VRIJE UNIVERSITEIT AMSTERDAM

UNIVERSITEIT VAN AMSTERDAM

Master Thesis

From Pixels to Poses: A Supervised Spatio-Temporal Refinement Pipeline for Monocular 3D Human Pose Estimation in Broadcast Football

by

Hendrik Hermann Werner Joel

(2722498)

First supervisor:	Mauricio Verano Merino
Daily supervisor:	Mauricio Verano Merino
Second reader:	Emitza Guzman Ortega

September 14, 2026

Submitted in partial fulfillment of the requirements for the joint UvA-VU degree of Master of Science in Computer Science

From Pixels to Poses: A Supervised Spatio-Temporal Refinement Pipeline for Monocular 3D Human Pose Estimation in Broadcast Football

Hendrik Hermann Werner Joel
Vrije Universiteit Amsterdam
Amsterdam, The Netherlands
h.h.w.joel@student.vu.nl

ABSTRACT

Extracting reliable 3D human pose from football footage remains an open challenge: players occupy as few as 50 pixels in height, occlusions are frequent, and the motion distribution of competitive sport lies far outside the laboratory benchmarks on which state-of-the-art lifting models are trained.

This thesis investigates whether a modular, supervised spatio-temporal refinement stage can meaningfully correct the systematic errors that monocular 3D pose lifting produces in this domain. We construct a five-stage pipeline: SAM 3-based segmentation, ByteTrack and OSNet re-identification, ViTPose-H keypoint estimation, deterministic pre- and post-lifting refinement, and MotionBERT 3D lifting and train a Spatio-Temporal Pose Refinement Transformer as a post-hoc correction stage on top of it, supervised by game-engine ground truth from a simulated 90-minute football match.

The Transformer achieves 80.4 mm MPJPE at its best checkpoint on held-out sequences of the simulated match, a 56.3% reduction from the 184.0 mm pipeline baseline. Gaussian temporal smoothing produces less than 1% improvement over the raw pipeline, confirming that the gains arise from learned structural pose priors rather than filtering alone. Notably, this result is comparable to the approximately 98 mm achieved by Yeung et al. [37] through domain-specific fine-tuning of MotionAGFormer on the AthletePose3D dataset, but without any sports-specific retraining of the lifting backbone, suggesting that a supervised post-hoc correction stage is a practical and modular alternative to end-to-end domain adaptation.

These findings validate the viability of supervised spatio-temporal refinement in simulation and establish a reproducible foundation for evaluating its performance on real broadcast footage.
[1]

1 INTRODUCTION

Professional football analysis is undergoing a fundamental shift. As clubs increasingly demand objective, high-throughput tactical and biomechanical insights, the manual tracking and annotation workflows that still underpin much of the industry are becoming a bottleneck: slow, expensive, and subjective. Broadcast footage represents the most abundant and historically rich source of football data available, yet extracting reliable 3D skeletal information from it remains an unsolved engineering challenge. Unlike controlled laboratory environments, broadcast streams are characterised by small player scales, rapid motion, frequent inter-player occlusions, and the fundamental irreversibility of monocular projection conditions that expose the limits of systems designed for and evaluated on standard benchmarks.

The advent of foundational vision models such as SAM 3 [3] and the emergence of sports-specific initiatives such as the FIFA Skeletal Light Tracking Challenge [11] have made it timely to ask whether a robust, reproducible 3D pose estimation pipeline can be built for use within the field of football, from modular off-the-shelf components and whether its systematic errors can be corrected by a dedicated learned refinement stage. This thesis addresses that question directly, through two concrete objectives.

The first objective is to construct a modular end-to-end pipeline for multi-player 3D pose estimation on synthetic footage, assembling and integrating components, SAM 3-based segmentation, ByteTrack and OSNet re-identification, ViTPose-H keypoint estimation, and MotionBERT 3D lifting, into a system that is robust to the failure modes that disqualify simpler approaches. The second objective is to investigate the central research question of this thesis:

> *Can a supervised spatio-temporal refinement stage, trained on game-engine ground truth, meaningfully correct the systematic errors produced by monocular 3D pose lifting?*

To answer this, we design and train a Spatio-Temporal Pose Refinement Transformer that operates as a post-hoc correction stage on top of the lifting pipeline. The model incorporates a novel learnable skeleton adjacency bias and is supervised by a composite loss enforcing temporal smoothness, bone-length integrity, and bilateral symmetry. Unlike black-box end-to-end systems, the decoupled architecture allows each component to be evaluated, swapped, and improved independently. This is a practical necessity in a domain where no single model is trained for all sub-tasks.

Our empirical results validate the approach: the refinement stage reduces MPJPE from 184.0 mm to 80.4 mm, a 56.3% improvement over the pipeline, and reduces mean body orientation error from 52.3 to 21.5. This provides a robust, reproducible foundation for future work in automated biomechanical analysis, tactical performance monitoring, and real-time sports broadcasting.

2 BACKGROUND

This chapter introduces the technical concepts and prior work that underpin the system developed in this thesis. The goal is not an exhaustive survey but a grounded understanding of the specific tools, architectures, and challenges relevant to estimating and refining 3D

[1] The full pipeline implementation, training code, and configuration files are available at https://github.com/Schorle21/SportsAnalysis.

human pose from broadcast football footage. Readers already familiar with pose estimation may wish to focus on Sections 2.4 and 2.8, which most directly motivate the contributions of this thesis.

2.1 An Introduction to 3D Human Pose Estimation

Human Pose Estimation (HPE) is the computational task of localising the anatomical keypoints of the human body from visual input and mapping them to a structured representation of joint locations in two or three dimensions [17]. 3D HPE is of particular interest in contexts where spatial reasoning is required: rather than predicting where a joint appears in the image plane, it recovers where it exists in the physical world as a coordinate (x, y, z) relative to a defined reference frame. Downstream tasks such as activity recognition, biomechanical analysis [21], tactical formation analysis, and motion prediction all benefit from a reliable skeletal estimates.

Historically, accurate 3D skeletal data required dedicated motion-capture infrastructure, such as arrays of calibrated cameras tracking retro-reflective markers, or body-worn IMUs, which is impractical outside controlled laboratory settings. Deep learning has made it feasible to infer pose from a single uncalibrated RGB camera, opening new application domains including in-the-wild sports analysis.

The dominant current approach is a modular two-stage pipeline: a 2D keypoint detector localizes joints in the image plane, and a separate 3D lifting network elevates these detections into world space [43, 44]. Within the lifting stage, three families of methods have emerged. GCN-based approaches represent the skeleton as a graph whose edges encode physical joint connectivity [16, 38], offering parameter efficiency and structural transparency at the cost of limited long-range reasoning. Transformer-based approaches apply self-attention globally across all joints and frames, learning arbitrary correlations without structural constraints [23, 40, 44], and currently represent the state of the art on standard benchmarks. Hybrid architectures combine both paradigms to achieve competitive accuracy at lower computational cost [26, 34]. A distinct direction is parametric body model recovery, which predicts a dense surface mesh via the SMPL model [21, 30] rather than a sparse joint set, enabling downstream volumetric analyses at the cost of fitting a high-dimensional parameter space.

A fundamental challenge underlying all monocular methods is depth ambiguity: infinitely many 3D configurations can produce identical 2D keypoint patterns. Methods address this through learned statistical priors [32], probabilistic multi-hypothesis generation [6, 23], or geometric constraints from the capture environment [15]. Without sufficient temporal regularisation, frame-by-frame estimation produces jittery trajectories and anatomically implausible configurations [21, 30] that undermine downstream analysis, a failure mode that directly motivates the refinement contribution of this thesis.

2.2 Technical Challenges of Monocular Pose Estimation

The core difficulty of monocular 3D HPE is that perspective projection is irreversible. A camera maps a three-dimensional coordinate (X, Y, Z) onto a two-dimensional image point (u, v) by:

$$u = f_x \frac{X}{Z} + c_x, \qquad v = f_y \frac{Y}{Z} + c_y \tag{1}$$

where f_x, f_y are the focal lengths and (c_x, c_y) the principal point. The depth Z is not recoverable from (u, v) alone [13]: a limb extended toward the camera and one extended sideways can produce identical image projections [15, 32]. This is the fundamental depth ambiguity that all learned lifting methods must overcome.

In multi-player sports, this problem is compounded by occlusion: defenders, attackers, and referees frequently overlap in the image plane, causing entire body segments to disappear for consecutive frames. A monocular system must rely on temporal context or learned priors to infer missing joints [6, 41]. Wide-angle cameras further introduce perspective distortion at frame edges, where players appear foreshortened in ways that are underrepresented in standard training data [39]. The specific challenge of player scale in broadcast footage and its implications for detection and lifting quality, is examined in detail in Section 4.1, where it directly motivates the architectural decisions of this thesis.

Taken together, depth ambiguity, inter-player occlusion, and limited camera perspective form a set of compounding difficulties that no single mechanism fully resolves. Addressing them requires a two-stage pipeline that separates robust 2D detection from learned 3D lifting, and the integration of temporal context to bridge occluded frames the architectural choices examined in the following section.

2.3 Architectural Evolution

The earliest deep learning approaches to 3D HPE were end-to-end regression models mapping raw pixels directly to 3D joint coordinates in a single forward pass [27, 32]. In practice, such models overfit to training data appearance and generalise poorly to unseen environments, a critical weakness for sports deployment. The framework is also rigid: improving any single component requires retraining the entire model.

The Two-Stage Lifting Paradigm. In response, the field converged on a two-stage approach [40, 44]: a dedicated 2D keypoint detector first localises body joints in image coordinates, and a separate 3D lifting network infers 3D joint positions from the resulting coordinate sequences. The lifting network never processes raw pixels as it operates entirely on structured coordinate sequences that are compact, appearance-invariant, and transferable across domains. Each stage can be pre-trained, swapped, and debugged independently. Successive transformer-based lifters [23, 26, 40, 44] have pushed Human3.6M accuracy from 44.3 mm to below 40 mm purely by operating on 2D coordinate sequences (Table 2). Figure 1 illustrates the generic architecture.

The lifting backbone adopted in this thesis is MotionBERT [46], which pre-trains a Dual-stream Spatio-Temporal Transformer on both motion-capture and in-the-wild video before fine-tuning for 3D pose estimation. Attending over long temporal windows gives the model statistical priors about valid human motion, directly addressing the depth ambiguity of Section 2.2. MotionBERT achieves 37.5 mm MPJPE on Human3.6M; the full justification for its adoption is given in Section 4. An alternative, RTMW3D [18], predicts

3D joints directly from single image crops but sacrifices the temporal context essential for trajectory coherence in sports footage, favouring the lifting approach. Beyond accuracy, modularity enables inference speedups, edge–cloud splitting, and independent component updates [9, 35, 39], directly motivating the architecture adopted in this thesis.

2.4 Limitations of Standard Benchmarks

The progress documented above has been measured almost exclusively against controlled laboratory benchmarks, most prominently Human3.6M [17] and COCO17 [24]. While instrumental in driving methodological advances, their construction conditions introduce a systematic gap between reported accuracy and real-world deployment performance that is particularly severe in the sports domain.

The Laboratory Assumption and the Motion Gap. Human3.6M [17] remains the dominant 3D HPE benchmark more than a decade after its introduction: 3.6 million marker-based poses from 11 actors performing everyday activities: walking, eating, sitting, under four fixed cameras in a controlled indoor laboratory. The annotation quality is exceptional, but the range of possible motions is intentionally constrained and dominated by slow, upright locomotion and seated postures, a distribution that shares very little with the rapid, multi-planar, high-acceleration movements of competitive sport: explosive directional changes, diving, jumping, contact situations, and extreme joint configurations appear in no laboratory-captured dataset. The AthletePose3D benchmark [37] was introduced specifically to address this gap, showing that models evaluated solely on Human3.6M may achieve low MPJPE on walking while failing catastrophically on sprinting or collision scenarios, a concern reinforced by sports deployment studies [35, 39].

Studio Conditions vs. Real-World Deployment. The capture conditions diverge just as sharply. Human3.6M was recorded in a small laboratory with calibrated cameras, professional lighting, and a single uncluttered subject [17]; sports footage is captured by wide-angle cameras covering an entire pitch under variable lighting, with multiple mutually occluding players each occupying a small fraction of the frame. This introduces three failure modes the benchmarks cannot expose: *scale degradation*, as players spanning 50–100 pixels in height make fine-grained joint localisation unreliable [37]; *inter-player occlusion*, which Human3.6M's single-subject scenes never require a model to handle; and *perspective distortion* at wide-angle frame edges, largely absent from benchmark training data.

Biomechanical Plausibility and Benchmark Fitness. A further consequence of training on laboratory benchmarks is the production of biomechanically implausible poses under the conditions that matter most for sports science. When a model encounters joint configurations outside its training distribution a full-extension sprint stride, a goalkeeper dive, or a contact tackle it has no learned prior to constrain its output to physically valid configurations. The result is estimates with hyperextended knees, impossible torso orientations, or limbs that drift through the ground plane [21]. These artefacts are not penalised by MPJPE on Human3.6M, where such extreme poses simply do not appear, but they directly undermine the reliability of any downstream biomechanical or tactical analysis.

