Research Journal: Video-to-Biomechanics-to-Performance Inference in Rowing

Purpose of this document

This document records (i) project decisions and rationale, (ii) progress milestones, (iii) experimental results, and (iv) revisions to the technical plan as the research evolves.

Project objective (scientific statement)

The objective of this research project is to infer rowing biomechanics and downstream performance-relevant quantities directly from standard (unconstrained) video footage, without requiring instrumented boats or wearable sensors at inference time.

Formally, given single- or multi-view video sequences $V(t)$ of a rowing athlete, the project aims to estimate a temporally consistent 3D human kinematic state $\hat{S}(t)$ and to derive interpretable biomechanical features $\hat{F}(t)$ (e.g., joint angles and segment kinematics) that can be mapped to performance targets $\hat{Y}(t)$ (e.g., force curve, power) via sequence models.

Core pipeline (high-level)

1. **Video input** (single- or multi-view; non-lab conditions)
2. **Preprocessing** (stabilization, normalization, view/time alignment)
3. **2D pose estimation** to obtain keypoints over time
4. **2D-to-3D lifting** to recover a temporally consistent 3D skeleton
5. **Kinematic and stroke-structure analysis** (biomechanics-first features)
6. **Sequence-based inference** from motion features to performance quantities
7. **Evaluation** against ground truth (numerical error) and biomechanical plausibility (structural validity)

Key constraints and principles (design requirements)

- **Unconstrained input:** the primary input is standard video (single camera permitted), potentially with variable lighting, background clutter, camera motion, compression artifacts, and occlusion.
- **No instrumentation at inference:** force sensors, ergometer telemetry, or boat telemetry are *not* assumed at inference time; such signals may be used only for supervised training and validation.

- **Interpretability-first:** intermediate representations should be physically interpretable (biomechanics-derived features) rather than purely end-to-end black-box predictions.
- **Generalization:** models should generalize across athletes, sessions, and filming conditions.
- **Dual evaluation:** accuracy is assessed both numerically (error vs. ground-truth analytics) and structurally (kinematics should obey rowing biomechanics and realistic timing).

Detailed research steps discussed to date (current plan)

Step 0: Feasibility and scoping (completed)

- Surveyed prior work and practical constraints to determine whether performance-relevant rowing metrics can plausibly be inferred from video-derived kinematics.
- Outcome: concluded that the project is feasible in principle, motivating the design of a modular, reproducible pipeline.

Step 1: Pipeline decomposition and specification (completed)

- Identified the major modules required for end-to-end inference: pose extraction, temporal/kinematic feature computation, stroke segmentation and alignment, and learning-based mapping to performance targets.
- Defined the guiding principle that intermediate outputs should be interpretable biomechanical quantities (e.g., joint angles and timing) suitable for scientific analysis.

Step 2: Data extraction; Video to pose time series (in progress)

Goal: obtain reliable per-frame human pose information for individual athletes from raw video.

1. **Data ingestion and organization:** collect and index video clips by athlete/session/camera; record metadata (frame rate, resolution, camera viewpoint).
2. **Preprocessing:** stabilize video if necessary; normalize scale/rotation when possible; optionally crop to the athlete/boat region to improve pose quality.
3. **2D pose estimation:** extract 2D keypoints $\hat{K}_{2D}(t)$ for relevant body landmarks (e.g., shoulders, elbows, wrists, hips, knees, ankles).
4. **Quality control:** detect and handle missing/low-confidence keypoints; apply temporal smoothing and outlier rejection as needed.

Step 3: Temporal biomechanics feature extraction (in progress)

Goal: transform pose time series into stroke-relevant kinematic features suitable for analysis and learning.

1. **Temporal alignment and stroke segmentation:** identify stroke cycles (catch, drive, finish, recovery) using kinematic cues; align cycles to a common phase variable.
2. **Kinematic reconstruction:**
   - Compute joint angles (e.g., hip angle, elbow angle) as functions of time/phase.
   - Compute segment velocities and accelerations where meaningful.
   - Define task-specific proxies such as estimated handle trajectory/velocity/acceleration (as permitted by the available keypoints and camera viewpoint).
3. **Coordination and timing features:** quantify drive-to-recovery ratio, catch timing consistency, inter-joint coordination patterns, and stroke-to-stroke variability.
4. **Canonical representation:** create a standardized feature vector sequence $\hat{F}(t)$ per stroke (or per unit phase) to support downstream modeling.

