtered": False,
                    "sample_ms": sample_ms,
                    "duration_limit_s": duration_limit_s,
                },
                "frames": frames,
            }, f, indent=2, ensure_ascii=False)


```

```
# ################---------------------------------------------------------------------------------------------- code block 4
## searching for best parameters for one euro - grid search
# loads raw json trials, extracts labels from filenames, evaluates each (min_cutoff, beta) by 0.5*rmse + 0.5*noise

def parse_video_name(name: str):
    base = Path(name).stem.lower()
    m_cam = re.search(r"(\d+)cam", base)
    cam = int(m_cam.group(1)) if m_cam else None

    m_flex = re.search(r"(\d+)f_", base)
    flex = int(m_flex.group(1)) if m_flex else None

    m_abd = re.search(r"f_(\d+)", base)
    abduction = float(m_abd.group(1)) if m_abd else np.nan

    condition = f"cam{cam}_f{flex}" if (cam is not None and flex is not None) else "unknown"
    return abduction, condition

def load_trials_from_raw(raw_root: Path):
    trials = []
    patient_folders = sorted([p for p in raw_root.iterdir() if p.is_dir()])

    for patient_dir in patient_folders:
        json_files = sorted([f for f in patient_dir.iterdir()
                             if f.name.startswith("pose_data_raw_") and f.suffix == ".json"])
        for jf in json_files:
            with open(jf, "r", encoding="utf-8") as f:
                j = json.load(f)

            frames = j.get("frames", [])
            if not frames:
                continue

            times = np.array([fr.get("timeframe_s", np.nan) for fr in frames], dtype=float)
            raw_angles = np.array([fr.get("abduction_total", np.nan) for fr in frames], dtype=float)

```

```python
            video_name = j.get("video_file", jf.name)
            abduction, condition = parse_video_name(video_name)
            if np.isnan(abduction):
                continue

            trials.append({
                "patient": patient_dir.name,
                "json_file": jf.name,
                "video_file": video_name,
                "condition": condition,
                "abduction": abduction,
                "times": times,
                "raw": raw_angles,
                "true_vals": np.full_like(raw_angles, abduction, dtype=float),
            })

    return trials

def filter_signal(raw_angles, times, params):
    f = OneEuro(min_cutoff=params["min_cutoff"], beta=params["beta"], d_cutoff=d_cutoff)
    return np.array([f.filter(t, x) for t, x in zip(times, raw_angles)], dtype=float)

def rmse(pred, true):
    pred = np.asarray(pred, dtype=float)
    true = np.asarray(true, dtype=float)
    m = ~np.isnan(pred) & ~np.isnan(true)
    return float(np.sqrt(np.mean((pred[m] - true[m]) ** 2))) if np.any(m) else np.nan

def noise(sig):
    s = np.array(sig, dtype=float)
    if len(s) < 2:
        return np.nan
    return float(np.nanmean(np.abs(np.diff(s))))

def run_grid_search(trials):
    min_cutoff_grid = np.linspace(min_cutoff_min, min_cutoff_max, n_min_cutoff)
    beta_grid = np.linspace(beta_min, beta_max, n_beta)

    obj_grid = np.zeros((n_min_cutoff, n_beta))
    rmse_grid = np.zeros((n_min_cutoff, n_beta))
    noise_grid = np.zeros((n_min_cutoff, n_beta))

    best_value = None
    best_params = None

    for i, min_c in enumerate(min_cutoff_grid):
        for j, beta in enumerate(beta_grid):
            params = {"min_cutoff": float(min_c), "beta": float(beta)}

            rmses = []
            noises = []
            for tr in trials:
                filtered = filter_signal(tr["raw"], tr["times"], params)
                rmses.append(rmse(filtered, tr["true_vals"]))
                noises.append(noise(filtered))

            mean_rmse = np.nanmean(rmses)
            mean_noise = np.nanmean(noises)
            value = 0.5 * mean_rmse + 0.5 * mean_noise

            rmse_grid[i, j] = mean_rmse
            noise_grid[i, j] = mean_noise
            obj_grid[i, j] = value

            if best_value is None or value < best_value:
                best_value = float(value)
                best_params = params.copy()

    return best_params, best_value, min_cutoff_grid, beta_grid, obj_grid, rmse_grid, noise_grid


# ################------------------------------------------------------------------------------------------ code block 5
## plotting outputs (heatmaps + group mean curves)
# creates summary visuals for the grid search and for average raw vs filtered signals per (abduction, condition) group

def plot_heatmap(grid, min_cutoff_grid, beta_grid, title, out_path: Path):
    plt.figure(figsize=(6, 5))
    extent = [beta_grid[0], beta_grid[-1], min_cutoff_grid[0], min_cutoff_grid[-1]]
    plt.imshow(grid, origin="lower", extent=extent, aspect="auto")
    plt.colorbar(label=title)
    plt.xlabel("beta")
    plt.ylabel("min_cutoff")
    plt.title(title)
    plt.tight_layout()
    plt.savefig(out_path, dpi=150)
    plt.close()

def make_group_plots(trials, best_params, out_dir: Path, n_points=200):
    out_dir.mkdir(parents=True, exist_ok=True)