The MPJPE Metric. The field has converged on Mean Per Joint Position Error (MPJPE), introduced alongside Human3.6M [17], as the de facto evaluation standard [23, 40, 44, 46]. MPJPE measures the average Euclidean distance in millimetres between predicted and ground-truth joint positions after aligning the root joint (typically the pelvis), isolating the accuracy of the predicted body configuration from global position uncertainty. Formally, for a skeleton with N_S joints:

$$\text{MPJPE} = \frac{1}{N_S} \sum_{i=1}^{N_S} \|\hat{\mathbf{p}}_i - \mathbf{p}_i\|_2 \tag{2}$$

where $\hat{\mathbf{p}}_i, \mathbf{p}_i \in \mathbb{R}^3$ are the predicted and ground-truth positions of joint i relative to the root, averaged over all frames of a sequence. An MPJPE of roughly 40 mm is broadly considered the threshold at which predictions become useful for biomechanical analysis; state-of-the-art models on Human3.6M now score in the high 30s (Table 2). A Procrustes-aligned variant, PA-MPJPE, additionally removes scale and rotation differences; this thesis reports the more demanding standard MPJPE, which is directly relevant to the absolute joint localisation required for sports tracking.

These numbers should nonetheless be interpreted with caution: they reflect a constrained motion distribution, and state-of-the-art accuracy on Human3.6M walking sequences provides no guarantee of reliable performance on the high-speed, occluded, low-resolution player crops of real sports footage. This benchmark–deployment gap is a central motivation for the validation strategy adopted in Section 7.

A Persistent Training Data Problem. These limitations are not only an evaluation concern as they are baked into model weights. Virtually every state-of-the-art lifter, including MotionBERT despite its diverse pre-training, converges its 3D capability through Human3.6M fine-tuning [23, 40, 44, 46], anchoring its learned motion priors in slow, single-subject scenarios. Deployed on competitive sport, such models produce temporal jitter and foot–ground contact violations regardless of which architecture is selected [21, 30]. Addressing this requires a dedicated post-processing stage that imposes the temporal smoothness and physical plausibility the lifting

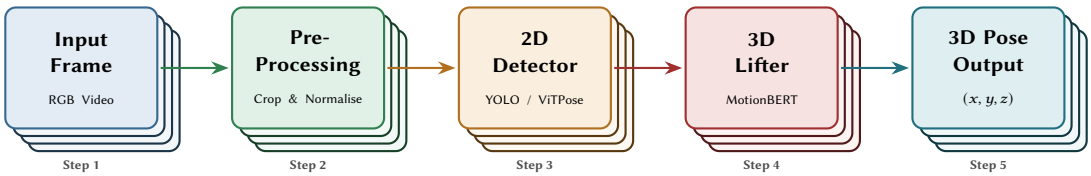

Figure 1: Overview of a generic modular 2D to 3D HPE lifting architecture.

model was never trained to provide. The design of such a stage is a central contribution of this thesis, and its motivation flows directly from the benchmark gap documented here.

2.5 Common 2D Detection Methods

The quality of any 3D pose estimation pipeline is fundamentally bounded by its upstream 2D keypoint detections. Errors introduced at the detection stage propagate directly and irreversibly into the lifting stage: a lifting network operating on noisy or anatomically inconsistent 2D inputs cannot recover the information that was lost at the detection boundary.

Three families of methods are relevant to this thesis. Top-down transformer-based estimators such as ViTPose-H [36] achieve state-of-the-art keypoint localisation accuracy by operating on tightly cropped person regions, at the cost of depending on an upstream person detector. Real-time alternatives such as RTMPose [7] sacrifice some accuracy for deployment efficiency, processing each frame in isolation and therefore exhibiting temporal jitter under fast motion. Single-stage unified detectors such as YOLO26 [28] handle detection and keypoint estimation simultaneously in one forward pass, prioritising speed and simplicity at the cost of fine-grained localisation in crowded scenes. All three were trained predominantly on general-purpose datasets such as H36M, that underrepresent the conditions characteristic of in-the-wild sports footage, and all are susceptible to the hallucination and jitter failure modes that motivate the refinement contribution of this thesis [31]. The specific choice between these approaches and the empirical evidence that informed it are discussed in Section 4.2.

2.6 Multi-Person Tracking and Re-Identification

Detecting the 2D keypoints of a single isolated person is a well-studied problem. The challenge grows considerably harder when the goal is to track multiple people simultaneously across a video, which is precisely the situation in broadcast football. The system must not only localise the joints of every visible player in every frame, but also answer a second equally important question: which detections across consecutive frames belong to the same person? Without a reliable answer, the temporal keypoint sequences fed to the lifting model are meaningless as there is no guarantee that frame t and frame $t + 1$ describe the same individual.

Motion-Based Tracking. The standard foundation for multi-object tracking is the Kalman filter [13], which maintains a running estimate of each player's position and velocity and uses this to predict where the player will appear in the next frame. When new detections arrive, each is matched to the nearest predicted location and the estimate is updated. This allows the tracker to bridge short gaps caused by missed detections and suppress spurious false positives. Its limitation is reliance on motion continuity: when two players cross paths or are occluded for more than a handful of frames, the motion model alone is insufficient to maintain correct identities.

ByteTrack [42] addresses one of the most persistent Kalman filter failure modes: the tendency to discard low-confidence detections entirely. In crowded sports footage, genuine players are frequently assigned low confidence scores during occlusion, precisely when maintaining track continuity matters most. ByteTrack performs a two-stage association: high-confidence detections are matched first in the standard way, then unmatched tracks are given a second chance using the low-confidence detections that were initially set aside. A spurious background detection is filtered out because it will not overlap with the Kalman-predicted location of the unmatched track, whereas a genuinely occluded player will. This substantially reduces broken trajectories in multi-player sports scenarios.

Appearance-Based Re-Identification. Even a robust tracker like ByteTrack eventually loses a player after a sufficiently long occlusion or when they leave the frame entirely. When the player reappears, motion alone cannot reconnect the new detection to the old track. Person re-identification addresses this by learning a visual appearance model for each individual so that players can be matched based on how they look rather than where they are.

OSNet [45] achieves this through omni-scale feature learning: rather than characterising appearance at a single spatial scale, it simultaneously captures fine-grained local details such as boot colour and shirt markings alongside coarser global characteristics such as overall silhouette. The result is a compact model of around 2.2 million parameters that produces discriminative appearance embeddings suitable for matching players across temporal gaps. In broadcast football, where players on the same team wear identical kits, combining OSNet's appearance embeddings with ByteTrack's motion-based association produces a substantially more robust pipeline than either approach alone. The specific design and configuration of this tracking system is described in Section 4.3.

2.7 Field-Space Projection

Once players have been detected and tracked in the image plane, translating their pixel coordinates into real-world pitch coordinates is necessary for any downstream analysis involving distances, formations, or trajectories. This thesis uses a planar homography [13]: given that a football pitch is a flat surface, a fixed mathematical mapping between image points and pitch coordinates can be estimated from a small number of known pitch markings, which include penalty spots, corner arcs, the center circle, and each player's foot position, to produce a metrically accurate top-down view of the pitch.

2.8 From Pipeline Output to Refined Motion

The components introduced above each perform well in isolation. Together, however, they expose a common problem: the raw output of such a pipeline on broadcast football footage is rarely clean enough for downstream analysis. Errors accumulate across stages, producing skeletal tracks that flicker, drift, and contort into anatomically impossible configurations.

Heuristic smoothing [29] reduces jitter but cannot distinguish genuine rapid motion from noise. Biomechanical constraints [21] improve plausibility but operate frame-by-frame. Learned temporal refinement [46] addresses both, and a Transformer [33] is the natural architecture, its full-window attention distinguishes genuine fast movements from spurious detection errors in a way no frame-by-frame method can. This motivates the primary contribution of this thesis, introduced in the next Section 3.

Related Work

Transformer-based 3D lifting. The two-stage lifting paradigm has been driven primarily by transformer-based lifters operating on 2D coordinate sequences, as discussed in 2.3. PoseFormer [44] established the transformer baseline at 44.3 mm MPJPE on Human3.6M, with MHFormer [23], MixSTE [40], and MotionBERT [46] progressively reducing error to 43.0, 39.8, and 37.5 mm respectively (Table 2). These gains are achieved entirely on controlled laboratory data, and do not transfer reliably to "in-the-wild" broadcast sports footage [37].

Sports-domain deployment. Yeung et al. [37] introduce AthletePose3D, the first large-scale benchmark capturing high-speed athletic movements across 12 sports, and demonstrate that MotionAGFormer trained on Human3.6M alone achieves 257 mm MPJPE on athletic motions, a more than six-fold degradation relative to its laboratory performance. Fine-tuning on AthletePose3D reduces this to approximately 98 mm, a 62% improvement, underscoring the severity of the benchmark–deployment gap and the need for sports-specific training data or correction mechanisms.

Post-hoc learned refinement. BioPose [21] demonstrates that a dedicated refinement stage operating on top of an initial mesh recovery model can achieve biomechanical accuracy competitive with marker-based motion capture on controlled single-subject footage. This validates the broader principle that post-hoc learned correction is a viable alternative to end-to-end retraining, the principle this thesis applies to the multi-player broadcast football setting.

3 OVERVIEW

The background chapter established that applying state-of-the-art monocular 3D HPE components to broadcast football footage produces skeletal tracks that are temporally unstable, anatomically implausible, and unreliable for downstream analysis. The proposed system, illustrated in Figure 2, responds by extending the generic two-stage pipeline in two directions.

First, the monolithic detection block is decomposed into three sub-modules: player segmentation, multi-object tracking, and 2D keypoint estimation, each with a clearly separated responsibility so that a failure in one does not contaminate the others.

Second, three refinement stages bracket the lifting backbone. Pre-lifting refinement cleans the 2D input before it reaches the lifting model. Post-lifting refinement enforces frame-level anatomical plausibility, the point at which a standard pipeline would terminate. The spatio-temporal pose refinement Transformer, the primary contribution of this thesis, then operates on the full temporal sequence, learning what physically consistent football motion looks like from game-engine ground truth and correcting the systematic errors that no deterministic rule can address. Each stage is described in detail in Section 4.

4 DESIGN

4.1 Design Philosophy: The Case for a Modular Two-Stage Lifting Approach

The choice of a modular two-stage lifting architecture is not made by convention but is a direct consequence of the constraints imposed by broadcast football footage. Single-stage systems such as RTMW3D [18] infer 3D joint positions directly from image crops in a single forward pass. This rests on an implicit assumption that the subject occupies a sufficiently large region of the image for fine-grained joint localisation to be feasible, an assumption that fails on "in-the-wild" footage.

Standard benchmark crops such as those used to train MMPose [7] usually show subjects at close range with every joint clearly resolvable. Players in our footage span approximately 50–200 pixels in height. At this scale the fine-grained appearance features that single-stage models rely upon simply do not exist in the image. A system asked to perform single-stage 3D inference on a 60-pixel-tall crop is operating entirely outside its training distribution.

The two-stage approach solves this by decoupling localisation from joint estimation: a dedicated detection stage first isolates each player into a normalised, resolution-corrected crop, and only then is joint-level inference attempted. It also offers a generalisation advantage as the lifting stage operates on coordinate sequences that do not depend on visual appearance rather than raw pixels, making it substantially more robust to the domain shift between laboratory training data and broadcast deployment. The empirical evidence for this, drawn from our own footage, is presented in Section 4.2.

4.2 2D Detection and Tracking

4.2.1 Player Segmentation: From YOLO to SAM 3. The first implementation decision concerns the detection frontend. The natural candidate, given its widespread use in real-time object detection, is a YOLO-based detector [28]. However, evaluation on our broadcast footage revealed systematic failure modes that disqualify it as a reliable foundation. As shown in Figure 3(a), YOLO drops the goalkeeper entirely despite remaining visible in the frame, caused by a confidence drop induced by an atypical posture. This track loss breaks temporal continuity at exactly the moment it is hardest to recover from, propagating corrupted input sequences to the lifting model.

The root cause is architectural. As a single-stage anchor-based detector, YOLO must simultaneously localise and classify players in a single forward pass over wide-angle footage where players occupy few pixels. At this scale, player boundaries blur, appearance features are impoverished, and confidence becomes unstable. Single-stage pose approaches such as YOLO-Pose [25] inherit all of these localisation failures and compound them with joint-level inference at broadcast scale.

SAM 3 as the Segmentation Frontend. The solution is architectural: split detection into two explicit responsibilities, first produce a reliable pixel-level mask per player, then pass each cropped region independently to the keypoint estimator. This ensures the keypoint estimator always receives a normalised, tightly bounded crop centred on a single real player.

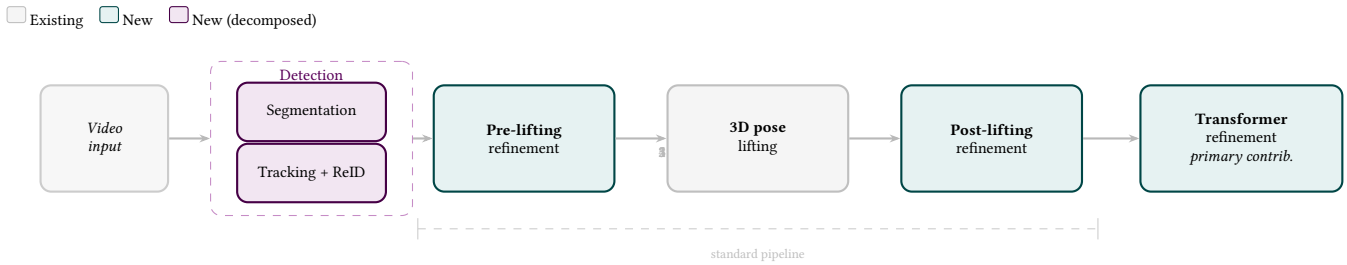

Figure 2: High-level overview of the proposed pipeline. The detection module (Stage 1) is a decomposition of the single block found in the generic architecture of Figure 1 into explicit segmentation and tracking sub-modules. Teal blocks indicate the three new refinement stages introduced in this thesis. The dashed bracket marks the extent of a standard two-stage lifting pipeline. Each stage is described in detail in Section 4.

