Circular Task Analysis Using MouseReMoCo: Technical Implementation and Validation

Victor Salvat and Jinwei Zhang[a]

[a]Digital and Movement Sciences for Health, Université de Montpellier

This manuscript was compiled on December 2, 2025

We present a comprehensive analysis of circular tracking task data recorded with MouseReMoCo software, successfully reproducing reference graphics and performance metrics. Key challenges included coordinate system transformations, angular unwrapping for lap counting, and data segmentation strategies, with validation performed on both reference data and self-generated recordings.

Motor control | Circular tracking | MouseReMoCo | Data analysis

Circular tracking tasks provide fundamental insights into human sensorimotor control through movement accuracy, speed, and information processing measures.(1) This report documents our MouseReMoCo implementation, detailing the technical challenges in coordinate system transformations, angular unwrapping for lap counting, and validation strategies that achieved precise replication of reference computations.

Technical Implementation

Data Segmentation Strategy. The continuous CSV file contains five recording periods (∼20s each) separated by pause intervals. Rather than relying solely on marker timestamps, we identified that original $X \approx 0$ points serve as natural split markers.

Coordinate System Transformation. Screen coordinates transformed to center-aligned system:

$$x_c = x_{raw} - c_x, \quad y_c = y_{raw} - c_y \tag{1}$$

Angles computed with negated y for Java screen coordinates:

$$\theta = \arctan 2(-y_c, x_c) \tag{2}$$

Angular Unwrapping for Lap Counting. Angular differences corrected for $\pm\pi$ discontinuities:

$$\Delta\theta_i = \begin{cases} \Delta\theta_i - 2\pi & \text{if } \Delta\theta_i > \pi \\ \Delta\theta_i + 2\pi & \text{if } \Delta\theta_i < -\pi \\ \Delta\theta_i & \text{otherwise} \end{cases} \tag{3}$$

Number of laps computed as:

$$n_{laps} = \frac{\sum \Delta\theta_i}{2\pi} \tag{4}$$

Negative values indicate counter-clockwise motion. Our implementation matched reference values.

Precision in Metric Calculations. Performance metrics computed with numerical precision (Table 1):

Table 1. Performance metric formulas

Metric	Formula
Effective Radius	$R_e = \frac{1}{N}\sum_{i=1}^{N}\sqrt{x_c^2 + y_c^2}$
Effective Tolerance	$T_e = \sqrt{2\pi e} \cdot \text{std}(r)$
Error Percentage	$\text{Error} = 100 \times \frac{\sum \lvert\Delta\theta_i\rvert_{outside}}{\lvert\theta_{total}\rvert}$
Movement Time/Lap	$MT/lap = \frac{T_{total}}{\lvert n_{laps}\rvert}$
Index of Difficulty/Lap	$IDe/lap = \frac{\lvert\theta_{total} \cdot R_e\rvert}{\lvert n_{laps}\rvert \cdot T_e}$
Bias	$B_e = \frac{T_e}{T_{theory}}$
Info. Processing Rate	$IP_e = \frac{ID_{e,total}}{T_{total}}$

where R_e is effective radius, T_e is effective tolerance accounting for movement variability, error percentage quantifies proportion of angular distance traveled outside the target, B_e represents bias relative to theoretical tolerance ($T_{theory} = 47.0$ px), and IP_e represents throughput in bits per second.

Verification Strategies

We employed multiple verification approaches: (1) **Visual comparison** – trajectory plots matched reference figures with correct green/red coloring; (2) **Numerical validation** – computed metrics matched reference within $< 0.5\%$ error across all five periods; (3) **Cycle detection** – confirmed 5 recordings at expected timestamps (∼20s each); (4) **Cross-dataset testing** – successfully processed self-generated data with different parameters, demonstrating robustness.

Self-Generated Data Analysis

We collected MouseReMoCo recordings with modified parameters (screen: 1902×973px, radii: 326–406px) comprising 6 cycles: 3 right hand, 3 left hand.

Software operation: MouseReMoCo presents two windows: the movement window where participants track the circular target, and the mouseReMoCoDispData window displaying real-time marker events. Target boundaries account for cursor radius and tolerance, with `true_signal` indicating inside/outside status.

Key findings: Left hand trials showed 15% higher radial deviation (12.3px vs. 10.7px), suggesting reduced non-dominant hand precision. Right hand movement time decreased from 2.1s/lap (Trial 1) to 1.8s/lap (Trial 3), indicating rapid motor adaptation within three trials.

Advanced Kinematic Analysis

Beyond core metrics, we computed velocity profiles, radial deviation dynamics, and frequency-domain characteristics. Speed profiles showed mean velocities of 1.05–1.45 px/ms with characteristic acceleration-deceleration patterns. FFT analysis revealed dominant frequencies (0.04–1.08 Hz) corresponding to circular motion periodicity, with cleaner spectral peaks correlating with higher performance (Figure 1).

Discussion

This project successfully replicated MouseReMoCo circular task analysis with $< 0.5\%$ error. Primary challenges centered on understanding coordinate systems and data structure rather than programming complexity.