    for tr in trials:
        tr["filtered"] = filter_signal(tr["raw"], tr["times"], best_params)

    groups = {}
    for tr in trials:
        key = (tr["abduction"], tr["condition"])
        groups.setdefault(key, []).append(tr)

    time_grid = np.linspace(0.0, 1.0, n_points)

    for (abduction, condition), group_trials in sorted(groups.items(), key=lambda x: (x[0][0], x[0][1])):
        raw_list, flt_list = [], []

        for tr in group_trials:
            raw = tr["raw"]
            flt = tr["filtered"]
            n = len(raw)
            if n < 2:
                continue

            t_norm = np.linspace(0.0, 1.0, n)
            raw_list.append(np.interp(time_grid, t_norm, raw))
            flt_list.append(np.interp(time_grid, t_norm, flt))

        if not raw_list:
            continue

        raw_mean = np.nanmean(np.vstack(raw_list), axis=0)
        flt_mean = np.nanmean(np.vstack(flt_list), axis=0)

        plt.figure(figsize=(8, 4))
        plt.plot(time_grid, raw_mean, label="mean raw signal")
        plt.plot(time_grid, flt_mean, label="mean filtered signal", linestyle="--")
        plt.xlabel("normalized time")
        plt.ylabel("abduction angle (deg)")
        plt.title(f"abduction {abduction:.0f} deg, condition: {condition}")
        plt.legend()
        plt.tight_layout()

        safe_condition = re.sub(r"[^a-zA-Z0-9_]+", "", condition)
        fname = f"abduction_{int(abduction)}_{safe_condition}.png"
        plt.savefig(out_dir / fname, dpi=150)
        plt.close()


# ################--------------------------------------------------------------------------- code block 6
## raw json -> filtered json writing
# applies best one euro parameters to each raw json signal and saves a filtered json file

def write_filtered_jsons(raw_root: Path, flt_root: Path, params):
    patient_folders = sorted([p for p in raw_root.iterdir() if p.is_dir()])

    for patient_dir in patient_folders:
        out_dir = flt_root / patient_dir.name
        out_dir.mkdir(parents=True, exist_ok=True)

        json_files = sorted([f for f in patient_dir.iterdir()
                          if f.name.startswith("pose_data_raw_") and f.suffix == ".json"])
        for jf in json_files:
            with open(jf, "r", encoding="utf-8") as f:
                j = json.load(f)

            frames = j.get("frames", [])
            if not frames:
                continue

            times = np.array([fr.get("timeframe_s", np.nan) for fr in frames], dtype=float)
            raw_angles = np.array([fr.get("abduction_total", np.nan) for fr in frames], dtype=float)

            flt_angles = filter_signal(raw_angles, times, params)

            flt_frames = [{"timeframe_s": float(t), "abduction_total": round(float(a), 3)}
                          for t, a in zip(times, flt_angles)]

            base_name = jf.name.replace("pose_data_raw_", "").replace(".json", "")
            out_path = out_dir / f"pose_data_one_euro_{base_name}.json"

            with open(out_path, "w", encoding="utf-8") as f:
                json.dump({
                    "video_file": j.get("video_file", ""),
                    "patient_folder": j.get("patient_folder", patient_dir.name),
                    "settings": {
                        "filtered": True,
                        "filter_type": "one_euro",
                        "min_cutoff": params["min_cutoff"],
                        "beta": params["beta"],
                        "d_cutoff": d_cutoff,
                        "sample_ms": sample_ms,
                        "duration_limit_s": duration_limit_s,
                    },
                    "frames": flt_frames,
                }, f, indent=2, ensure_ascii=False)


# ################---------------------------------------------------------------------- code block 7
## eexecution

def main():
    extract_raw_from_videos(video_root, raw_root)

    trials = load_trials_from_raw(raw_root)
    if not trials:
        return

    best_params, best_value, min_c_grid, beta_grid, obj_grid, rmse_grid, noise_grid = run_grid_search(trials)

    with open(results_dir / "one_euro_global_params_grid.json", "w", encoding="utf-8") as f:
        json.dump({
            "min_cutoff": best_params["min_cutoff"],
            "beta": best_params["beta"],
            "d_cutoff": d_cutoff,
            "best_combined_value": float(best_value),
            "grid_min_cutoff": min_c_grid.tolist(),
            "grid_beta": beta_grid.tolist(),
        }, f, indent=2, ensure_ascii=False)

    plot_heatmap(obj_grid, min_c_grid, beta_grid,
                 "combined (0.5*rmse + 0.5*noise)", heatmaps_dir / "heatmap_combined.png")
    plot_heatmap(rmse_grid, min_c_grid, beta_grid,
                 "mean rmse", heatmaps_dir / "heatmap_rmse.png")
    plot_heatmap(noise_grid, min_c_grid, beta_grid,
                 "mean noise", heatmaps_dir / "heatmap_noise.png")

    make_group_plots(trials, best_params, plots_dir)
    write_filtered_jsons(raw_root, flt_root, best_params)


if __name__ == "__main__":
    try:
        main()
    finally:
        pose.close()
        cv2.destroyAllWindows()
```

D R code

```r
################## ---------------------------------- 1. PACKAGES.