(a) Track loss (b) SAM3 Track caught

Figure 3: Failure modes observed in YOLO-based detection [28] applied to broadcast football footage. (a) The goalkeeper is dropped entirely by the tracker despite remaining clearly visible in the frame, caused by a confidence drop induced by an atypical player posture. (b) SAM 3 successfully re-acquires and tracks the same player correctly. Together, these failure modes motivate the adoption of the SAM 3-based segmentation pipeline introduced in Section 4.2.

The model adopted for this masking step is SAM 3 (Segment Anything with Concepts) [3]. Given a natural language prompt such as `"soccer player"`, SAM 3 automatically detects and segments all matching instances, producing pixel-level masks without per-frame initialisation and preserving instance identities across frames. Its text-prompted, concept-level segmentation is substantially less sensitive to the appearance variations that cause YOLO's confidence to collapse, which is directly addressing the track loss shown in Figure 3(a), with successful tracking demonstrated in Figure 3(b).

From Mask to Crop. Once SAM 3 produces a mask for each detected player, it is converted to an axis-aligned bounding box expanded by a scale factor of 1.1 to prevent joint clipping at the crop boundary. The crop is passed to the 2D keypoint estimator (Section 4.4), and the mask is forwarded to the multi-object tracker (Section 4.3). By separating *who is in the frame* from *where their joints are*, the pipeline eliminates an entire class of downstream errors before they can propagate.

4.3 Multi-object tracking and re-identification.

With reliable player masks produced by SAM 3, the next challenge is maintaining consistent player identities across the video sequence. In broadcast football this is a correctness requirement rather than a convenience: a tracker that loses or confuses identities introduces silent discontinuities into the temporal keypoint sequences fed to the lifting model, which cannot recover from a window that switches between two different individuals mid-sequence. The tracking system must therefore be robust to rapid player motion, frequent inter-player occlusion, extended periods off-camera, and the additional difficulty that players on the same team wear identical kits.

ByteTrack and OSNet Configuration. As introduced in Section 2.6, the tracking backbone is ByteTrack [42], augmented with OSNet-x0.25 [45] appearance embeddings for re-identification. ByteTrack's two-stage association is configured with a high-confidence IoU threshold of 0.5 for the primary association pass and a relaxed threshold of 0.1 for the low-confidence recovery pass. Intersection-over-Union (IoU) [10] measures the ratio of the overlapping area between a predicted and a detected bounding box to their combined area, producing a score between 0 and 1 where a score of 1 indicates a perfect spatial match. A threshold of 0.5 therefore requires at least half of the bounding box area to overlap for a detection to be confidently matched to an existing track, while the relaxed threshold

of 0.1 allows partially occluded or low-confidence detections to be recovered against unmatched tracks in the second association pass. Unmatched tracks are kept alive for up to 30 frames using Kalman-filter prediction before being deleted, a horizon chosen to cover the typical duration of a player occlusion in broadcast footage without accumulating stale tracks. Tracks shorter than 10 frames at the conclusion of Stage 1 are discarded as likely false positives.

OSNet-x0.25 is selected over larger re-identification models deliberately. At 50–100 pixels of player height, the appearance detail available in each crop is limited, and a lightweight model that produces robust embeddings from low-resolution inputs is preferable to a heavier model that relies on fine-grained appearance features that are simply not present at broadcast scale. Track association combines both spatial overlap and appearance similarity into a composite cost matrix that is minimized to find the optimal assignment between existing tracks and new detections. The matrix deliberately weights appearance over geometry in broadcast football, where player trajectories frequently cross and motion is rapid. Appearance similarity is a more reliable long-range discriminator than spatial proximity alone.

Fragment Linking. Even with two-stage association and appearance re-identification, track fragmentation occasionally occurs when a player is occluded for an extended period or briefly leaves the frame. A post-hoc fragment linking stage is therefore applied after all frames have been processed. Two track fragments are merged into a single continuous identity if their appearance similarity, computed within the cost matrix as a combination of the OSNet and ByteTrack scores, exceeds a minimum threshold and their field positions at the respective endpoints are within a maximum distance of one another. Pairs satisfying both conditions are merged with a linearly interpolated gap, producing a continuous track that the downstream pipeline treats as unbroken.

The output of this stage is a set of per-player bounding box sequences, each associated with a persistent identity, that are passed to the 2D keypoint estimator described in the following section.

4.4 From Segmentation to Keypoints: ViTPose-H

With each player isolated into an identity-associated crop, 2D keypoint estimation becomes far more tractable than on the full broadcast frame. Of the three families of keypoint estimators introduced in Section 2.5, bottom-up and single-stage approaches both face the resolution problem of Section 4.1: at 50–100 pixels of player height, the full frame does not contain the joint-level signal needed to localise wrists and ankles across dozens of players, and coupling joint estimation with detection additionally inherits the failure modes of Section 4.2.1. The top-down approach sidesteps both — the SAM 3 and ByteTrack stages have already reduced the problem to single-person, single-crop inference, precisely the setting in which top-down models are strongest.

The estimator adopted is ViTPose-H [36], the large-scale variant of the Vision Transformer-based model of Xu et al., which achieves state-of-the-art COCO keypoint accuracy while remaining fast enough for a multi-player pipeline. Its globally attending backbone is particularly suited to the top-down paradigm, resolving joints under partial occlusion by leveraging context from the visible body. In practice, each crop is resized to 192×256 pixels and forwarded to ViTPose-H via ONNX Runtime on the GPU; the model outputs 17 COCO-format keypoints with pixel coordinates (x, y) and a confidence score, and keypoints below confidence 0.3 are flagged for the interpolation mechanism of Section 4.5. The pipeline also supports RTMPose-x [7] and YOLO11x-Pose as faster alternative backends, but ViTPose-H is used exclusively in the reported experiments, as its accuracy gains are significant at the low player resolutions characteristic of our footage.

4.5 Stage 2: Pre-Lifting Refinement

The quality of the interface between the 2D detection stage and the 3D lifting stage determines the quality of the lifting output, errors introduced here are unrecoverable downstream. This refinement stage ensures the lifting model receives the cleanest possible input. Dedicated pre-processing is necessary because MotionBERT was pre-trained on motion-capture and in-the-wild sequences whose noise regime differs substantially from the gross failures common in broadcast footage: missing joints, bone-length violations, and frame-to-frame instability. Cleaning the input is therefore complementary to, rather than redundant with, the lifting model's own robustness mechanisms.

Online 2D Temporal Smoothing. Each keypoint coordinate is smoothed online using a 1-Euro filter [4], a causal adaptive low-pass filter that suppresses high-frequency detector noise while preserving genuine motion. Its cut-off frequency adapts to signal velocity, stronger smoothing during slow poses where jitter is most damaging, higher cut-off during fast motion to preserve the true trajectory. A separate filter instance is maintained per track and per keypoint so that smoothing state is never shared across players. Parameters are summarised in Table 4.

4.5.1 Low-Confidence Joint Interpolation. Joints that fall below a confidence threshold of 0.3 due to occlusion, motion blur, or detection failures are recovered via cubic spline interpolation [8] rather than discarded or zeroed. Each joint is scanned independently; gaps spanning no more than 10 frames with at least two valid observations on both sides are reconstructed with a smooth curve that preserves the velocity characteristics of the surrounding trajectory. Gaps longer than 10 frames are left as missing, as boundary observations no longer provide sufficient constraint beyond this horizon.

4.5.2 Bone-Length Validation. Gross detection failures can produce anatomically implausible joint configurations that confidence scores alone do not flag. Most commonly this takes the form of joint snapping, where partial occlusion causes the detector to place a joint at a geometrically plausible but physically inconsistent location. For each of the 12 standard COCO-17 bones, the per-frame bone length is compared against the median across the track. Frames where any bone deviates by more than a factor of 2.5 from the median, i.e.:

$$\ell_t > 2.5 \cdot \tilde{\ell} \quad \text{or} \quad \ell_t < \frac{\tilde{\ell}}{2.5} \tag{3}$$

are flagged and replaced by linear interpolation from the nearest valid neighbouring frames. Together, these three steps ensure that

the 2D sequences forwarded to the lifting model are clean, complete, and anatomically consistent.

4.6 Intermediate Step: Converting COCO-17 to Human3.6M-17

Before the cleaned 2D keypoint sequences can be forwarded to MotionBERT, a format conversion step is required. ViTPose-H produces keypoints in COCO-17 format, while MotionBERT [46] was trained on the Human3.6M dataset [17], which defines a different 17-joint skeleton with a distinct topology, joint ordering, and coordinate convention. Feeding COCO-17 sequences directly into MotionBERT would produce meaningless outputs, as the model would interpret each joint index according to its Human3.6M meaning rather than its COCO 17 meaning. A deterministic remapping is therefore applied to every 2D keypoint sequence before lifting.

Joint Remapping. The remapping is largely a reordering of existing joints, with two exceptions. The Human3.6M skeleton includes a *Pelvis* joint and a *Spine* joint that have no direct equivalent in COCO-17 and must therefore be synthesised. The Pelvis is computed as the midpoint of the left and right hip joints, and the Spine as the midpoint of the synthesised Pelvis and the Neck joint:

$$\mathbf{p}_{\text{Pelvis}} = \frac{\mathbf{p}_{\text{L-Hip}} + \mathbf{p}_{\text{R-Hip}}}{2} \tag{4}$$

$$\mathbf{p}_{\text{Spine}} = \frac{\mathbf{p}_{\text{Pelvis}} + \mathbf{p}_{\text{Neck}}}{2} \tag{5}$$

Both synthesised joints rest on a geometric approximation that holds for an upright torso and degrades under the bending and twisting typical of athletic motion. The Pelvis is the safer of the two, as the hips sit on a rigid structure the Spine midpoint is the weaker approximation. Both are retained because MotionBERT requires the full Human3.6M-17 topology and COCO-17 provides no alternative estimate. These two synthetic joints are necessary because the Human3.6M skeleton is rooted at the Pelvis and uses the Spine as an intermediate joint between the hips and the neck. Both of which are structurally important for the lifting model's spatial attention mechanism. All remaining joints are remapped by index reordering.

Table 3 provides the complete joint correspondence used in the conversion step described in this Section 4.6.

Root Centering and Axis Alignment. After remapping, two coordinate transforms are applied per frame to match the convention assumed by MotionBERT during training.

The first is **root centering**. The (x, y) coordinates of all 17 joints are shifted so that the Pelvis joint which takes index 0, serves as the root of the Human3.6M skeleton, which is being placed at the origin. For each frame t and joint j:

$$\tilde{\mathbf{p}}_{t,j} = \mathbf{p}_{t,j} - \mathbf{p}_{t,0} \tag{6}$$

where $\mathbf{p}_{t,0}$ denotes the Pelvis position at frame t. Root centering makes the representation translation-invariant with respect to the player's absolute position in the image, ensuring that the lifting model receives the same skeletal configuration regardless of where in the frame the player is located. This is important in broadcast footage where players move across large portions of the frame between windows.

The second transform is a **Y-axis flip**. Image coordinate systems are defined with the y-axis pointing downward, where the origin is at the top-left corner of the image and y increases toward the bottom. Three-dimensional world coordinates are conventionally Y-up. Since MotionBERT was trained under the Y-up convention, feeding it sequences in image convention would cause it to interpret all vertical relationships in the skeleton as inverted, producing 3D estimates in which the hips appear above the shoulders. All y coordinates are therefore negated after root centering:

$$\tilde{y}_{t,j} = -y_{t,j} \tag{7}$$

Critical Joint Confidence Gate. A final validity check is applied at the frame level following the coordinate transforms. The four joints that anchor the core of the Human3.6M skeleton such as, the left shoulder, right shoulder, left hip, and right hip, are treated as critical. They define both the root position and the torso orientation of the skeleton. If the mean confidence across these four joints falls below the threshold of 0.3 in a given frame, the entire frame is marked as invalid and replaced by linear interpolation from the nearest valid neighbouring frames. An unreliable torso estimate is particularly damaging in the root-centered representation: because all other joint positions are expressed relative to the Pelvis, an incorrect root or torso configuration propagates geometric errors to every joint in the skeleton.

The converted, root-centered, and validated sequences are now in the format expected by MotionBERT and are forwarded to the 3D lifting stage described in the following section.

4.7 Stage 3: 3D Pose Lifting with MotionBERT

With the 2D keypoint sequences cleaned, validated, and converted to Human3.6M-17 format, the pipeline is ready for 3D pose lifting. The lifting model adopted in this thesis is MotionBERT [46], and this section justifies that choice before describing the normalisation and inference procedure.

Why MotionBERT. A practically important but often overlooked property of 3D lifting models is their input interface. Many state-of-the-art lifting architectures are either tightly coupled to their own upstream detection pipeline, such as RTMPose3D, or MMPose for instance, [7, 18], accepting intermediate feature representations rather than raw keypoint coordinates, or require additional inputs such as camera intrinsics, pixel-level features, or depth estimates derived from the original image. These requirements are incompatible with the modular pipeline developed in this thesis as shown in Figure 2, where the upstream detection stage produces only 2D keypoint coordinate sequences from heavily pixelated, low-resolution player crops.

MotionBERT is one of the few high-performance lifting models that accepts raw 2D keypoint sequences as its sole input: no pixel features, no camera parameters, no auxiliary signals. This is not a limitation of the model but a deliberate design choice that directly benefits our pipeline: a model that operates purely on coordinate sequences is entirely agnostic to the visual quality of the upstream crops, and the distributional mismatch between our low-resolution

broadcast crops and the close-up laboratory crops of Human3.6M is therefore minimized as much as possible to the lifting stage. The unchanged form of coordinate sequences, first highlighted as a key advantage of the two-stage paradigm in Section 4.1, is fully exploited here.