Major difficulties encountered:

1. Data segmentation: Initial attempts using only X-position thresholds ($|X| < 0.5$) produced distorted trajectories with cross-segment connections. Resolution required dual-condition filtering on both X and Y coordinates with noise filtering (minimum 5 points per segment), ensuring clean separation of 5 recording periods.

2. Trajectory classification: Distance-based point classification introduced artificial tolerance errors. Solution: directly utilized the boolean `true_signal` column from raw data, eliminating subjective thresholds and maintaining experimental design integrity.

3. Y-axis orientation: Plotted Y-positions appeared inverted relative to expected patterns. Cross-referencing confirmed plots accurately reflected recorded values; discrepancy attributed to instructor's data pre-processing rather than analysis error.

Conclusion: We provide a verified pipeline for circular tracking analysis with visualization, metrics, and kinematics. Collaborative GitHub workflow maintained code quality through peer review.

Team Organization

Division of responsibilities: Jinwei implemented visual reproduction and metric verification (Sections 1–2), including data loading, trajectory plotting, and algorithm validation. Victor collected self-generated data and conducted advanced analysis (Sections 3–4), including kinematics, FFT, and report authorship.

Workflow management: Victor served as project manager, creating the repository and managing merges. Jinwei worked on `visualization` and `metrics` branches, while Victor developed `analysis`. Pull requests required peer review before merging. We minimized conflicts by separating file ownership and maintained shared standards with consistent naming.

References

[1] Tifenn Fauviaux, Camille Muller, Germain Faity, Karima Bakhti, Isabelle Laffont, et al (2021). *Can a circular steering task quantify sensorimotor integration in persons with stroke?*. Congrès de la Société Française de Médecine Physique et de Réadaptation 2021, Lille, France. hal-03834500