# install.packages(c("dplyr","tidyr","stringr","purrr","tibble","readxl",
# "jsonlite","ggplot2","lme4","lmerTest"))
# install.packages(c("randomForest","xgboost","nnet","pROC","e1071")) # optional ML

library(dplyr)
library(tidyr)
library(stringr)
library(purrr)
library(tibble)
library(readxl)
library(jsonlite)
library(ggplot2)
library(lme4)
library(lmerTest)


################## ---------------------------------- 2. FUNCTIONS.

# small helper: use left if it exists, otherwise right
`%||%` <- function(a, b) if (!is.null(a)) a else b

# quick numeric conversion (Excel/factors sometimes mess this up)
as_num <- function(x) as.numeric(as.character(x))

# folders look like n1 / n012 etc.
extract_patient_id_from_folder <- function(folder_name) {
  as.numeric(str_extract(folder_name, "\\d+"))
}

# find patient folders in a directory (only first level)
find_patient_dirs <- function(root_path) {
  dirs <- list.dirs(root_path, full.names = TRUE, recursive = FALSE)
  dirs[str_detect(basename(dirs), "^n\\d+$")]
}

# simple measure of how ""jumpy the signal is
noise_mean_abs_diff <- function(x) {
  x <- as.numeric(x)
  mean(abs(diff(x)), na.rm = TRUE)
```

```r
}

# map angle to zone based on thresholds in cfg$zone
classify_zone <- function(angle, zone) {
  if (is.na(angle)) return(NA_character_)
  if (angle >= zone$white && angle < zone$green) return("white")
  if (angle >= zone$green && angle < zone$yellow) return("green")
  if (angle >= zone$yellow && angle < zone$red) return("yellow")
  if (angle >= zone$red) return("red")
  NA_character_
}

# read questionnaire and rename main columns to standard names
read_quiz <- function(quiz_file) {
  quiz <- read_excel(quiz_file)

  quiz %>%
    select(1:(ncol(.) - 3)) %>% # keep same logic as before
    rename_with(~ case_when(
      str_detect(.x, "žamius|žAmius|amzius|ius") ~ "age",
      str_detect(.x, "voris|svoris") ~ "weight",
      str_detect(.x, "gisū|gis|ugis") ~ "height",
      str_detect(.x, "ytis|lytis") ~ "gender",
      str_detect(.x, "koja") ~ "dominant_leg",
      str_detect(.x, "paciento") ~ "patient_id",
      str_detect(.x, "daryti") ~ "trauma",
      TRUE ~ .x
    )) %>%
    mutate(patient_id = as.numeric(str_extract(as.character(patient_id), "\\d+")))
}

# parse trial info from video file name
parse_video_file <- function(video_file) {
  stem <- tools::file_path_sans_ext(basename(video_file))
  m <- str_match(stem, ".*n(\\d+)_([0-3])_([0-9]+)cam_([0-9]+)f_([0-9]+)")

  tibble(
    video_file = video_file,
    patient_id = as.numeric(m[1, 2]),
    trial_code = m[1, 3],
    rep_index = as.numeric(m[1, 3]),
    camera_deg = as.numeric(m[1, 4]),
    hip_flexion_deg = as.numeric(m[1, 5]),
    abduction_target_deg = as.numeric(m[1, 6])
  )
}

# process one json: take frame data -> trial summary row
process_one_json <- function(json_path, cfg) {
  obj <- fromJSON(json_path, flatten = TRUE)
  frames <- obj$frames

  # metadata comes either from json fields or from folder
  video_file <- obj$video_file %||% NA_character_
  patient_folder <- obj$patient_folder %||% NA_character_
  pf_id <- extract_patient_id_from_folder(patient_folder)

  meta <- if (!is.na(video_file)) {
    parse_video_file(video_file)
  } else {
    tibble(
      video_file = NA_character_,
      patient_id = pf_id,
      trial_code = NA_character_,
      rep_index = NA_real_,
      camera_deg = NA_real_,
      hip_flexion_deg = NA_real_,
      abduction_target_deg = NA_real_
    )
  }

  # main signal per frame
  abduction_raw <- as.numeric(frames$abduction_total)

  # visibility is optional (depends on json columns)
  v_cols <- grep("\\.v$", names(frames), value = TRUE)
  frame_visibility <- if (length(v_cols) > 0) {
    rowMeans(as.matrix(frames[, v_cols, drop = FALSE]), na.rm = TRUE)
  } else {
    rep(NA_real_, length(abduction_raw))
  }
  mean_visibility <- mean(frame_visibility, na.rm = TRUE)