Beyond its input interface, MotionBERT's Dual-stream Spatio-Temporal Transformer architecture processes sequences of 243 frames simultaneously, allowing it to resolve depth ambiguity through temporal context rather than single-frame appearance features. This represents a significant advantage in broadcast footage where individual frames frequently provide insufficient information for reliable depth inference. Its pre-training on both motion-capture data and in-the-wild video also makes it more robust to the noisy, imperfect 2D detections produced by our upstream pipeline than models trained exclusively on clean laboratory sequences.

Input Normalisation. Before the converted keypoint sequences are forwarded to MotionBERT, a normalisation step is applied to match the input distribution assumed during the model's training. Two normalisation modes are implemented in the pipeline, and the choice between them has a significant and non-obvious impact on lifting quality in the broadcast domain.

The first mode, and the default, is **crop-scale normalisation**. The root-centered keypoint sequence is scaled so that the bounding box of all joint positions across the entire sequence fits within the range $[-1, 1]$. This is the same normalisation applied by MotionBERT's own inference script [46], and it ensures that the model receives sequences whose scale is consistent with it.

4.7.1 MotionBERT Sliding-Window Inference. MotionBERT operates on fixed-length temporal windows of 2D pose sequences and regresses the corresponding 3D joint positions for each frame in the window. The model is used in global mode, meaning it estimates absolute 3D coordinates rather than pelvis-relative positions, which is an important distinction for downstream analysis where the absolute position of each player on the pitch is required.

Player tracks in our footage span up to 148,280 frames, far exceeding the model's fixed clip length of 243 frames. Three design decisions govern how the model is applied to tracks of arbitrary length: the sliding window strategy, test-time augmentation, and window merging.

Sliding Window with Overlap. The track is processed by sliding a window of fixed length $T_{\text{clip}} = 243$ frames (≈ 9.7 seconds at 25 fps) along the sequence with a stride of 121 frames, corresponding to a 50% overlap between consecutive windows. The overlap is deliberate: a Transformer attending over a 243-frame window produces its most reliable estimates for frames near the centre of the window, where the model has access to the maximum amount of surrounding temporal context. Frames near the window boundaries, by contrast, have access to context on only one side and are consequently less reliable. By overlapping consecutive windows by 50%, every frame in the sequence, except those at the very beginning and end of the track, appears near the centre of at least one window, ensuring that the final merged output benefits from a well-grounded prediction for every frame. When the final window would extend beyond the end of the track, it is padded by repeating the final frame until the full clip length is reached.

Test-Time Flip Augmentation. For each clip, MotionBERT is run twice: once on the original sequence and once on a horizontally mirrored copy in which the spatial x-coordinates are negated and the left and right joint indices are swapped to reflect the bilateral symmetry of the Human3.6M skeleton.The mirrored prediction is then unflipped and averaged with the original prediction. This test-time augmentation reduces the systematic left-right asymmetry bias that can arise from asymmetries in the Transformer's learned attention patterns, which are a known artefact of models trained on datasets in which subjects face a preferred direction, such as H36M [17]. The cost is a doubling of inference time per clip, which is acceptable in the offline processing context of this pipeline.

Gaussian-Weighted Window Merging. After MotionBERT inference, the predictions from overlapping windows must be blended into a single continuous 3D sequence. A naive approach, which consists of simply taking the prediction from the most recent window for each frame and produces visible discontinuities at window boundaries, where the model's estimate jumps as a new window takes over. These discontinuities are particularly damaging for the downstream Transformer refinement stage, which expects a temporally smooth input sequence.

To eliminate boundary artefacts, overlapping window predictions are merged using a Gaussian-weighted blend. Each window contributes to the shared overlap region with a weight proportional to a Gaussian profile centred on the window's middle frame:

$$w(t) = \exp\left(-\frac{(t - t_{\text{mid}})^2}{2\sigma^2}\right), \quad \sigma = \frac{T_{\text{clip}}}{4} \tag{8}$$

where t_{mid} is the index of the central frame of the window and σ is set to one quarter of the clip length. Frames near the centre of a window receive high weight, reflecting the greater reliability of centre-frame predictions, while frames near the edges receive low weight and are therefore dominated by predictions from neighbouring windows in which those frames appear more centrally. The final 3D position for each frame is the normalised weighted average across all windows that cover it:

$$\hat{\mathbf{P}}_t = \frac{\sum_k w_k(t) \cdot \mathbf{P}_t^{(k)}}{\sum_k w_k(t)} \tag{9}$$

where $\mathbf{P}_t^{(k)}$ denotes the 3D joint prediction for frame t from window k and $w_k(t)$ is the corresponding Gaussian weight. The result is a single, smoothly blended 3D sequence that transitions continuously across window boundaries without hard cuts or discontinuities.

The output of the lifting stage is a per-player sequence of absolute 3D joint positions in Human3.6M-17 format, one per frame, which is forwarded to the post-lifting refinement stage described in the following section.

4.8 Stage 4: Post-Lifting Refinement

The raw MotionBERT output exhibits three classes of artefact requiring correction: high-frequency temporal noise from per-frame prediction instability, isolated frame-level outliers caused by gross lifting failures, and anatomically implausible configurations from the distributional mismatch between Human3.6M training data and athletic motion. Four successive passes address these in the following order.

4.8.1 Butterworth Temporal Smoothing. A zero-phase second-order Butterworth low-pass filter [2], which attenuates signal components above a specified cut-off while passing lower frequencies without distortion, is applied independently to each joint coordinate. The forward-backward `sosfiltfilt` implementation introduces zero temporal lag, preserving alignment between the 3D output and the video frames. A cut-off of 12 Hz is consistent with established practice in football biomechanics [5] and is applied only to tracks of at least 12 frames. Filter parameters are summarised in Table 5.

4.8.2 Velocity and Acceleration Outlier Rejection. Butterworth smoothing weakens distributed noise but leaves isolated frame-level outliers, gross lifting failures on individual frames caused by sudden occlusion or out-of-distribution poses that are only partially suppressed. A dedicated two-pass rejection stage flags these after smoothing.

For each joint independently, the frame-to-frame displacement magnitude is compared against three times the running median; frames exceeding this threshold are flagged as velocity outliers. A second pass computes the second-order finite difference as a stepwise acceleration estimate and flags frames exceeding four times the median acceleration. The higher multiplier reflects the fact that genuine athletic movements can produce large instantaneous accelerations that should not be flagged.

Flagged frames are replaced using a cubic Hermite spline [8] which interpolates between boundary points while matching the first derivative at each endpoint, preserving velocity continuity with tangents estimated from the surrounding unflagged frames via numerical gradient. Linear interpolation is used as a fallback where fewer than four good surrounding frames are available.

4.8.3 Bone-Length Enforcement. As discussed in the previous Section 4.5.2, bone lengths are an anatomical invariant that should remain constant across a sequence. MotionBERT can nonetheless produce time-varying bone lengths in the 3D output due to depth ambiguity as the same 2D projection is consistent with multiple 3D configurations, and the model may resolve this ambiguity differently across frames. This pass enforces temporal consistency by clamping each of the 16 Human3.6M bones to within $[\,0.7\times,\ 1.3\times\,]$ of its sequence-median length. Where a violation is detected, the positional error is split equally between the parent and child joint, except for bones originating at the root pelvis joint where only the farthest joint is adjusted, preserving the stability of the root position on which all other joints depend.

A bilateral symmetry pass is applied immediately after. For five symmetric bone pairs hip-to-knee, thigh, shin, upper arm, and forearm frames where one limb deviates by more than 15% from the bilateral mean length are snapped to that mean. This exploits the fact that a healthy player's left and right limb segments are approximately equal in length, and that asymmetric deviations are more likely to reflect depth estimation errors than genuine anatomical differences.

4.8.4 Joint Angle Limit Enforcement. The final post-lifting refinement pass enforces biomechanical plausibility at the level of individual joint angles. Even after smoothing, outlier rejection, and bone-length enforcement, MotionBERT can produce joint configurations that violate the physical range of motion of the human body including: hyperextended knees, impossibly wide shoulder abductions, or collapsed spinal angles. These are particularly often produced when the lifting model encounters pose configurations that fall outside its training distribution. These violations are not merely cosmetic: anatomically implausible joint angles in the input to the Transformer refinement stage introduce a class of error that the model was not trained to correct, as the game-engine ground truth from which it learns never exhibits such configurations.

For each of 10 anatomically constrained joint triples, the joint angle is computed from the three underlying joint positions and checked against the biomechanical range of motion limits summarised in Table 6. Where a violation is detected, the angle is corrected using the Rodrigues rotation formula [13], applied to the distal limb segment. The Rodrigues rotation formula provides a computationally efficient way to rotate a vector by a specified angle around an arbitrary axis without constructing a full rotation matrix. Given a unit rotation axis $\hat{\mathbf{k}}$ and a rotation angle θ, the rotated vector $\mathbf{v}'$ is:

$$\mathbf{v}' = \mathbf{v}\cos\theta + (\hat{\mathbf{k}} \times \mathbf{v})\sin\theta + \hat{\mathbf{k}}(\hat{\mathbf{k}} \cdot \mathbf{v})(1 - \cos\theta) \tag{10}$$

In this context, the rotation axis is the vector perpendicular to the plane defined by the three joint positions of the constrained triple, the joint centre is kept fixed, and only the child joint position is rotated until the angle is brought within the allowed range. This ensures that the correction is minimal and that only the violated joint is adjusted, by the smallest angle necessary to restore plausibility.

A fallback is applied for the special case where the three joints are nearly collinear, making the rotation axis numerically indeterminate. In this case, scaled planar interpolation is used to move the child joint to the nearest plausible position in the plane of the surrounding skeleton segments.

With the four post-lifting refinement passes complete: Butterworth smoothing, outlier rejection, bone-length enforcement, and joint angle correction, the 3D pose sequences are as clean and anatomically consistent as fixed post-processing can make them. This is the point at which a standard pipeline would terminate. The following section introduces the learned refinement stage that takes these sequences as input and imposes the temporal structure that no fixed pass can provide.

4.9 Stage 5: The Spatio-Temporal Pose Refinement Transformer

The four post-lifting refinement passes described in Section 4.8 address the most severe artefacts in the MotionBERT output through fixed, rule-based corrections. However, they share a fundamental limitation: they operate either on individual frames in isolation or on single joints independently, with no mechanism to reason about the relationship between joint positions across the full temporal context of a player's movement. A heuristic filter cannot distinguish between a genuine rapid change of direction and a spurious single-frame outlier. A bone-length enforcement pass cannot correct a systematic depth bias that gradually appears across dozens of frames. These are failure modes that require learned temporal reasoning, not hand-crafted rules.

The primary contribution of this thesis is therefore a supervised Spatio-Temporal Pose Refinement Transformer trained on game-engine ground truth. The model takes the post-processed MotionBERT output as its noisy input and learns a correction function that maps it toward clean, temporally coherent, physically consistent 3D pose sequences. This section describes the architecture, input encoding, loss function, and key design decisions of this model in detail.

The Transformer implemented here is based on the Transformer architecture of Vaswani et al. [33], in which multi-head self-attention allows each token to attend to every other token in the sequence simultaneously, learning context-aware representations without the sequential processing constraints of recurrent models.

4.9.1 Design Philosophy: Residual Correction. The model is designed as a **residual corrector** rather than a pose predictor. Rather than predicting absolute 3D joint positions from scratch, the model predicts a delta correction Δ that is added to the noisy input:

$$\hat{\mathbf{X}} = \mathbf{X} + \Delta \tag{11}$$

where $\mathbf{X} \in \mathbb{R}^{B\times W\times J\times 3}$ is the noisy input sequence and $\hat{\mathbf{X}}$ is the refined output. The output projection layer is initialised with near-zero weights, meaning that at the start of training $\Delta \approx 0$ and the model acts as the identity function. Training then progressively moves the corrections toward meaningful improvements.

This design choice is motivated by the quality of the upstream pipeline. The post-processed MotionBERT output, while imperfect, already provides a reasonable estimate of the player's 3D configuration. Asking the model to predict absolute poses from scratch would require it to learn the full distribution of valid 3D human motion, which is a much harder learning problem than learning the *difference* between noisy monocular estimates and clean ground truth. The residual formulation also provides a natural safety property: A model that hasn't learned anything useful yet will simply reproduce its noisy input as output, which serves as a meaningful and reliable baseline behavior.

4.9.2 Input Specification and Encoding. Each forward pass operates on a window of $W = 243$ frames and receives three input tensors, illustrated in Figure 4:

- $\mathbf{X} \in \mathbb{R}^{B\times W\times J\times 3}$: the noisy 3D pose sequence, root-relative (pelvis at origin), with missing joints zeroed.
- $\mathbf{M}_{\text{in}} \in \mathbb{R}^{B\times W\times J}$: a binary validity mask where 1 indicates a reliably detected joint and 0 indicates an occluded, missing, or otherwise unreliable joint.
- $\mathbf{r} \in \mathbb{R}^{B\times W\times 4}$: a root context vector containing the player's field position in metres (channels 0–1) and frame-to-frame velocity (channels 2–3), computed via forward finite difference.

These three streams are projected into the model's embedding space of dimension $D = 256$ and combined before entering the encoder, as illustrated in Figure 4.