Progress update (January 21- February 4, 2026): data extraction experiments and findings

This section records what has been implemented and learned so far in the video-to-pose portion of the pipeline.

Pose extraction models evaluated. For 2D pose extraction, I focused on two strong, widely used options:

- **OpenMMLab MMPose** (pose estimation toolbox; used as the main integration framework).
- **RTMPose** models within the MMPose ecosystem (real-time multi-person pose estimation).

I also surveyed **Ultralytics YOLOv8-Pose** models as an alternative family of modern pose extractors.

Initial extraction: 2D keypoints from rowing video. My first attempt used straightforward per-frame 2D keypoint extraction on rowing video. Using MMPose, the results were reasonably precise in clear single-person views, but performance degraded under occlusion, and temporal smoothness was sometimes missing (visible frame-to-frame jitter). I experimented with temporal smoothing as a post-processing step; this helped somewhat but did not fully resolve the issue.

Figure 1: Single-athlete 2D overlay: reasonably precise keypoints, but imperfect temporal smoothness and sensitivity to occlusion in harder frames.

Camera motion and stabilization. A major issue in real footage was camera shake and inconsistent framing (the athlete was not consistently centered). I addressed this by stabilizing the video via anchoring/tracking a rigid visual reference on the boat (the rigger). After stabilization, I cropped the video so that the athlete remained approximately centered.

Multi-athlete identity switching and tracking attempt. Stabilization and cropping alone did not prevent occasional switches where the pose estimator would "jump" between athletes (especially in larger boats). To mitigate this, I attempted explicit single-person tracking using **DeepSORT** so that only one athlete identity would be followed over time. In practice, this approach was ineffective in my setting (frequent failures / unstable identity assignment), so I deprioritized it and moved on.

Qualitative examples (single-person success vs. multi-person failure). Figure 8 illustrates a representative case where a single athlete is clearly visible; in this regime, pose extraction worked well and individual strokes could be identified reliably from the resulting kinematic traces.

Figure 2 illustrates a failure mode: the camera was oriented obliquely (roughly along the boat, rather than approximately perpendicular), causing multiple athletes to overlap in image space and increasing the probability of identity mixing. In this setting, strict ID tracking did not work reliably, and the resulting extracted pose/kinematic data were poor even for very short clips (on the order of two strokes). Because this branch of the pipeline also incurred a large runtime cost (several minutes per short clip in my initial implementation), I discontinued this approach.

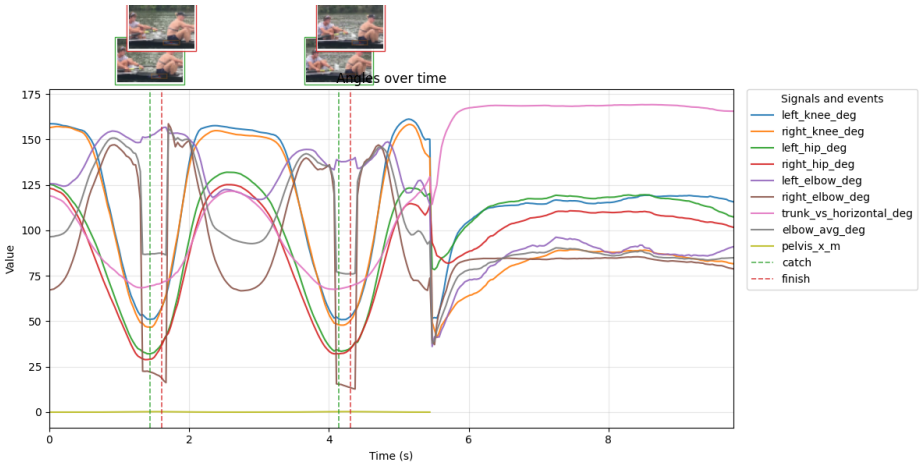

Figure 2: Failure case ("bad"): oblique camera angle and multi-athlete overlap led to identity mixing and poor pose/kinematic outputs; DeepSORT-based strict tracking was ineffective in this setting.