  # true target angle (from parsed name)
  target <- meta$abduction_target_deg[1]

  # trial-level summaries
  mean_abduction <- mean(abduction, na.rm = TRUE)
  abs_error_vec <- abs(abduction - target)

  bind_cols(
    tibble(
      json_file = basename(json_path),
      patient_folder = patient_folder,
      n_frames = length(abduction),
      mean_visibility = mean_visibility,
      true_abduction_deg = target,
      estimated_abduction_deg = mean_abduction,
      mean_abduction = mean_abduction,
      mae_frame_mean = mean(abs_error_vec, na.rm = TRUE),
      rmse_abduction = sqrt(mean((abduction - target)^2, na.rm = TRUE)),
      noise_mean_abs_diff = noise_mean_abs_diff(abduction),
      true_zone = classify_zone(target, cfg$zone),
      estimated_zone = classify_zone(mean_abduction, cfg$zone),
      abs_error_vec = list(abs_error_vec)
    ),
    meta
  )
}

# go through folders and all data collected
read_all_json <- function(cfg) {
  patient_dirs <- unlist(lapply(cfg$data_dirs, find_patient_dirs))
  map_dfr(patient_dirs, function(patient_dir) {
    json_files <- list.files(patient_dir, pattern = "\\.json$", full.names = TRUE)
    map_dfr(json_files, ~process_one_json(.x, cfg))
  })
}

# join trial data with quiz
build_master_raw <- function(df_trial, quiz) {
  master_table <- df_trial %>%
    mutate(bias = mean_abduction - true_abduction_deg) %>%
    left_join(quiz, by = "patient_id") %>%
    filter(!(grepl("^\\d{2}$", as.character(trial_code)))) %>%
    select(-trial_code)

  master_condition <- master_table %>%
    filter(!is.na(patient_id),
           !is.na(camera_deg),
           !is.na(hip_flexion_deg),
           !is.na(abduction_target_deg)) %>%
    group_by(patient_id, camera_deg, hip_flexion_deg, abduction_target_deg) %>%
    summarise(
      n_trials = n_distinct(rep_index),
      theta_mean = mean(mean_abduction, na.rm = TRUE),
      mae_mean = mean(mae_frame_mean, na.rm = TRUE),
      rmse_mean = mean(rmse_abduction, na.rm = TRUE),
      noise_mean = mean(noise_mean_abs_diff, na.rm = TRUE),
      vis_mean = mean(mean_visibility, na.rm = TRUE),
      mae_median = median(unlist(abs_error_vec), na.rm = TRUE),
      mae_q1 = quantile(unlist(abs_error_vec), 0.25, na.rm = TRUE, names = FALSE),
      mae_q3 = quantile(unlist(abs_error_vec), 0.75, na.rm = TRUE, names = FALSE),
      .groups = "drop"
    )

  list(master_table = master_table, master_condition = master_condition)
}

# mixed model for MAE (log scale)
run_lmm_model_suite <- function(dat0, outcome,
                                out_dir = NULL,
                                save_files = TRUE) {

  need_all <- c(outcome,
                "camera_deg", "hip_flexion_deg", "angle_true", "patient_id",
                "gender", "age", "height", "weight", "bmi", "vis_mean")

  dat_m <- dat0 %>%
    select(any_of(need_all)) %>%
    drop_na() %>%
    mutate(
      patient_id = factor(patient_id),
      camera_deg = factor(camera_deg),
      hip_flexion_deg = factor(hip_flexion_deg),
      gender = factor(gender),
      age = as_num(age),
      height = as_num(height),
      weight = as_num(weight),
      bmi = as_num(bmi),
      vis_mean = as_num(vis_mean)
    ) %>%
    droplevels()


  f_m0 <- as.formula(paste0(outcome, " ~ camera_deg * hip_flexion_deg * angle_true + (1|patient_id)"))
  f_m1 <- as.formula(paste0(outcome, " ~ camera_deg * hip_flexion_deg * angle_true + gender + (1|patient_id)"))
  f_m2 <- as.formula(paste0(outcome, " ~ camera_deg * hip_flexion_deg * angle_true + gender + age + (1|patient_id)"))
  f_m3 <- as.formula(paste0(outcome, " ~ camera_deg * hip_flexion_deg * angle_true + gender + age + height + weight + (1|patient_id)"))
  f_m4 <- as.formula(paste0(outcome, " ~ camera_deg * hip_flexion_deg * angle_true + height + (1|patient_id)"))
  f_m5 <- as.formula(paste0(outcome, " ~ angle_true + (1|patient_id)"))
  f_m6 <- as.formula(paste0(outcome, " ~ camera_deg * hip_flexion_deg * angle_true + height + weight + (1|patient_id)"))
  f_m7 <- as.formula(paste0(outcome, " ~ camera_deg * hip_flexion_deg * angle_true + height + vis_mean + weight + (1|patient_id)"))
  f_m8 <- as.formula(paste0(outcome, " ~ camera_deg * hip_flexion_deg * angle_true + height + vis_mean + gender + age + weight + (1|patient_id)"))
  f_m9 <- as.formula(paste0(outcome, " ~ camera_deg * hip_flexion_deg * angle_true + height + vis_mean + (1|patient_id)"))

  models <- list(
    m0 = lmerTest::lmer(f_m0, data = dat_m, REML = FALSE),
    m1 = lmerTest::lmer(f_m1, data = dat_m, REML = FALSE),
    m2 = lmerTest::lmer(f_m2, data = dat_m, REML = FALSE),
    m3 = lmerTest::lmer(f_m3, data = dat_m, REML = FALSE),
    m4 = lmerTest::lmer(f_m4, data = dat_m, REML = FALSE),
    m5 = lmerTest::lmer(f_m5, data = dat_m, REML = FALSE),
    m6 = lmerTest::lmer(f_m6, data = dat_m, REML = FALSE),
    m7 = lmerTest::lmer(f_m7, data = dat_m, REML = FALSE),
    m8 = lmerTest::lmer(f_m8, data = dat_m, REML = FALSE),
    m9 = lmerTest::lmer(f_m9, data = dat_m, REML = FALSE)
  )
}