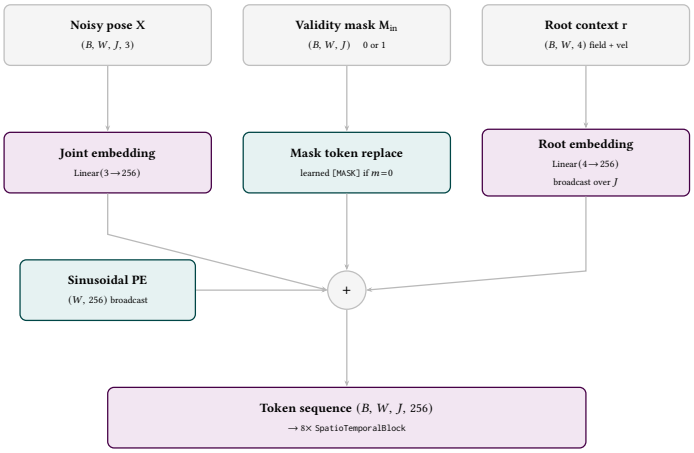

Figure 4: Input encoding for the Spatio-Temporal Pose Refinement Transformer. Three streams — noisy pose coordinates with joint embedding, a validity mask with learnable `[MASK]` token replacement for occluded joints, and root context providing global field position and velocity, are independently projected into the 256-dimensional embedding space, summed with sinusoidal positional encodings, and forwarded to the encoder stack.

Joint coordinate embedding. Each (x, y, z) coordinate triplet is independently projected via a linear layer: Linear$(3 \to 256)$. Joints where $\mathbf{M}_{\text{in}} = 0$ are replaced by a learnable `[MASK]` token, a single trainable vector of dimension 256, before this projection. This is a critical design decision: without the mask token, a zero-valued missing joint is indistinguishable from the Pelvis joint at the root-centered origin. The mask token gives the model an explicit *"I know this joint is missing"* signal rather than a silent zero that the model would have no way of identifying as absent.

Root context embedding. The per-frame field position and velocity are projected via Linear$(4 \to 256)$ and broadcast across all $J = 17$ joints, adding a global motion and position context to every token in the frame. This conditioning is important for correcting systematic errors that depend on the player's absolute position on the pitch, such as for instance, camera projection artefacts that show themselves differently near the sidelines than near the centre of the field.

Temporal positional encoding. Sinusoidal positional encodings of shape $(W, 256)$ are added once per frame and broadcast across all joints, following the encoding scheme introduced by Vaswani et al. [33]. Sinusoidal rather than learned encodings are used deliberately: learned positional embeddings are tied to the specific window length seen during training, whereas sinusoidal encodings generalize to unseen sequence lengths, providing flexibility for future deployment at different frame rates or window sizes.

The three streams are summed to produce the final token sequence of shape $(B, W, J, 256)$, which is forwarded to the encoder stack.

4.9.3 The Spatio-Temporal Encoder. The encoder consists of $N = 8$ stacked `SpatioTemporalBlock` modules, each illustrated in Figure 10. Every block alternates between two attention phases that are kept explicitly separate: spatial attention operating across joints within each frame, and temporal attention operating across frames within each joint. This alternating design is a deliberate architectural choice: the two reasoning tasks — enforcing anatomically consistent joint relationships within a frame, and smoothing each joint's trajectory across time — are complementary but distinct, and separating them is both computationally more efficient than full 3D attention over the $(W \times J)$ token sequence and more interpretable, as each phase has a clearly defined responsibility.

Spatial attention. The token sequence is reshaped from (B, W, J, D) to $(B \cdot W, J, D)$ so that multi-head self-attention operates across all $J = 17$ joints simultaneously within each frame. Eight attention heads are used with a head dimension of $d_{\text{head}} = 32$, giving a total model dimension of $D = 256$. A **learnable skeleton adjacency bias** matrix of shape $(J \times J)$ is added to the attention logits before the softmax operation. At initialisation, anatomically adjacent joints (1-hop neighbours in the Human3.6M skeleton graph, such as the knee and ankle) receive a bias of +2.0, and 2-hop neighbours (such as the hip and ankle) receive a bias of +1.0. All other pairs receive a bias of 0. This injects anatomical structure as a soft prior into the attention mechanism: connected joints are encouraged to attend to one another more strongly than unconnected ones, but the model is free to override this prior during training if the data supports different attention patterns. This is preferable to hard-coding the skeleton connectivity as a mask, which would prevent the model from learning long-range joint correlations that the fixed graph topology does not capture.

Formally, the biased attention logit for joint pair (i, j) is:

$$e_{ij} = \frac{\mathbf{q}_i \cdot \mathbf{k}_j}{\sqrt{d_{\text{head}}}} + b_{ij} \tag{12}$$

where $b_{ij} \in \{0, 1, 2\}$ is the learnable adjacency bias for the joint pair.

Temporal attention. After the spatial sub-layer, the sequence is reshaped from (B, W, J, D) to $(B \cdot J, W, D)$ so that multi-head self-attention operates across all $W = 243$ frames simultaneously for each joint independently. The same head configuration is used as in spatial attention, but without the adjacency bias, as temporal context is sequential rather than graph-structured. Full self-attention over the entire 243-frame window allows the model to attend to any frame in the sequence when making its prediction for any given frame, giving it access to the full 9.7 seconds of temporal context simultaneously.

Sub-layer structure. Both attention phases follow the same pre-norm residual pattern: LayerNorm is applied to the input before each sub-layer, and the sub-layer output is added back to the input via a residual connection. Each attention phase is followed by a feed-forward network consisting of two linear layers with a GELU activation [14] and dropout of 0.2:

$$\begin{aligned}\text{FFN}(\mathbf{x}) = \text{Linear}(1024 \to 256)\\ \big(\text{Dropout}(\text{GELU}(\text{Linear}(256 \to 1024)(\mathbf{x})))\big)\end{aligned} \tag{13}$$

4.9.4 Output Head. After the 8 encoder blocks, a final LayerNorm is applied and the token sequence is projected from $D = 256$ to 3 dimensions via a linear layer, producing the delta correction $\Delta \in \mathbb{R}^{B \times W \times J \times 3}$. The refined pose is then:

$$\hat{\mathbf{X}} = \mathbf{X} + \Delta \tag{14}$$

The output projection is initialised with near-zero weights so that $\Delta \approx 0$ at the start of training, ensuring a stable initialisation as discussed in Section 4.9.1.

4.9.5 Loss Function. Training is supervised using paired sequences of noisy pipeline poses $\mathbf{X}$ and clean game-engine ground truth $\mathbf{Y}$. All loss terms are masked by the ground truth validity mask $\mathbf{M}_{\text{gt}}$, ensuring that gradient flows only through frames and joints where reliable ground truth is available. The two joints that cannot be derived from the game-engine skeleton — Nose (joint 9) and Head (joint 10) — are always excluded from the loss.

The total loss is a weighted sum of four complementary terms:

$$\mathcal{L} = \mathcal{L}_{\text{pose}} + 0.1\,\mathcal{L}_{\text{vel}} + 0.05\,\mathcal{L}_{\text{bone}} + 0.02\,\mathcal{L}_{\text{symm}} \tag{15}$$

Pose loss $\mathcal{L}_{pose}$. Masked mean squared error between predicted and ground truth joint positions:

$$\mathcal{L}_{\text{pose}} = \frac{\sum_{t,j} \|\hat{\mathbf{x}}_{t,j} - \mathbf{y}_{t,j}\|^2 \cdot \mathbf{M}^{\text{gt}}_{t,j}}{N_{\text{valid}}} \tag{16}$$

This is the primary loss term and carries the highest weight. It directly supervises the spatial accuracy of each predicted joint position.

Velocity loss $\mathcal{L}_{vel}$. MSE on the frame-to-frame joint displacements:

$$\mathcal{L}_{\text{vel}} = \frac{\sum_{t,j} \|(\hat{\mathbf{x}}_{t+1,j} - \hat{\mathbf{x}}_{t,j}) - (\mathbf{y}_{t+1,j} - \mathbf{y}_{t,j})\|^2 \cdot \mathbf{M}^{\text{vel}}_{t,j}}{N_{\text{vel}}} \tag{17}$$

Supervising temporal derivatives encourages the model to produce smooth, temporally consistent motion trajectories rather than independently correcting each frame. $\mathbf{M}^{\text{vel}}_{t,j}$ is active only where both consecutive GT frames are valid.

Bone length loss $\mathcal{L}_{bone}$. MSE on predicted versus ground truth bone lengths across all 16 Human3.6M bones:

$$\mathcal{L}_{\text{bone}} = \sum_{(i,j)\in\mathcal{B}} \big(\|\hat{\mathbf{x}}_i - \hat{\mathbf{x}}_j\| - \|\mathbf{y}_i - \mathbf{y}_j\|\big)^2 \cdot \mathbf{M}^{\text{bone}} \tag{18}$$

where $\mathcal{B}$ is the set of bone pairs in the Human3.6M skeleton. This prevents the model from producing implausible bone stretches as a side effect of its corrections.

Symmetry loss $\mathcal{L}_{symm}$. An L2 penalty on the length difference between symmetric left and right limb bones:

$$\mathcal{L}_{\text{symm}} = \frac{1}{|\mathcal{S}|} \sum_{(l,r)\in\mathcal{S}} \Big(\|\hat{\mathbf{b}}_l\| - \|\hat{\mathbf{b}}_r\|\Big)^2 \tag{19}$$

where $\mathcal{S}$ is the set of four symmetric bone pairs (left/right thigh, shin, upper arm, forearm). This encourages bilateral anatomical symmetry in the predicted skeleton.

The weighting of the four terms reflects their relative importance: pose accuracy ($\lambda = 1.0$) is the primary objective, velocity smoothness ($\lambda = 0.1$) provides temporal regularisation, and bone integrity ($\lambda = 0.05$) and symmetry ($\lambda = 0.02$) act as lightweight anatomical constraints without dominating the primary signal.

4.9.6 Inference: Overlapping Window Strategy. At inference time, the full player trajectory is too long to process as a single sequence. The model is applied using overlapping windows of length 243 with a stride of 32 frames — a much shorter stride than the 50% overlap used during training, maximising the number of predictions that contribute to each frame. For each window, the model produces a delta correction that is accumulated in a per-frame buffer:

$$\hat{\mathbf{X}} = \mathbf{X} + \frac{\sum_k \Delta^{(k)}}{\sum_k \mathbf{1}^{(k)}} \tag{20}$$

where $\Delta^{(k)}$ is the delta from window k and $\mathbf{1}^{(k)}$ is a binary indicator of which frames window k covers. Because consecutive windows overlap heavily, each interior frame is refined by multiple independent predictions, and the per-frame average acts as a soft ensemble that reduces window-boundary artefacts. The model is never exposed to a full-sequence input during inference, it always operates on the same 243-frame window format it was trained on.

4.10 Field-Position Projection

Once per-player 3D joint sequences are available, each player's field position is projected onto a metrically accurate top-down pitch representation using a planar homography [13]. A fixed 3×3 homography matrix is estimated from a small set of known pitch markings that include penalty spots, corner arcs, and the centre circle, used to establish a correspondence between image-plane pixel coordinates and real-world pitch coordinates in metres. For each frame, the player's foot position, taken as the midpoint of the two ankle joints projected onto the ground plane, is mapped through this homography to produce a 2D field coordinate. The resulting per-player pitch trajectories are forwarded and serve as the root context signal provided to the Spatio-Temporal Pose Refinement Transformer.

5 DATA GENERATION AND GROUND TRUTH CONSTRUCTION

The supervised training of the Spatio-Temporal Pose Refinement Transformer requires paired sequences of noisy pipeline poses and clean ground-truth 3D annotations. Obtaining such data from real broadcast footage is not feasible, as live video provides no skeletal ground truth. This thesis therefore adopts a simulation-based approach, generating ground-truth data from a modified football game engine and aligning it with the outputs of the upstream pipeline.

5.1 Simulation Environment

Ground-truth 3D pose data is produced by a heavily modified Unity-based football simulation engine that renders matches with full per-player skeletal tracking, exporting world-space joint positions alongside a calibrated camera configuration. Alternative simulators, including Google Research Football [22] and a modified version of FIFA 16, were evaluated but discarded due to build incompatibilities and time constraints.

Several modifications were made to the simulator to produce training-compatible data. Player appearance was diversified by varying jersey colour, skin tone, and kit type across players, since the OSNet re-identification model requires visual discriminability between players to maintain stable tracks. The number of active players was reduced to six after early experiments with eleven players produced excessive track fragmentation under occlusion. The simulator was configured to export frames at 25 fps, matching the pipeline's expected input rate.

5.2 Data Generation

The primary training clip spans a full 90-minute simulated match comprising 148,280 frames. Raw simulator output occupies approximately 171 GB of disk space as per-frame PNG images with accompanying JSON annotations. This was compressed to approximately 10–20 GB using H.264 encoding at CRF 18, a near-lossless setting suitable for sports footage, and uploaded to the Snellius HPC cluster for pipeline processing. The full upstream pipeline was run on the simulated footage on the cluster, requiring 64 GB of host RAM and a 32-hour wall time.

5.3 Track Merging

The tracker produced 249 distinct track IDs for six real players due to re-identification failures. A graph colouring algorithm was developed to consolidate these fragments. The key insight is that two tracks active in the same frame cannot belong to the same player. This mutual exclusion constraint is encoded as a conflict graph over all track pairs, which is then coloured using a DSATUR-style greedy heuristic [1]. Tracks assigned the same colour are merged into a single continuous trajectory, with gaps filled by NaN placeholders.

5.4 Ground Truth Alignment

The simulator exports joints in a 15-joint world-space coordinate system that must be converted to the Human3.6M-17 pipeline space before it can serve as a training target. This is performed in three steps. First, each pipeline track is assigned to its corresponding ground-truth player via a least-squares 2D similarity transform followed by Hungarian assignment on RMS trajectory distance. Tracks whose best-match distance exceeds 8 metres are excluded as ghost detections. Second, ground-truth joints are remapped to Human3.6M-17 format and made root-relative by subtracting the hip midpoint. Third, a global 3D Procrustes alignment [12] is estimated via SVD over all matched frames and players, producing a single rotation R and scale s that maps ground-truth coordinates into the pipeline's 3D space for the full sequence.