2D-to-3D lifting. Because the pipeline still produced only 2D poses, I integrated **MotionBERT** to lift the 2D pose sequences to a temporally consistent 3D pose. Spatially, the 3D reconstruction appeared very accurate; the primary limitation was that errors and jitter in the upstream 2D pose could propagate into the lifted 3D.

Figure 3: Single-athlete 3D overlay after 2D-to-3D lifting: strong spatial accuracy, with failure cases driven primarily by upstream 2D pose errors.

Runtime bottleneck. Although the overall pipeline (stabilization → 2D pose → 3D lifting) produced good qualitative results, it was extremely slow end-to-end in my initial implementation (e.g., approximately five minutes for processing a clip containing only a couple of strokes). This is a key practical limitation to address going forward (e.g., model selection, batching, acceleration, and/or reducing redundant computation).

Journal note (February 4, 2026): aligning video kinematics with telemetry domains. In on-water telemetry, a common representation is force plotted against the gate/oar angle, whereas

in video we naturally observe motion as a function of time. A core challenge for this project is therefore domain alignment: either (i) reparameterize video-derived kinematics from time t into a gate-angle (or other stroke-phase) domain, or (ii) convert telemetry from gate-angle into time. Both directions are nontrivial; my current intuition is to focus on re-indexing video frames into a gate-angle (or stroke-phase) domain so that small increments of gate-angle can be matched to corresponding parts of the force curve. Of course, perspective effects (viewpoint, foreshortening, and occlusion) make this difficult in real boat footage.

As a first "proof ground" for this idea, I am considering ergometer video, especially RP3 (a dynamic ergometer) because it provides high-quality force curves and naturally reports force versus distance (stroke length). Distance is plausibly easier to extract from video than gate-angle in the boat setting: by tracking the handle and identifying its start/finish positions, I can segment a stroke by handle displacement rather than by time, and then attempt to align force-versus-distance with video-derived motion. This is still challenging, but it seems like a more tractable first step toward demonstrating viability and research value. Practically, this also fits current constraints: the Charles is frozen right now, so collecting new on-water video is not straightforward.

Pivot to an integrated sports-focused pipeline (Sports2D + MotionBERT). After implementing stabilization, tracking, and related preprocessing myself, I concluded that continued iteration on a bespoke pipeline was not the best use of time: it requires extensive tuning and evaluation, and the slow runtime makes experimental iteration costly.

I therefore evaluated open-source pipelines designed specifically for sports/biomechanics use. Many options either required multi-view camera setups (lab-grade markerless motion capture) or were not optimized for sports footage. I selected **Sports2D** as the most practical alternative: it provides an integrated, configurable pipeline for 2D pose extraction (and associated filtering/tracking utilities) and outputs dense keypoint trajectories that are suitable for downstream biomechanics.

I implemented a new pipeline in which Sports2D provides the 2D keypoints, and I then perform 3D lifting via MotionBERT. A priority in this integration was to preserve as much of Sports2D's pose point data as possible (rather than reducing to a minimal subset), in order to support more fine-grained kinematic feature engineering downstream.

Sports2D-first qualitative result. In an initial Sports2D-first test, the apparent pose accuracy was high and the resulting keypoint trajectories were notably smooth. Importantly, this held even when running inference with relatively large frame gaps (i.e., not estimating pose on every single frame), and overall processing speed was relatively fast compared to my earlier bespoke pipeline.