# small theme so plots look similar
theme_thesis <- function(base_size = 13) {
  theme_minimal(base_size = base_size) +
    theme(
      plot.title = element_text(face = "bold", size = base_size + 2),
      axis.title = element_text(face = "bold"),
```

```r
      panel.grid.minor = element_blank()
    )
}

# build confusion matrix + plots for zones
make_confusion_outputs <- function(master_condition, cfg) {
  mc_raw <- master_condition %>%
    mutate(
      angle_true = as_num(abduction_target_deg),
      theta_mean = as_num(theta_mean),
      true_zone = vapply(angle_true, classify_zone, character(1), zone = cfg$zone),
      est_zone = vapply(theta_mean, classify_zone, character(1), zone = cfg$zone)
    ) %>%
    filter(!is.na(true_zone), !is.na(est_zone)) %>%
    mutate(
      true_zone = factor(as.character(true_zone), levels = c("white","green","yellow","red")),
      est_zone = factor(as.character(est_zone), levels = c("white","green","yellow","red"))
    )

  cm <- with(mc_raw, table(true_zone, est_zone))
  cm_df <- as.data.frame(cm)
  names(cm_df) <- c("true_zone", "est_zone", "n")

  cm_prop <- prop.table(cm, margin = 1)
  cm_prop_df <- as.data.frame(cm_prop)
  names(cm_prop_df) <- c("true_zone", "est_zone", "prop")

  zone_labels <- c(
    white = "White (<60°)",
    green = "Green -(6089°)",
    yellow = "Yellow -(90109°)",
    red = "Red  (110°)"
  )

  p_counts <- ggplot(cm_df, aes(x = est_zone, y = true_zone, fill = n)) +
    geom_tile(color = "white", linewidth = 0.6) +
    geom_text(aes(label = n), size = 4) +
    scale_x_discrete(labels = zone_labels, drop = FALSE) +
    scale_y_discrete(labels = zone_labels, drop = FALSE) +
    coord_equal() +
    labs(title = "Confusion matrix (counts)",
        x = "Estimated zone", y = "True zone", fill = "Count") +
    theme_thesis(13)

  p_prop <- ggplot(cm_prop_df, aes(x = est_zone, y = true_zone, fill = prop)) +
    geom_tile(color = "white", linewidth = 0.6) +
    geom_text(aes(label = sprintf("%.1f%%", 100 * prop)), size = 4) +
    scale_x_discrete(labels = zone_labels, drop = FALSE) +
    scale_y_discrete(labels = zone_labels, drop = FALSE) +
    coord_equal() +
    labs(title = "Confusion matrix (row %)",
        x = "Estimated zone", y = "True zone", fill = "Row %") +
    theme_thesis(13)

  zone_acc <- mean(as.character(mc_raw$true_zone) == as.character(mc_raw$est_zone), na.rm = TRUE)

  list(
    data_used = mc_raw,
    counts_table = cm_df,
    prop_table = cm_prop_df,
    zone_accuracy = zone_acc,
    plot_counts = p_counts,
    plot_prop = p_prop
  )
}

# -BlandAltman plot (true vs estimated)
make_bland_altman_outputs <- function(master_condition) {
  mc_raw <- master_condition %>%
    mutate(
      angle_true = as_num(abduction_target_deg),
      theta_mean = as_num(theta_mean)
    ) %>%
    filter(angle_true != 0)

  ba <- mc_raw %>%
    mutate(
      mean_angle = (theta_mean + angle_true) / 2,
      diff_angle = theta_mean - angle_true
    )

  ba_mean <- mean(ba$diff_angle, na.rm = TRUE)
  ba_sd <- sd(ba$diff_angle, na.rm = TRUE)
  loa_hi <- ba_mean + 1.96 * ba_sd
  loa_lo <- ba_mean - 1.96 * ba_sd

  annot_txt <- paste0(
    "Mean bias = ", sprintf("%.2f°", ba_mean), "\n",
    "SD = ", sprintf("%.2f°", ba_sd), "\n",
    "LOA = [", sprintf("%.2f°", loa_lo), ", ", sprintf("%.2f°", loa_hi), "]"
  )

  p_ba <- ggplot(ba, aes(x = mean_angle, y = diff_angle)) +
    geom_hline(yintercept = 0, linewidth = 0.4, linetype = "dotted") +
    geom_hline(yintercept = ba_mean, linewidth = 0.8) +
    geom_hline(yintercept = loa_hi, linewidth = 0.6, linetype = "dashed") +
    geom_hline(yintercept = loa_lo, linewidth = 0.6, linetype = "dashed") +
    geom_point(size = 2, alpha = 0.75) +
    annotate("label",
             x = min(ba$mean_angle, na.rm = TRUE),
             y = max(ba$diff_angle, na.rm = TRUE),
             hjust = 0, vjust = 1,
             label = annot_txt, label.size = 0.2) +
    labs(title = "-BlandAltman plot",
         x = "Mean of (estimated, true) angle (°)",
         y = "Difference (estimated -true) (°)") +
    theme_thesis(13)

  list(
    data_used = ba,
    limits = tibble(mean_diff = ba_mean, sd_diff = ba_sd, loa_lo = loa_lo, loa_hi = loa_hi),
    plot = p_ba
  )
}