5.5 Dataset Construction and Augmentation

Paired training samples are constructed by extracting sliding windows of $W = 243$ frames with a training stride of 121 frames and a validation stride of 243 frames. A stratified split holds out every fifth window block for validation, ensuring coverage across the full match rather than only its final segment. The final dataset comprises approximately 3,200 training windows and 800 validation

windows. Five augmentation operations are applied during training: random yaw rotation, temporal flipping, additive Gaussian noise ($\sigma = 0.05$ m), random frame dropout ($p = 0.20$), and uniform bone scaling ([0.85, 1.15×]). These parameters were increased from conservative initial values after a five-fold train/validation gap was observed in early training, indicating overfitting to the limited variety of a single match recording.

6 TRANSFORMER TRAINING

The Spatio-Temporal Pose Refinement Transformer was trained on the SURF Snellius HPC cluster using a single NVIDIA A100 40 GB SXM4 GPU. Training was implemented in PyTorch using the AdamW optimiser with a peak learning rate of 1×10^{-3}, a linear warmup over 5 epochs, and cosine annealing to zero over a maximum of 150 epochs. Gradients were clipped to a maximum norm of 1.0 and mixed precision training was applied using bfloat16 autocast. The effective batch size was 16, reduced from larger values to fit within the GPU memory budget imposed by the temporal attention tensor of shape $(B \cdot J, W, W)$.

Early stopping was applied with a patience of 25 epochs, using validation MPJPE as the monitored metric. The model converged rapidly in the first 30 epochs, improving from an initial MPJPE of 152.9 mm to a best validation MPJPE of 79.2 mm at epoch 41. After epoch 30 the model began to overfit, with training loss continuing to decrease while validation MPJPE plateaued and slowly increased. Training was terminated by early stopping before significant degradation occurred.

A significant train/validation gap of approximately 5× was observed in an initial training run. Root cause analysis identified two contributing factors: a training window stride of 50 frames producing 79% overlap between consecutive windows, causing the same frame to appear in up to five training windows and reducing the effective diversity of the training set; and insufficient regularisation for a model trained on a single 90-minute recording. The stride was increased to 121 frames (50% overlap) and dropout was raised from 0.1 to 0.2. These changes produced the best result of 79.2 mm reported above.

The training hyperparameters are summarised in Table 7.

7 EVALUATION

7.1 Experimental Setup

Evaluation is performed on 800 held-out validation windows extracted from the stratified split described in Section 5.5. Each window spans 243 frames (≈ 9.7 seconds at 25 fps). The primary metric is Mean Per Joint Position Error (MPJPE) in millimetres, computed over all valid ground-truth frames and joints, excluding the Pelvis (fixed at origin by construction), Nose, and Head (no ground-truth equivalent). Mean input completeness across validation windows is 97.0%, with ground-truth completeness of 85.3%.

7.2 Quantitative Results

Figure 5 shows the validation MPJPE over the full training run of 67 epochs across two combined Snellius jobs. The model converges from an initial MPJPE of 152.9 mm to a best of 79.2 mm at epoch 41, after which validation error begins to rise slowly while training loss continues to decrease, consistent with the overfitting behaviour discussed in Section 6. Early stopping terminates training at epoch 66. The best checkpoint at epoch 41 is used and applied for all subsequent evaluation.

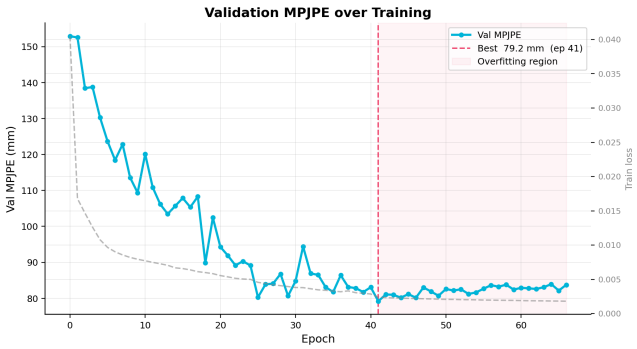

Figure 5: Validation MPJPE over 67 training epochs across two combined training runs. The best checkpoint of 79.2 mm is reached at epoch 41, marked by the dashed vertical line. After this point the model begins to overfit, with validation MPJPE rising slowly while training loss continues to decrease.

Table 1 reports the frame-level MPJPE for three methods evaluated on the validation set. The pipeline output and Gaussian temporal smoothing perform near-identically at 184.0 mm and 182.4 mm respectively, a difference of less than 1%. The Transformer achieves 80.4 mm, a reduction of 103.6 mm (56.3%), when run post hoc on all frames. The 79.2 mm reported during training was obtained on the segmented validation windows; full-sequence inference additionally covers sequence boundaries, where context is padded, and the more conservative 80.4 mm figure is used throughout. The near-zero improvement from Gaussian smoothing is a critical finding: it confirms that the Transformer's improvement arises from learned structural pose priors rather than from temporal filtering alone, directly supporting the architectural motivation presented in Section 4.9.

Table 1: Frame-level MPJPE on the validation set. Lower is better.

Method	Mean	Median	P90	<100mm
Pipeline (out post-Stage 4)	184.0 mm	172.8 mm	277.3 mm	6.7%
Gaussian smoothing	182.4 mm	—	—	7.0%
Transformer	**80.4 mm**	**77.6 mm**	**105.7 mm**	**85.4%**

Figure 6 shows the cumulative distribution of frame-level MPJPE across all three methods. The pipeline and Gaussian smoothing curves are nearly indistinguishable, while the Transformer curve is shifted substantially leftward, with 85.4% of frames falling below 100 mm compared to 6.7% for the pipeline.

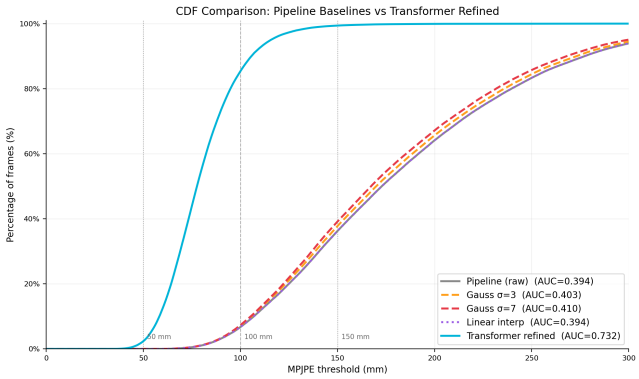

Figure 6: Cumulative distribution of frame-level MPJPE for the pipeline, Gaussian smoothing, and Transformer refinement. The near-identical pipeline and Gaussian curves confirm that temporal filtering alone does not address the structural failure modes of monocular lifting on broadcast footage.

7.3 Per-Joint Analysis

Figure 7 shows the per-joint MPJPE before and after Transformer refinement. A clear proximal-to-distal error gradient is present in both the pipeline and refined outputs: hips improve from 66 mm to 30 mm, knees from 168 mm to 79 mm, and ankles from 293 mm to 125 mm. The gradient is reduced but not eliminated by refinement, consistent with the fundamental difficulty of localising distal joints from a high-angle camera where feet are frequently occluded by the body or the pitch surface. Upper-body joints show similarly large improvements: shoulders improve from 231 mm to 81 mm and wrists from 252 mm to 109 mm. Figure 8 shows this gradient directly for a single frame: the correction applied at the hips and torso is small, while the hands and feet are displaced substantially further, and the refined skeleton is visibly more compact than the pipeline input.

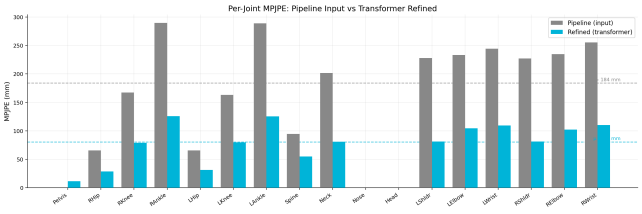

Figure 7: Per-joint MPJPE before and after Transformer refinement. The proximal-to-distal error gradient is smallest at the hips, largest at the ankles and is reduced but not eliminated by refinement, consistent with the occlusion patterns characteristic of football footage.

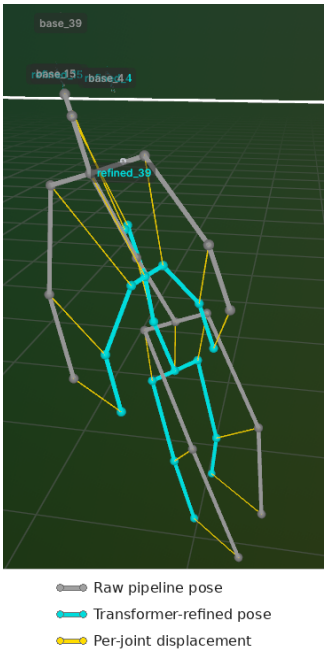

Figure 8: Qualitative comparison of a single player pose before and after Transformer refinement. The deterministic pipeline output after Stage 4 is shown in grey, the refined output in cyan, and yellow lines connect corresponding joints to indicate the magnitude and direction of the per-joint correction. The displacement is largest at the extremities and smallest near the torso, consistent with the proximal-to-distal error gradient reported. The refined skeleton is also smaller overall, reflecting the correction of the pipeline's tendency to over-scale players.

7.4 Facing Direction Error

Beyond joint position accuracy, the Transformer substantially improves body orientation estimation. The mean angular error in the facing direction which is computed from the shoulder axis projected onto the ground plane reduces from 52.3° to 21.5°, a reduction of 58.9%. This result demonstrates that the Transformer's temporal attention mechanism learns to infer body orientation from sequential context even when individual frames provide insufficient signal for reliable orientation estimation. Facing direction is a practically relevant metric for football analytics, as it can be used to determine whether a player is pressing, retreating, or tracking a run, that is not captured by a standard MPJPE evaluation.

7.5 Failure Cases

Analysis of the six worst-performing validation windows reveals two systematic failure modes. The first is re-identification switches mid-window: when the upstream tracker swaps player identities during an occlusion event, the Transformer receives a discontinuous pose sequence and produces a smoothed trajectory that matches neither player. This is an upstream tracking failure that the refinement stage cannot detect or correct from pose information alone.

The second failure mode is explosive athletic movement: short-duration high-velocity actions such as direction changes, jumps, and kicks produce sharp ground-truth pose spikes that are absent from the smoothed pipeline input. the velocity stabilization loss, while effective at suppressing jitter, smoothe out these legitimate fast movements, and the model trained on a single match has seen too few examples of such dynamics to learn to amplify rather than smooth them. This is reflected in the acceleration analysis: the Transformer output has a mean acceleration of 1952 mm/s² compared to the ground-truth value of 3823 mm/s², indicating over-smoothing of actual athletic dynamics.

Figure 9 shows a typical upper-body joint movement that highlights both the gains and the remaining limits of the refinement stage. The Transformer reduces the left elbow MPJPE from 446 mm to 169 mm by suppressing the large frame-level oscillations in the pipeline input, recovering a smooth trajectory that is physically consistent with natural arm movement. However a systematic positive height offset from the ground truth persists across the full 9.7-second window. This is not frame-level noise, the refined trajectory is smooth and temporally consistent but a remaining depth scale bias inherited from the monocular lifting stage. MotionBERT, trained on Human3.6M subjects at approximately 4 metres from the camera, systematically misjudges the depth of players at the 30–50 metre broadcast distances characteristic of our footage. The Transformer learns to partially correct this bias from the training data, but a single match simply does not provide enough data of the full range of player distances, movements and camera angles for the correction to be absolutely complete.

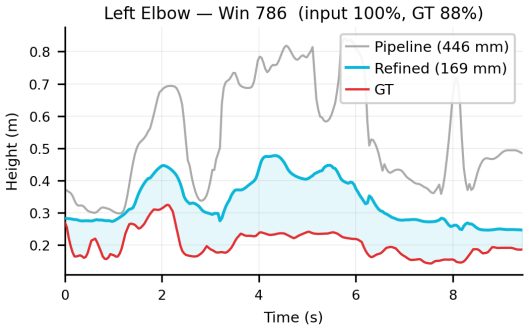

Figure 9: Root-relative height trajectory of the left elbow joint (validation window 786). The pipeline input (grey, 446 mm MPJPE) exhibits large frame-level oscillations, while the Transformer-refined output (cyan, 169 mm MPJPE) recovers a smooth trajectory. A residual depth scale bias persists throughout (red).

8 DISCUSSION

8.1 Interpreting the Main Result

The central finding of this thesis is a 56.3% reduction in MPJPE from an initial 184.0 mm to 80.4 mm, achieved by the Spatio-Temporal Pose Refinement Transformer operating as a post-hoc correction step on top of the standard lifting pipeline. This result should be understood as an answer to a specific research question: can a supervised spatio-temporal refinement stage meaningfully improve the output of a monocular lifting pipeline when deployed in a domain it was not designed for? The answer is yes, within the simulated domain in which the model was evaluated. Classical temporal filtering, being the simplest possible post-processing baseline, produces less than 1% improvement over the pipeline. The Transformer's 56.3% improvement therefore cannot be attributed to smoothing alone. It reflects the model learning structural pose priors from game-engine ground truth that MotionBERT, trained on slow laboratory motion, was never exposed to. This is the validation the architecture was designed to provide.