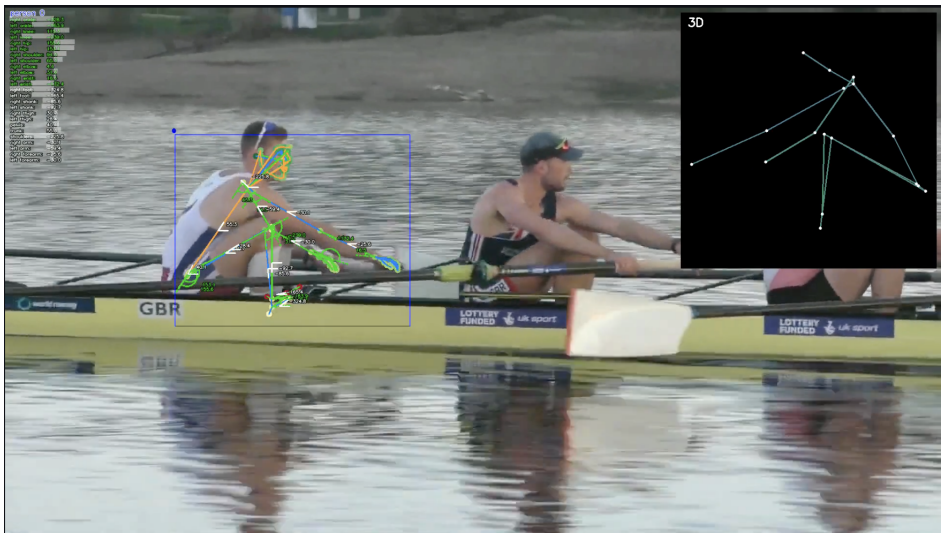

Figure 4: Sports2D-first test: high apparent accuracy and smooth keypoint trajectories, with relatively fast processing even under sparser (frame-gap) inference. Good individual person identification and selection. Lots of data to work with.

Additional Sports2D qualitative examples (erg vs. boat/tank). As shown by the following images, both erg and boat movements look much better tracked using Sports2D, even under some occlusion.

Figure 5: Boat/tank example with Sports2D: improved tracking quality and stability under partial occlusion.

Figure 6: Erg example with Sports2D: improved tracking quality and stability under partial occlusion.

Erg side-view under heavy occlusion (Sports2D + filtering). Even in cases of high self-occlusion (e.g., limbs occluding other body parts), Sports2D remained highly effective and smooth in my tests. This robustness was aided by post-processing: rejecting outliers with a Hampel filter and then applying a 4th-order Butterworth low-pass filter (6 Hz cutoff) to the keypoint trajectories.

Figure 7: Erg side-view with Sports2D: strong tracking under heavy self-occlusion, with Hampel outlier rejection and 4th-order 6 Hz Butterworth filtering for additional smoothness.

Angle features and stroke event identification. With the extracted poses, I computed joint angles of interest (notably elbow, hip, and knee angles). I then plotted these time series (via Matplotlib) to infer stroke events, focusing on detecting the *catch* and *finish* positions as prerequisites for determining the drive phase.

Empirically, once I identified the relevant peaks/troughs in the angle traces, event detection became

reliable:

- **Catch:** occurs at maximal knee compression (knee angle minimum).
- **Finish:** occurs at maximal elbow compression (elbow angle minimum).

This heuristic was very effective on the videos tested, and it provides a concrete path toward robust stroke segmentation using kinematics-first signals.

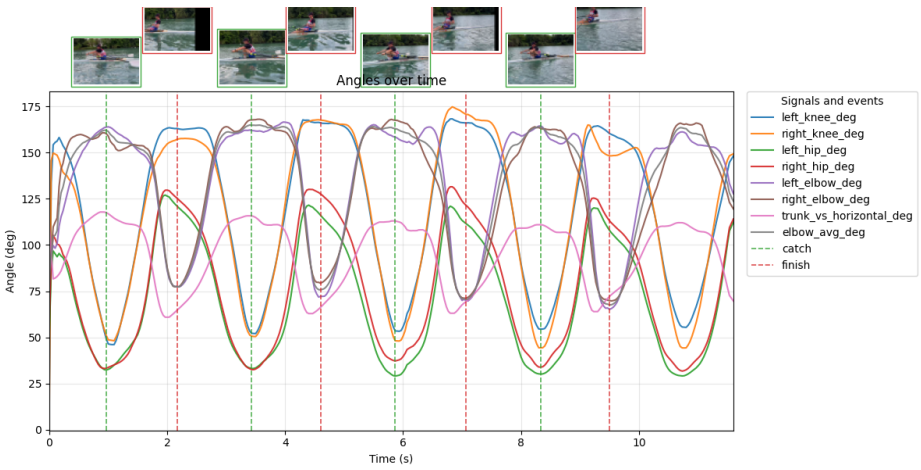

Figure 8: Single-person case ("with elbows"): pose extraction behaved well, and individual strokes were clearly identifiable.