# scatter plot (true vs predicted), split by camera/flexion
make_true_vs_est_plot <- function(master_condition) {
  mc_raw <- master_condition %>%
    mutate(
      angle_true = as_num(abduction_target_deg),
      theta_mean = as_num(theta_mean)
    ) %>%
    filter(angle_true != 0)

  ggplot(mc_raw, aes(x = angle_true, y = theta_mean)) +
```

```r
    geom_point(alpha = 0.75, size = 2) +
    geom_abline(intercept = 0, slope = 1, linetype = "dashed", linewidth = 0.6) +
    facet_grid(hip_flexion_deg ~ camera_deg, drop = TRUE) +
    labs(title = "True vs estimated angle",
         x = "True angle (°)", y = "Estimated angle (°)") +
    theme_thesis(12)
}

# to make sure train/test have the same dummy variables
make_mm <- function(X_train, X_test) {
  for (nm in names(X_train)) {
    if (is.character(X_train[[nm]])) X_train[[nm]] <- factor(X_train[[nm]])
    if (is.factor(X_train[[nm]])) {
      X_test[[nm]] <- factor(X_test[[nm]], levels = levels(X_train[[nm]]))
    }
  }

  mm_train <- model.matrix(~ . - 1, data = X_train)
  mm_test <- model.matrix(~ . - 1, data = X_test)

  miss <- setdiff(colnames(mm_train), colnames(mm_test))
  if (length(miss) > 0) {
    mm_test <- cbind(mm_test, matrix(0, nrow(mm_test), length(miss),
                                     dimnames = list(NULL, miss)))
  }

  mm_test <- mm_test[, colnames(mm_train), drop = FALSE]
  list(mm_train = mm_train, mm_test = mm_test)
}

# macro F1: average per class
macro_f1 <- function(y_true, y_pred) {
  y_true <- factor(y_true)
  y_pred <- factor(y_pred, levels = levels(y_true))
  lv <- levels(y_true)

  cm <- table(truth = y_true, pred = y_pred)

  f1s <- sapply(lv, function(cl) {
    tp <- cm[cl, cl]
    fp <- sum(cm[, cl]) - tp
    fn <- sum(cm[cl, ]) - tp
    prec <- tp / (tp + fp)
    rec <- tp / (tp + fn)
    2 * prec * rec / (prec + rec)
  })

  mean(f1s, na.rm = TRUE)
}

# macro AUC
macro_auc <- function(y_true, prob_mat) {
  y_true <- factor(y_true)
  lv <- levels(y_true)
  prob_mat <- as.data.frame(prob_mat)

  aucs <- sapply(lv, function(cl) {
    y_bin <- as.integer(y_true == cl)
    r <- pROC::roc(response = y_bin, predictor = prob_mat[[cl]], quiet = TRUE)
    as.numeric(pROC::auc(r))
  })

  mean(aucs, na.rm = TRUE)
}

# MAE/RMSE/R2 for regression
reg_metrics <- function(true, pred) {
  err <- pred - true
  mae <- mean(abs(err), na.rm = TRUE)
  rmse <- sqrt(mean(err^2, na.rm = TRUE))
  r2 <- 1 - sum((true - pred)^2, na.rm = TRUE) /
    sum((true - mean(true, na.rm = TRUE))^2, na.rm = TRUE)
  list(n = length(true), MAE = mae, RMSE = rmse, R2 = r2)
}

# LOPO classification
run_lopo_classification <- function(df, feature_cols,
                               target_col = "true_zone",
                               patient_col = "patient_id",
                               models = c("Baseline", "RandomForest", "XGBoost", "
                                   Multinom", "SVM")) {

  df <- df %>% filter(!is.na(.data[[patient_col]]), !is.na(.data[[target_col]]))
  feature_cols <- intersect(feature_cols, names(df))
  patients <- sort(unique(df[[patient_col]]))

```

```r
results <- tibble(test_patient = character(), model = character(),
                  accuracy = numeric(), macro_F1 = numeric(), macro_AUC = numeric())

for (pid in patients) {
  train <- df %>% filter(.data[[patient_col]] != pid)
  test <- df %>% filter(.data[[patient_col]] == pid)