The facing direction result reinforces this interpretation. Reducing mean angular error from 52.3° to 21.5° demonstrates that the Transformer's temporal attention mechanism learns to infer body orientation from sequential context even when individual frames provide insufficient signal. This capability is directly relevant to football analytics where correctly determining whether a player is pressing, retreating, or tracking a run requires reliable body orientation estimation that single-frame monocular systems cannot provide. That the Transformer recovers this information from temporal context alone, without any camera geometry information, is a practically significant finding that standard MPJPE evaluation does not capture.

A question of practical relevance is whether 80 mm MPJPE is sufficient for the downstream analytics applications that motivate this work. The answer depends on the task. For coarse tactical analysis, player positioning, formation shape, pressing triggers, and field coverage, 80 mm average joint error is likely acceptable, as these analyses depend primarily on field position and body orientation rather than precise limb localisation. The facing direction improvement to 21.5° directly supports this use case. For fine-grained biomechanical analysis such as joint angle computation, injury risk assessment, and technique coaching, 80 mm is probably insufficient, particularly given the residual ankle errors of 125 mm and the systematic depth bias in upper-body joints visible in Figure 9. The 40 mm threshold commonly cited in the biomechanics literature [37] as the minimum for reliable joint angle computation has, however, not been reached. This does not take away from the contribution; it explains where the system currently stands on the accuracy-utility spectrum and clearly points to what should be done next as described in Section 8.4.

8.2 Contextualisation Against Related Work

A direct numerical comparison between this work and standard benchmarks is not meaningful. MotionBERT achieves 37.5 mm MPJPE on Human3.6M [17], a controlled laboratory dataset where subjects fill a large portion of the frame under professional lighting with no inter-subject occlusion. Our evaluation operates under fundamentally different conditions: broadcast footage where players span 50–200 pixels in height, with frequent occlusion, no sports-specific lifting model fine-tuning, and ground truth derived from a game engine rather than motion capture. Comparing these numbers directly would misrepresent both results.

The more informative comparison is against work that explicitly targets sports-domain deployment. Yeung et al. [37] demonstrate that MotionAGFormer trained on Human3.6M alone achieves 257 mm MPJPE on athletic movements, and that fine-tuning on their sports-specific AthletePose3D dataset reduces this to approximately 98 mm a reduction of around 62%. Our pipeline achieves 184 mm output, after deterministic processing, and 80.4 mm after Transformer refinement, without any sports-specific fine-tuning of the lifting model. That our post-hoc refinement approach reaches a similar level of accuracy to domain-specific lifting model fine-tuning is a meaningful finding: it suggests that a supervised correction stage can compensate for the distributional mismatch between laboratory training data and sports deployment without requiring the lifting model itself to be retrained. This makes the approach particularly practical, MotionBERT can be replaced with any future lifting model and the refinement stage retrained independently, realistically resulting in even better results.

The architectural motivation for this approach is further supported by BioPose [21], which demonstrates that a dedicated refinement stage operating on top of an initial mesh recovery model achieves state-of-the-art biomechanical accuracy on controlled footage. While BioPose targets a different regime, specifically a single-subject close-range video with full biomechanical skeleton constraints. This validates the broader principle that post-hoc learned refinement is a viable and competitive alternative to end-to-end retraining. This thesis applies that principle to the previously underexplored problem of multi-player broadcast football footage, where the deployment constraints of monocular wide-angle capture make end-to-end retraining impractical.

8.3 Limitations

Multiple limitations bound the current results and should be understood by anyone seeking to deploy or extend this system.

The most significant limitation concerns the evaluation split. Training and validation windows are drawn from a single simulated match, held out in blocks of every fifth window (Section 5.5). Because football motion is strongly correlated and consecutive training windows overlap by 50%, validation windows adjacent to a block boundary share frames, players, and phases of play with the training set. The reported error therefore reflects performance on motion that is temporally proximate to the training data rather than on genuinely unseen footage, and should be read as an upper bound on the model's accuracy. The train/validation gap observed in the initial training run and discussed in Section 7 is consistent with this: the regularisation changes reduced the gap, but the underlying correlation between the two sets was not removed. A clean evaluation requires holding out either separate matches, (for which we lacked the data), or contiguous segments of several minutes, so that no validation window shares temporal context with training data.

This leads directly to another significant limitation of **single-match training data**. The Transformer is trained and evaluated on the same simulated recording, which means the validation results reflect in-distribution performance on a known match rather than generalisation to unseen footage. The explosive movement failure mode documented in Section 7 is a direct consequence of this: one 90-minute match does not contain sufficient variety of high-acceleration athletic data for the model to learn to amplify rather than smooth them. The over-smoothing observed in the acceleration analysis, from 1952 mm/s² against a ground-truth of 3823 mm/s² is not a fundamental architectural problem but a data coverage problem that would be addressed by training on multiple matches.

A further limiting factor is the **ground-truth fidelity gap**. While the game engine exports skeletal data derived from a real professional football match, the ground-truth joint positions are interpreted by the engine's internal skeletal representation rather than by direct motion capture of real players, meaning the engine's skeletal model approximates human biomechanics and may not faithfully reproduce precise joint centre locations for contact situations and extreme athletic configurations. Beyond the skeletal representation, the raw data carried practical quality issues, also created due to the nature of the game engine footage, such as reusing the same avatars making it inherently difficult for the re id layer to accurately differentiate the players from one another. Furthermore there is a systematic depth bias visible in Figure 9, where the refined output runs consistently above the ground truth despite temporal smoothness, which is likely a partial consequence of this: MotionBERT's depth underestimation at 30–50 metre distances is partially but not fully corrected by a model whose training signal carries residual alignment uncertainty, due to errors that were introduced when this 3D footage was captured. Validating against direct motion-capture ground truth would establish the practical impact of this fidelity gap. The synthesised Pelvis and Spine joints (Section 4.6) are a further source of systematic error. Because the Spine is the parent of the Neck in the Human3.6M skeleton, a misplacement there propagates into the shoulder and arm estimates and may contribute, alongside the depth bias discussed above, to the upper-body errors reported in section 7.3. The simulator exports true pelvis and spine positions, so the size of this approximation error could be measured directly against the validation set.

Lastly there is **residual identity ambiguity**. While the fragment linking and per-frame ground-truth assignment steps described in Section 5 and 4.3 resolve the majority of re-identification failures before data reaches the Transformer, identity switches that occur within a single 243-frame window and involve spatially proximate players, two defenders running side by side, for instance may not be caught by the distance threshold. In these cases the Transformer receives a pose sequence that contains a discontinuity it cannot detect from pose information alone, and the resulting smoothed output corresponds to neither player's true trajectory. This is an edge case rather than a systematic failure, but it represents a ceiling on performance in crowded scenes that the refinement stage cannot address without access to upstream identity confidence scores.

8.4 Future Work

Multiple directions follow directly from the limitations above.

Training on multiple simulated and real matches is the highest-priority extension. Diverse athletic scenarios, explosive direction changes, jumping, and contact situations, are systematically under-represented in a single match recording, and exposing the model to this variety would directly address the over-smoothing of fast

movements. Retraining under a leakage-free split is a prerequisite for interpreting the reported accuracy. Holding out contiguous multi-minute segments, or entire separate matches, would remove the temporal correlation between training and validation data and establish how much of the current result reflects generalisation rather than proximity to the training set.

Incorporating the AthletePose3D dataset [37] as a source of sports-specific ground truth would provide the kinematic diversity that simulated football footage alone cannot supply.

Another broader opportunity exists in the area of ground-truth data generation. The primary bottleneck for supervised refinement approaches is not model architecture but training data diversity, as mentioned above. Two directions are particularly promising: modifying existing realistic football game engines such as FIFA 16, which offers substantially more variance in athletes movements and representation than the engine used in this thesis, and Google Research Football [22], an open-source simulation environment that provides programmatic control over match scenarios, player behaviour, and camera viewpoints. Both were considered during the development of this project but set aside due to time constraints. Either would provide the athletic diversity across player speeds, body types, and occlusion patterns that a single match recording cannot supply, and would directly address the generalisation limitations identified in this thesis.

Finally, replacing the simulated ground truth with real motion-capture annotations or evaluating on a public sports benchmark such as AthletePose3D for direct comparability, would establish whether the simulation-to-real gap affects deployment performance and provide a basis for comparing future refinement approaches against the results reported here.

9 CONCLUSION

This thesis set out to answer two questions: whether a robust 3D pose estimation pipeline can be built from scratch for broadcast football footage within realistic hardware constraints, and whether a supervised spatio-temporal refinement stage can meaningfully correct the systematic errors that monocular lifting produces in this domain. The answer to both is yes.

The pipeline developed here decomposes the standard two-stage lifting architecture into five explicitly separated modules, these being: segmentation, tracking, keypoint estimation, pre-lifting refinement, and post-lifting refinement, each with a clearly defined responsibility. The adoption of SAM 3 as the segmentation frontend directly addresses the track loss and confidence instability that disqualify YOLO-based detection for broadcast footage. The pre- and post-lifting refinement stages ensure that both the input to and the output of MotionBERT are as anatomically consistent and temporally stable as deterministic processing allows.

The primary contribution, the Spatio-Temporal Pose Refinement Transformer, demonstrates that the residual errors remaining after this deterministic processing are not irreducible noise but structured, learnable patterns. Trained on game-engine ground truth and evaluated on held-out validation windows, the model achieves 80.4 mm MPJPE at its best checkpoint marking a 56.3% reduction from the 184.0 mm baseline of the pipeline. Critically, Gaussian temporal smoothing produces less than 1% improvement over the pipeline, confirming that the Transformer's gains arise from learned structural pose priors rather than filtering alone. The facing direction error reduction from 52.3 to 21.5 further demonstrates that the model recovers body orientation from temporal context in a way that no single-frame method can.

These results have a clear practical interpretation. At 80 mm average joint error, the system is well suited for coarse tactical analysis including: player positioning, pressing triggers, formation shape, and body orientation. However in its current state the system falls short of the 40 mm threshold required for fine-grained biomechanical analysis. The primary bottleneck is not architectural but data-driven: a single simulated match cannot supply the athletic diversity needed to correct explosive movements and extreme joint configurations. Training on multiple matches, incorporating the AthletePose3D dataset [37], or leveraging more realistic game engines such as FIFA 16 or Google Research Football [22] are the most direct paths to closing this gap.

More broadly, this work validates the principle that a modular, supervised post-hoc refinement stage is a practical and effective alternative to end-to-end retraining of the lifting backbone. Because the Transformer operates on the output of any upstream lifter, it can be retrained independently as better lifting models emerge, a property that makes the approach particularly suitable for deployment in a rapidly evolving field. The pipeline, the training procedure, and the simulation-based ground truth generation methodology together constitute a reproducible foundation for future work in automated biomechanical analysis, tactical performance monitoring, and real-time sports broadcasting.

REFERENCES

[1] Daniel Brelaz. 1979. New Methods to Color the Vertices of a Graph. *Commun. ACM* 22, 4 (1979), 251–256.

[2] S. Butterworth. 1930. On the Theory of Filter Amplifiers. *Experimental Wireless & the Wireless Engineer* 7 (Oct. 1930), 536–541.

[3] Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, Jie Lei, Tengyu Ma, Baishan Guo, Arpit Kalla, Markus Marks, Joseph Greer, Meng Wang, Peize Sun, Roman Rädle, Triantafyllos Afouras, Effrosyni Mavroudi, Katherine Xu, Tsung-Han Wu, Yu Zhou, Liliane Momeni, Rishi Hazra, Shuangrui Ding, Sagar Vaze, Francois Porcher, Feng Li, Siyuan Li, Aishwarya Kamath, Ho Kei Cheng, Piotr Dollár, Nikhila Ravi, Kate Saenko, Pengchuan Zhang, and Christoph Feichtenhofer. 2025. SAM 3: Segment Anything with Concepts. arXiv:2511.16719 [cs.CV] https://arxiv.org/abs/2511.16719

[4] Géry Casiez, Nicolas Roussel, and Daniel Vogel. 2012. 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* (Austin, Texas, USA) *(CHI '12)*. Association for Computing Machinery, New York, NY, USA, 2527–2530. https://doi.org/10.1145/2207676.2208639

[5] Ching-Hsuan Chen et al. 2021. Biomechanical Characteristics for Identifying the Cutting Direction of Professional Soccer Players. *Applied Sciences* 11, 16 (2021), 7193.

[6] Jeongjun Choi, Dongseok Shim, and H. Jin Kim. 2023. DiffuPose: Monocular 3D Human Pose Estimation via Denoising Diffusion Probabilistic Model. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE.

[7] MMPose Contributors. 2020. OpenMMLab Pose Estimation Toolbox and Benchmark. https://github.com/open-mmlab/mmpose.

[8] Carl de Boor. 1978. *A Practical Guide to Splines*. Springer-Verlag, New York. https://api.semanticscholar.org/CorpusID:122101452

[9] Moritz Einfalt, Katja Ludwig, and Rainer Lienhart. 2023. Uplift and Upsample: Efficient 3D Human Pose Estimation with Uplifting Transformers. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2902–2911.

[10] Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. 2010. The Pascal Visual Object Classes (VOC) Challenge. *International Journal of Computer Vision* 88, 2 (2010), 303–338.

[11] FIFA Skeletal Light Tracking Challenge. 2026. FIFA Skeletal Tracking Starter Kit 2026. https://github.com/FIFA-Skeletal-Light-Tracking-Challenge/FIFA-Skeletal-Tracking-Starter-Kit-2026/tree/main.

[12] John C. Gower. 1975. Generalized Procrustes Analysis. *Psychometrika* 40, 1 (1975), 33–51.

[13] Richard Hartley and Andrew Zisserman. 2004. *Multiple View Geometry in Computer Vision* (2 ed.). Cambridge University Press.