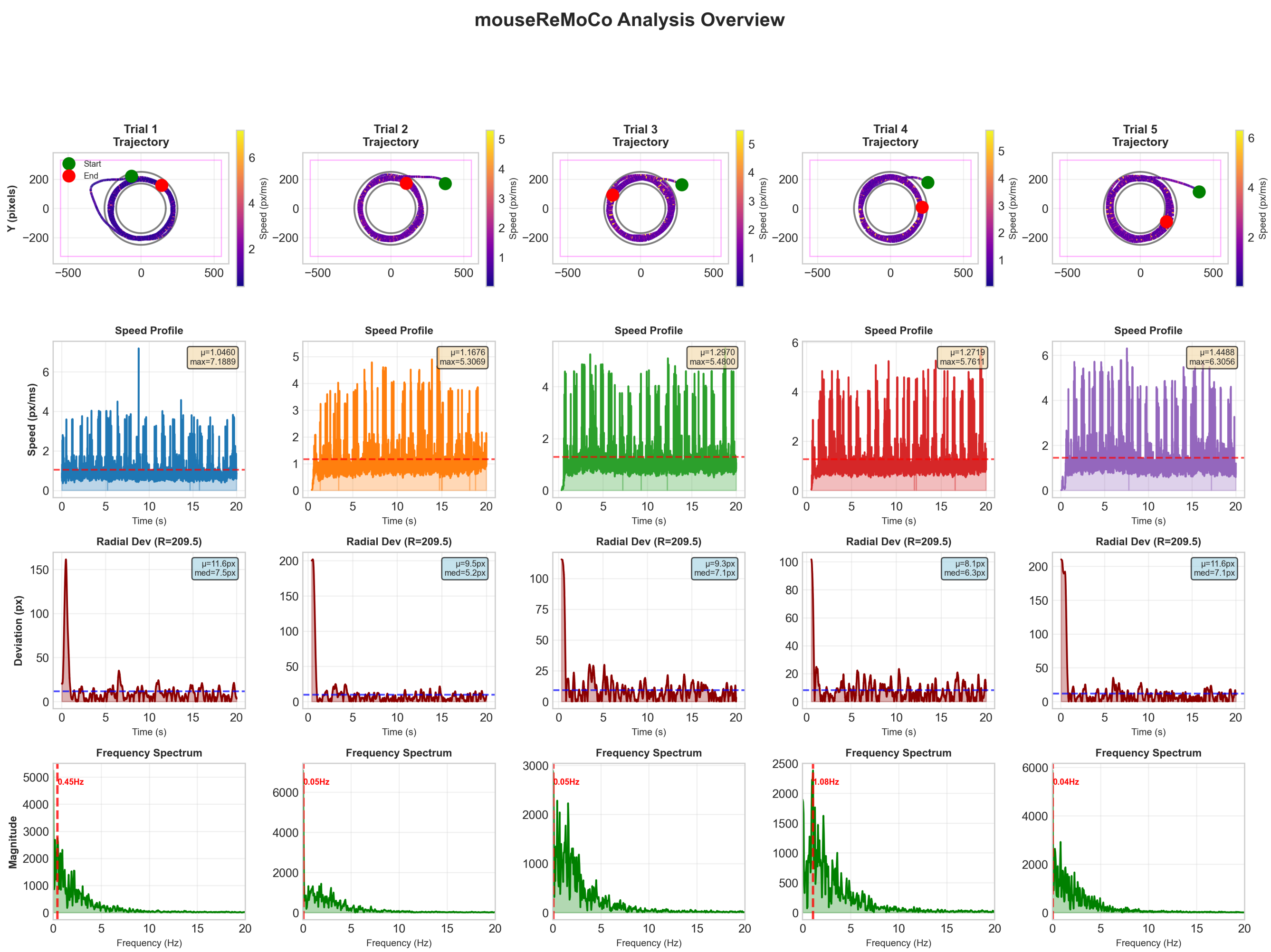

Fig. 1. Multi-trial kinematic analysis. Each column represents one trial. Row 1: speed-colored trajectories. Row 2: speed profiles over time. Row 3: radial deviation from target radius (209.5px). Row 4: FFT frequency spectra with dominant frequencies marked.

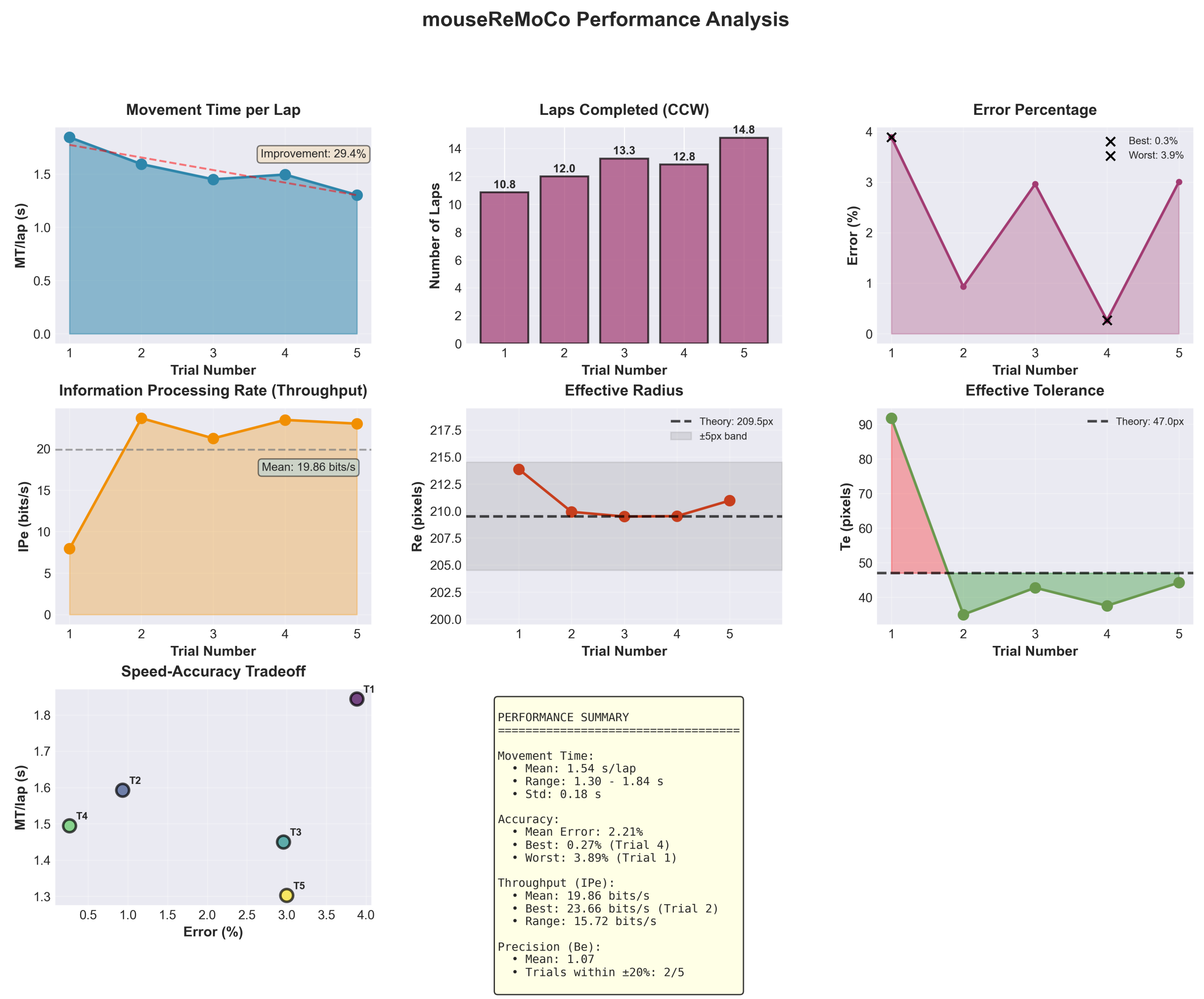

Fig. 2. Performance metrics across trials. Top: movement time, lap count, error percentage. Middle: information processing rate, effective radius, effective tolerance. Bottom: speed-accuracy tradeoff and summary statistics. Dashed lines indicate theoretical values where applicable.