  X_train <- train[, feature_cols, drop = FALSE]
  X_test <- test[, feature_cols, drop = FALSE]
  y_train <- factor(train[[target_col]])
  y_test <- factor(test[[target_col]], levels = levels(y_train))
  class_levels <- levels(y_train)

  add_row <- function(model_name, y_pred, prob_mat = NULL) {
    tibble(test_patient = as.character(pid), model = model_name,
           accuracy = mean(y_pred == y_test, na.rm = TRUE),
           macro_F1 = macro_f1(y_test, y_pred),
           macro_AUC = if (is.null(prob_mat)) NA_real_ else macro_auc(y_test, prob_mat))
  }

  # baseline
  if ("Baseline" %in% models) {
    maj <- names(which.max(table(y_train)))
    pred <- factor(rep(maj, length(y_test)), levels = class_levels)
    results <- bind_rows(results, add_row("Baseline_Majority", pred, NULL))
  }

  # random forest
  if ("RandomForest" %in% models) {
    rf <- randomForest::randomForest(x = X_train, y = y_train, ntree = 500)
    prob <- predict(rf, newdata = X_test, type = "prob")
    prob <- prob[, class_levels, drop = FALSE]
    pred <- factor(colnames(prob)[max.col(prob)], levels = class_levels)
    results <- bind_rows(results, add_row("RandomForest", pred, prob))
  }

  # xgboost multiclass
  if ("XGBoost" %in% models) {
    mm <- make_mm(X_train, X_test)
    K <- length(class_levels)
    y_num <- as.integer(y_train) - 1L
    dtrain <- xgboost::xgb.DMatrix(mm$mm_train, label = y_num)
    dtest <- xgboost::xgb.DMatrix(mm$mm_test)

    params <- list(objective = "multi:softprob", num_class = K, eval_metric = "mlogloss",
                   max_depth = 4, eta = 0.1, subsample = 0.8, colsample_bytree = 0.8)
    xgb <- xgboost::xgb.train(params = params, data = dtrain, nrounds = 200, verbose = 0)

    prob_vec <- predict(xgb, dtest)
    prob <- matrix(prob_vec, ncol = K, byrow = TRUE)
    colnames(prob) <- class_levels
    pred <- factor(class_levels[max.col(prob)], levels = class_levels)
    results <- bind_rows(results, add_row("XGBoost", pred, prob))
  }

  # multinomial logistic regression
  if ("Multinom" %in% models) {
    mlr <- nnet::multinom(y_train ~ ., data = data.frame(y_train = y_train, X_train),
                          trace = FALSE, MaxNWts = 100000, maxit = 500)
    prob <- as.matrix(predict(mlr, newdata = X_test, type = "probs"))
    prob <- prob[, class_levels, drop = FALSE]
    pred <- factor(class_levels[max.col(prob)], levels = class_levels)
    results <- bind_rows(results, add_row("Multinom_LogReg", pred, prob))
  }

  # svm with probabilities
  if ("SVM" %in% models) {
    svm <- e1071::svm(y_train ~ ., data = data.frame(y_train = y_train, X_train),
                      type = "C-classification", kernel = "radial",
                      probability = TRUE, scale = TRUE, cost = 1)
    pred_obj <- predict(svm, newdata = X_test, probability = TRUE)
    pred <- factor(pred_obj, levels = class_levels)
    prob <- attr(pred_obj, "probabilities")[, class_levels, drop = FALSE]
    results <- bind_rows(results, add_row("SVM", pred, prob))
  }
}

summary_tbl <- results %>%
  group_by(model) %>%
  summarise(
    mean_acc = mean(accuracy, na.rm = TRUE),
    mean_macro_F1 = mean(macro_F1, na.rm = TRUE),
      mean_macro_AUC = mean(macro_AUC, na.rm = TRUE),
      .groups = "drop"
    )

  list(results = results, summary = summary_tbl)
}

# LOPO regression
run_lopo_regression <- function(df, feature_cols,
                                target_col = "angle_true",
                                patient_col = "patient_id",
                                models = c("LM","RandomForest","XGBoost","SVM")) {

  df <- df %>% filter(!is.na(.data[[patient_col]]), !is.na(.data[[target_col]]))
  feature_cols <- intersect(feature_cols, names(df))
  patients <- sort(unique(df[[patient_col]]))

  res <- tibble(test_patient = character(), model = character(),
                n_test = integer(), MAE = numeric(), RMSE = numeric(), R2 = numeric())

  for (pid in patients) {
    train <- df %>% filter(.data[[patient_col]] != pid)
    test <- df %>% filter(.data[[patient_col]] == pid)

    X_train <- train[, feature_cols, drop = FALSE]
    X_test <- test[, feature_cols, drop = FALSE]
    y_train <- train[[target_col]]
    y_test <- test[[target_col]]

    add_row <- function(model_name, pred) {
      m <- reg_metrics(y_test, pred)
      tibble(test_patient = as.character(pid), model = model_name,
             n_test = length(y_test), MAE = m$MAE, RMSE = m$RMSE, R2 = m$R2)
    }

    # linear model baseline
    if ("LM" %in% models) {
      lm_mod <- lm(y_train ~ ., data = data.frame(y_train = y_train, X_train))
      pred <- as.numeric(predict(lm_mod, newdata = X_test))
      res <- bind_rows(res, add_row("LM", pred))
    }

    mm <- make_mm(X_train, X_test)
    Xtr <- mm$mm_train; Xte <- mm$mm_test

    # random forest regression
    if ("RandomForest" %in% models) {
      rf <- randomForest::randomForest(x = Xtr, y = y_train, ntree = 500)
      pred <- as.numeric(predict(rf, newdata = Xte))
      res <- bind_rows(res, add_row("RandomForest", pred))
    }