[14] Dan Hendrycks and Kevin Gimpel. 2016. Bridging Nonlinearities and Stochastic Regularizers with Gaussian Error Linear Units. *CoRR* abs/1606.08415 (2016). arXiv:1606.08415 http://arxiv.org/abs/1606.08415

[15] Mingyang Hu et al. 2023. A Geometric Knowledge Oriented Single-Frame 2D-to-3D Human Absolute Pose Estimation Method. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*. IEEE.

[16] Wei Hu et al. 2021. Conditional Directed Graph Convolution for 3D Human Pose Estimation. In *Proceedings of the ACM International Conference on Multimedia (MM)*. ACM.

[17] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. 2014. Human3.6M: Large Scale Datasets and Predictive Methods for 3D Human Sensing in Natural Environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36, 7 (2014), 1325–1339.

[18] Tao Jiang, Xinchen Xie, and Yining Li. 2024. RTMW: Real-Time Multi-Person 2D and 3D Whole-body Pose Estimation. arXiv:2407.08634 [cs.CV] https://arxiv.org/abs/2407.08634

[19] Adalbert Ibrahim Kapandji. 2007. *The Physiology of the Joints: Upper Limb* (6th ed.). Vol. 1. Churchill Livingstone.

[20] Adalbert Ibrahim Kapandji. 2010. *The Physiology of the Joints: Lower Limb* (6th ed.). Vol. 2. Churchill Livingstone.

[21] Farnoosh Koleini et al. 2025. BioPose: Biomechanically-Accurate 3D Pose Estimation from Monocular Videos. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. IEEE.

[22] Karol Kurach, Anton Raichuk, Piotr Stańczyk, Michał Zając, Olivier Bachem, Lasse Espeholt, Carlos Riquelme, Damien Vincent, Marcin Michalski, Olivier Bousquet, and Sylvain Gelly. 2020. Google Research Football: A Novel Reinforcement Learning Environment. arXiv:1907.11180 [cs.LG] https://arxiv.org/abs/1907.11180

[23] Wenhao Li et al. 2022. MHFormer: Multi-Hypothesis Transformer for 3D Human Pose Estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.

[24] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. 2015. Microsoft COCO: Common Objects in Context. arXiv:1405.0312 [cs.CV] https://arxiv.org/abs/1405.0312

[25] Debapriya Maji, Soyeb Nagori, Manu Mathew, and Deepak Poddar. 2022. YOLO-Pose: Enhancing YOLO for Multi Person Pose Estimation Using Object Keypoint Similarity Loss. arXiv:2204.06806 [cs.CV] https://arxiv.org/abs/2204.06806

[26] Soroush Mehraban et al. 2024. MotionAGFormer: Enhancing 3D Human Pose Estimation with a Transformer-GCNFormer Network. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. IEEE.

[27] Francesc Moreno-Noguer. 2017. 3D human pose estimation from a single image via distance matrix regression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.

[28] Ranjan Sapkota, Rahul Harsha Cheppally, Ajay Sharda, and Manoj Karkee. 2026. YOLO26: Key Architectural Enhancements and Performance Benchmarking for Real-Time Object Detection. arXiv:2509.25164 [cs.CV] https://arxiv.org/abs/2509.25164

[29] Abraham. Savitzky and M. J. E. Golay. 1964. Smoothing and Differentiation of Data by Simplified Least Squares Procedures. *Analytical Chemistry* 36, 8 (1964), 1627–1639. https://doi.org/10.1021/ac60214a047 arXiv:https://doi.org/10.1021/ac60214a047

[30] Xiaohan Shen et al. 2023. Global-to-Local Modeling for Video-Based 3D Human Pose and Shape Estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.

[31] Vineet Srivastav et al. 2024. A survey on deep 3D human pose estimation. *Artificial Intelligence Review* (2024).

[32] Denis Tome, Chris Russell, and Lourdes Agapito. 2017. Lifting from the deep: Convolutional 3D pose estimation from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.

[33] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need. *CoRR* abs/1706.03762 (2017). arXiv:1706.03762 http://arxiv.org/abs/1706.03762

[34] Mingkun Wei et al. 2025. Learning Pyramid-Structured Long-Range Dependencies for 3D Human Pose Estimation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. IEEE.

[35] Xiuwen Xi et al. 2024. Enhancing human pose estimation in sports training: Integrating spatiotemporal transformer for improved accuracy and real-time performance. *Expert Systems with Applications* (2024).

[36] Yufei Xu, Jing Zhang, Qiming Zhang, and Dacheng Tao. 2024. ViTPose++: Vision Transformer for Generic Body Pose Estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 46, 2 (2024), 1212–1230. https://doi.org/10.1109/TPAMI.2023.3330016

[37] N. Yeung et al. 2025. AthletePose3D: A Benchmark Dataset for 3D Human Pose Estimation and Feedback Generation for Sports. In *Proceedings of the IEEE/CVF CVPR Workshops (CVSports)*.

[38] Bin-Bin X. Yu et al. 2023. GLA-GCN: Global-local Adaptive Graph Convolutional Network for 3D Human Pose Estimation from Monocular Video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE.

[39] Shengjie Yuan et al. 2025. GTA-Net: An IoT-integrated 3D human pose estimation system for real-time adolescent sports posture correction. *Internet of Things* (2025).

[40] Jinlu Zhang et al. 2022. MixSTE: Seq2seq Mixed Spatio-Temporal Encoder for 3D Human Pose Estimation in Video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.

[41] Jingbo Zhang et al. 2022. Uncertainty-Aware 3D Human Pose Estimation from Monocular Video. In *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer.

[42] Yifu Zhang, Peize Sun, Yi Jiang, Dongdong Yu, Fucheng Weng, Zehuan Yuan, Ping Luo, Wenyu Liu, and Xinggang Wang. 2022. ByteTrack: Multi-Object Tracking by Associating Every Detection Box. arXiv:2110.06864 [cs.CV] https://arxiv.org/abs/2110.06864

[43] Ce Zheng, Matias Mendieta, Taojiannan Yang, Guo-Jun Qi, and Chen Chen. 2023. FeatER: An Efficient Network for Human Reconstruction via Feature Map-Based TransformER. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.

[44] Ce Zheng, Sijie Zhu, Matias Mendieta, Taojiannan Yang, Chen Chen, and Zhengming Ding. 2021. 3D Human Pose Estimation with Spatial and Temporal Transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE.

[45] Kaiyang Zhou, Yongxin Yang, Andrea Cavallaro, and Tao Xiang. 2019. Omni-Scale Feature Learning for Person Re-Identification. arXiv:1905.00953 [cs.CV] https://arxiv.org/abs/1905.00953

[46] Wentao Zhu, Xiaoxuan Ma, Zhaoyang Liu, Libin Liu, Wayne Wu, and Yizhou Wang. 2023. MotionBERT: A Unified Perspective on Learning Human Motion Representations. arXiv:2210.06551 [cs.CV] https://arxiv.org/abs/2210.06551

A TABLES

Table 2: MPJPE (mm) on Human3.6M using detected 2D poses. Lower is better. Results reported from original papers.

Method	Year	MPJPE (mm)
PoseFormer [44]	2021	44.3
MHFormer [23]	2022	43.0
MixSTE [40]	2022	39.8
MotionBERT [46]	2023	37.5

Table 3: Complete COCO-17 to Human3.6M-17 joint remapping. Joints marked as synthetic are computed by averaging the listed COCO-17 joints.

H36M idx	Joint	Derived from (COCO-17)
0	Pelvis	Mean of L-Hip (11) and R-Hip (12) *(synthetic)*
1	R-Hip	R-Hip (12)
2	R-Knee	R-Knee (14)
3	R-Ankle	R-Ankle (16)
4	L-Hip	L-Hip (11)
5	L-Knee	L-Knee (13)
6	L-Ankle	L-Ankle (15)
7	Spine	Mean of Pelvis and Neck *(synthetic)*
8	Neck	Mean of L-Shoulder (5) and R-Shoulder (6)
9	Nose	Nose (0)
10	Head	Mean of L-Eye (1), R-Eye (2), L-Ear (3), R-Ear (4)
11	L-Shoulder	L-Shoulder (5)
12	L-Elbow	L-Elbow (7)
13	L-Wrist	L-Wrist (9)
14	R-Shoulder	R-Shoulder (6)
15	R-Elbow	R-Elbow (8)
16	R-Wrist	R-Wrist (10)

Table 4: 1-Euro filter parameters for online 2D keypoint smoothing.

Parameter	Value	Effect
`min_cutoff`	2.5 Hz	Base cut-off — higher values apply more smoothing at rest
`beta`	0.007	Speed coefficient — higher values increase cut-off during fast motion
`d_cutoff`	1.0 Hz	Cut-off for the derivative estimate

Table 5: Butterworth low-pass filter parameters for 3D joint coordinate smoothing.

Parameter	Value	Description
`smooth_3d`	True	Enable or disable smoothing
`smooth_3d_cutoff_hz`	12.0 Hz	Low-pass cut-off frequency
`smooth_3d_order`	2	Filter order

Table 6: Biomechanical joint angle limits enforced in the post-lifting refinement stage, based on established anatomical range of motion values [19, 20].

Joint	Range	Description
R-Knee	10–175	Knee flexion
L-Knee	10–175	Knee flexion
R-Elbow	5–175	Elbow flexion
L-Elbow	5–175	Elbow flexion
R-Hip	5–170	Hip flexion (pelvis–hip–knee)
L-Hip	5–170	Hip flexion (pelvis–hip–knee)
R-Shoulder	30–180	Shoulder abduction (spine–neck–shoulder)
L-Shoulder	30–180	Shoulder abduction (spine–neck–shoulder)
Spine	120–180	Lateral spinal angle — kept mostly straight
Neck	100–180	Forward neck bend

Table 7: Transformer training hyperparameters.

Parameter	Value
Optimiser	AdamW
Peak LR	1×10^{-3}
LR schedule	Linear warmup (5 epochs) + cosine annealing
Weight decay	1×10^{-4}
Gradient clip	1.0
Batch size	16
Max epochs	150
Early stopping	Patience 25 (val MPJPE)
Mixed precision	bfloat16
Best val MPJPE	79.2 mm (epoch 41)

B SPATIOTEMPORALBLOCK ARCHITECTURE

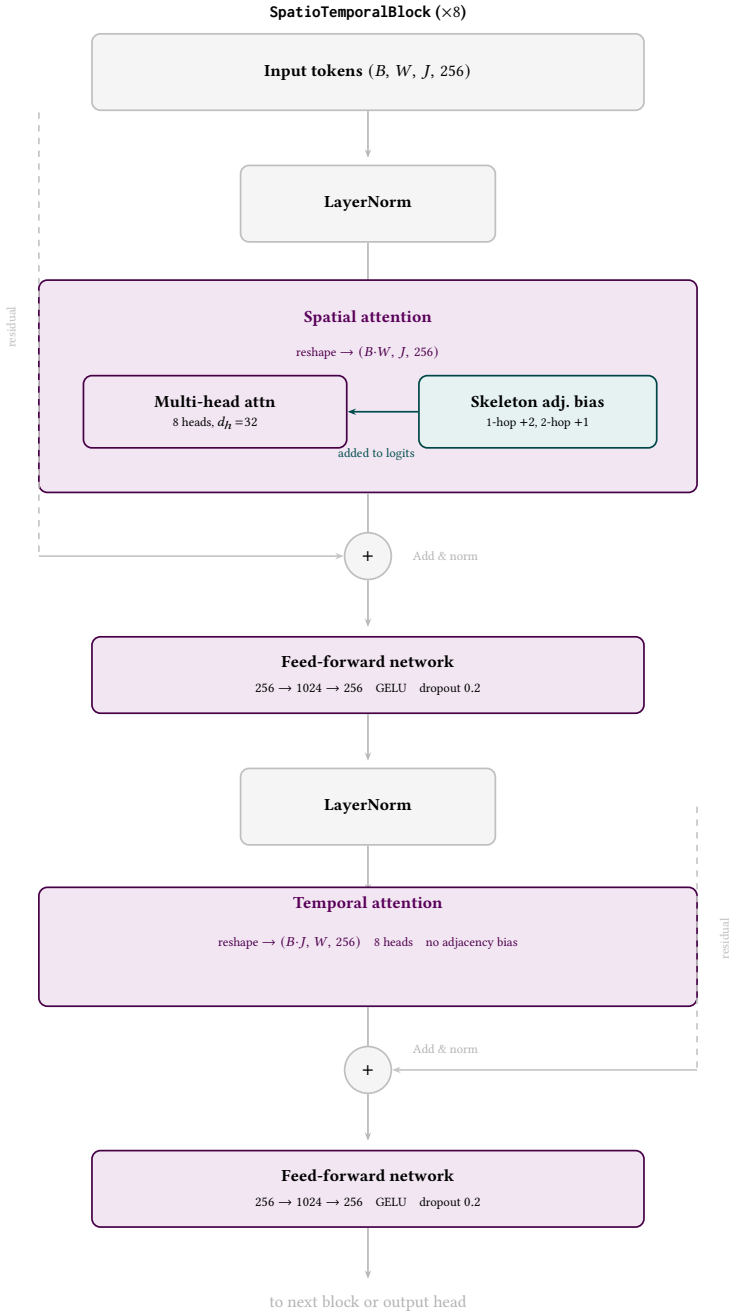

Figure 10: A single `SpatioTemporalBlock` (×8). Spatial attention operates across all $J = 17$ joints within each frame, with a learnable skeleton adjacency bias matrix added to the attention logits as a soft anatomical prior. Temporal attention operates across all $W = 243$ frames independently per joint. Both sub-layers use pre-norm residual connections followed by a feed-forward network with GELU activation and dropout.