Sports2D graph evidence (erg test). The following plot clearly suggested that I was on the correct path: the inferred kinematic traces were extremely smooth and showed very high apparent precision. Although this is not conclusive (this example is from an erg video rather than on-water footage), it demonstrated the potential for very high accuracy—to the point where even fine details such as individual fingers appeared to be tracked correctly.

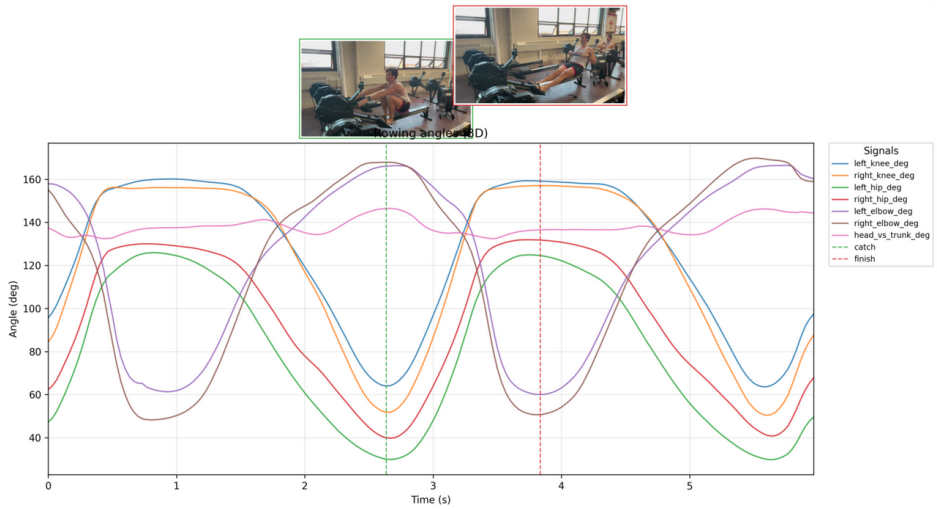

Figure 9: Sports2D erg test: extremely smooth kinematic traces and high apparent precision (promising, though not yet conclusive for on-water conditions).

Step 4: "Average stroke" synthesis (planned)

Goal: build an athlete-specific reference model representing typical technique.

- Aggregate multiple aligned strokes and compute a representative trajectory in feature space (e.g., phase-averaged joint angle profiles with variability bands).
- Use the average stroke as a baseline for measuring technical deviations and consistency across sessions.

Step 5: Performance inference via supervised sequence modeling (planned)

Goal: map video-derived kinematic features to performance targets when ground truth is available for training/validation.

1. **Ground truth acquisition (training/validation only):** pair video clips with synchronized force curves / power output from (i) stationary ergometer signals or (ii) boat telemetry.
2. **Learning problem formulation:** learn a function f such that $\hat{Y}(t) = f(\hat{F}(t))$, where $\hat{Y}(t)$ may include force curve shape, power output, efficiency proxies, and technique indicators.
3. **Model class:** use sequence models suited to time-series regression (e.g., recurrent/temporal convolution/transformer-style models), while preserving interpretability by limiting inputs to biomechanical features and reporting feature importance/sensitivity.
4. **Generalization testing:** evaluate across athletes, sessions, and filming conditions.

Step 6: Evaluation protocol (planned)

- **Numerical evaluation:** report error between predicted and ground-truth targets (e.g., waveform error for force curves; RMSE/MAE for scalar performance metrics).
- **Structural/biomechanical validity:** verify predicted motion and derived timing obey expected rowing constraints (e.g., plausible joint angle ranges; consistent event ordering catch→drive→finish→recovery).
- **Ablations:** measure sensitivity to camera viewpoint, stabilization, smoothing, and 2D-to-3D lifting assumptions.