    # xgboost regression
    if ("XGBoost" %in% models) {
      dtrain <- xgboost::xgb.DMatrix(Xtr, label = y_train)
      dtest <- xgboost::xgb.DMatrix(Xte, label = y_test)
      params <- list(objective = "reg:squarederror", max_depth = 4, eta = 0.1,
                  subsample = 0.8, colsample_bytree = 0.8)
      xgb <- xgboost::xgb.train(params = params, data = dtrain, nrounds = 200, verbose = 0)
      pred <- as.numeric(predict(xgb, dtest))
      res <- bind_rows(res, add_row("XGBoost", pred))
    }

    # svm regression
    if ("SVM" %in% models) {
      svm <- e1071::svm(y_train ~ ., data = data.frame(y_train = y_train, X_train),
                     type = "eps-regression", kernel = "radial",
                     scale = TRUE, cost = 1, epsilon = 0.1)
      pred <- as.numeric(predict(svm, newdata = X_test))
      res <- bind_rows(res, add_row("SVM", pred))
    }
  }

  summary_tbl <- res %>%
    group_by(model) %>%
    summarise(
      mean_MAE = mean(MAE, na.rm = TRUE),
      mean_RMSE = mean(RMSE, na.rm = TRUE),
      mean_R2 = mean(R2, na.rm = TRUE),
      .groups = "drop"
    )

  list(results = res, summary = summary_tbl)
}
```

```
# feature sets
make_feature_sets <- function() {
  list(
    est_only = c("theta_mean"),
    est_cam_flex = c("theta_mean","camera_deg","hip_flexion_deg"),
    est_vis = c("theta_mean","vis_mean"),
    est_all = c("theta_mean","camera_deg","hip_flexion_deg","vis_mean")
  )
}

# main function: run everything and return it as a list
run_pipeline_raw <- function(cfg) {
  quiz <- read_quiz(cfg$quiz_file)
  df_trial <- read_all_json(cfg)

  masters <- build_master_raw(df_trial, quiz)
  master_table <- masters$master_table
  master_condition <- masters$master_condition

  mae_out <- run_mae_lmm(master_condition, quiz)
  vis_out <- run_visibility_lmm(master_condition, quiz)

  confusion_out <- make_confusion_outputs(master_condition, cfg)
  bland_altman_out <- make_bland_altman_outputs(master_condition)
  true_vs_est_plot <- make_true_vs_est_plot(master_condition)

  # dataset for LOPO
  ml_df <- master_condition %>%
    mutate(
      angle_true = as_num(abduction_target_deg),
      theta_mean = as_num(theta_mean),
      vis_mean = as_num(vis_mean),
      true_zone = vapply(angle_true, classify_zone, character(1), zone = cfg$zone),
      true_zone = factor(true_zone, levels = c("white","green","yellow","red")),
      patient_id = factor(patient_id),
      camera_deg = factor(camera_deg),
      hip_flexion_deg = factor(hip_flexion_deg)
    ) %>%
    filter(angle_true != 0) %>%
    select(patient_id, true_zone, angle_true, theta_mean, vis_mean, camera_deg, hip_
        flexion_deg)

  feature_sets <- make_feature_sets()
```

```r

  classification_summary <- imap(feature_sets, function(feats, nm) {
    run_lopo_classification(ml_df, feats)$summary %>% mutate(feature_set = nm)
  }) %>% bind_rows()

  regression_summary <- imap(feature_sets, function(feats, nm) {
    run_lopo_regression(ml_df, feats)$summary %>% mutate(feature_set = nm)
  }) %>% bind_rows()

  list(
    cfg = cfg,
    quiz = quiz,
    df_trial = df_trial,
    master_table = master_table,
    master_condition = master_condition,
    mixed_models = list(mae = mae_out, visibility = vis_out),
    thesis_plots = list(
      confusion = confusion_out,
      bland_altman = bland_altman_out,
      true_vs_est = true_vs_est_plot
    ),
    ml_df = ml_df,
    classification_summary = classification_summary,
    regression_summary = regression_summary
  )
}


################## ---------------------------------- 3. EXECUTION.

cfg <- list(
  data_dirs = file.path(".", "data"),
  quiz_file = file.path(".", "data.xlsx"),
  visibility_min = NA_real_,
  smooth_k = NA_integer_,
  zone = list(white = 0, green = 60, yellow = 90, red = 110)
)

out <- run_pipeline_raw(cfg)

master_condition <- out$master_condition
mae_anova <- out$mixed_models$mae$anova
vis_anova <- out$mixed_models$visibility$anova
```

```

p_cm_counts <- out$thesis_plots$confusion$plot_counts
p_ba <- out$thesis_plots$bland_altman$plot
p_true_est <- out$thesis_plots$true_vs_est

# print(p_cm_counts)
# print(p_ba)
# print(p_true_est)
```