

Mini-Project (ML for Time Series) - MVA 2025/2026

Adonis Jamal adonis.jamal@student-cs.fr
Fotios Kapotos fotiskapotos@gmail.com

December 14, 2025

1 Introduction and contributions

In modern sports analytics, identifying team formations is essential for interpreting tactics. However, tracking data in fluid sports like soccer is noisy; players frequently swap positions temporarily, making it difficult to distinguish permanent tactical shifts from transient adjustments. To address this, we examine SoccerCPD [2], a change-point detection framework designed to identify distinct tactical phases. The method operates in two stages: first, FormCPD detects formation changes using role-adjacency matrices derived from Delaunay triangulations. Second, RoleCPD identifies role swaps within those formations using permutation sequences.

Contributions and Work Repartition In alignment with the course requirements, this report details our reproduction and extension of the SoccerCPD framework:

- **Work Repartition:** The workload followed the pipeline's structure. One student focused on the **FormCPD** stage, implementing the Delaunay-based adjacency generation and formation change detection. The other concentrated on **RoleCPD**, handling the generation of synthetic role data, permutation analysis, and the recursive segmentation algorithm.
- **Source Code & Implementation:** We reproduced the core analysis by implementing it in Python, integrating an R-based backend for the Generalized Edge-Count test (g-segmentation) to improve detection accuracy. We also implemented a Python-based fallback for change-point detection to ensure robustness when the R-bridge encounters errors.
- **Experiments:** We verified the method on synthetic datasets, successfully detecting ground-truth change points. For real-world application, we analyzed a Barcelona match, specifically detecting tactical phases and visualizing role allocations.
- **Novel Analysis (Stationarity & Possession):** Beyond reproduction, we extended the analysis by introducing a "Player Stationarity" metric. This measures the spatial standard deviation of players to quantify how "fixed" or "fluid" a role is. Furthermore, we refined the formation analysis by separating frames based on Possession Context (Attack vs. Defense), allowing for a more granular understanding of team shape during different game states.

2 Methodology

2.1 Formation Change-Point Detection (FormCPD)

The FormCPD pipeline transforms raw player trajectories $X(t) \in \mathbb{R}^{N \times 2}$ into a temporal sequence of formation graphs $\{A(t)\}_{t \in T}$. Discrete g-segmentation is then applied to the sequence $\{A(t)\}$ to detect change-points $T_1 < \dots < T_m$ such that $A(t) \sim \mathcal{F}_i$ for all $t \in T_i$. Finally, the resulting segments are clustered through role alignment and hierarchical aggregation to extract canonical formation types.

2.1.1 Role Assignment

The first step of FormCPD assigns each player a latent spatial role following [1], aiming to model stable positional zones rather than player identities. Each role $k \in \{1, \dots, N\}$ is represented by a Gaussian component (μ_k, Σ_k) . At time t , player positions $x_{i,t} \in \mathbb{R}^2$ are modeled as

$$x_{i,t} \sim \mathcal{N}(\mu_{z_{i,t}}, \Sigma_{z_{i,t}}),$$

under the hard constraint that no two players share the same role at a given frame.

Parameters are estimated via a constrained EM algorithm. In the E-step, a cost matrix built from negative log-likelihoods is solved using Hungarian matching to assign players to roles one-to-one. In the M-step, (μ_k, Σ_k) are updated from all positions assigned to each role across time. This produces hard, frame-wise coupled assignments while enforcing temporal consistency through shared role parameters.

Figure 3 illustrates how role-based coloring reveals coherent spatial clusters even when player trajectories cross, highlighting the ability of the model to capture stable formation structure.

2.1.2 Encoding Spatial Structure via Delaunay Triangulation

Once role assignments are available, each frame is encoded as a graph describing the local spatial relationships between roles. This is obtained by computing a Delaunay triangulation of the role locations

$$V(t) = \{v_1(t), \dots, v_N(t)\} \subset \mathbb{R}^2.$$

The Delaunay triangulation connects two roles when they share an edge in the triangulated mesh, defining adjacency based solely on relative geometry without introducing distance thresholds or fixed neighborhood sizes. From this, a binary role-adjacency matrix $A(t) \in \{0, 1\}^{N \times N}$ is constructed.

Each frame is therefore represented as an undirected graph whose vertices correspond to roles and whose edges encode local spatial proximity. This representation emphasizes the *topological* structure of the formation rather than precise metric distances. Combined with role indexing, it remains invariant to player permutations: transient exchanges of positions do not alter the role-adjacency graph as long as players remain within the same spatial zones.

Figure 4 shows Delaunay triangulations computed from simulated role positions before and after a formation change from 4-4-2 to 3-4-3. The change induces a clear reconfiguration of spatial adjacency relationships.

Figure 5 presents the corresponding role-adjacency matrices. Instantaneous matrices (top row) exhibit small fluctuations due to geometric noise, whereas averaging over each formation segment (bottom row) yields stable, formation-specific patterns used as inputs for clustering.

2.1.3 Change-Point Detection via Discrete g-Segmentation

Formation change-points are detected on the temporal sequence of role-adjacency matrices $\{A(t)\}$ using discrete g-segmentation [5], a graph-based CPD method suitable for high-dimensional and discrete observations. Pairwise dissimilarities are measured with the Manhattan distance

$$d_M(A(t), A(t')) = \|A(t) - A(t')\|_{1,1}.$$

Frames with excessive role switching (rate > 0.7) are removed to discard abnormal game situations. The scan statistic identifies candidate change-points τ , which are retained as significant if they satisfy constraints on p -value (< 0.01), segment duration (at least 5 minutes on both sides), and inter-segment dissimilarity. A recursive procedure is applied to recover multiple change-points, yielding a partition of the match into formation intervals $T_1 < \dots < T_m$.

2.1.4 Formation Clustering

Each formation segment T_i is represented by a formation graph $F(T_i) = (V(T_i), A(T_i))$, obtained by averaging the role locations and role-adjacency matrices over the stable frames $t \in T_i^*$.

When comparing formations across segments or matches, role indices are not directly comparable since labels are arbitrary. We therefore align roles between two formation graphs by solving an optimal assignment problem using the Hungarian algorithm on the Euclidean distances between vertex sets $V(T_i)$ and $V(T_j)$, yielding a permutation matrix Q . This alignment ensures that homologous spatial roles are matched prior to comparison.

Formation similarity is then measured using the Manhattan distance between aligned adjacency matrices,

$$d(F_i, F_j) = \|QA(T_i)Q^\top - A(T_j)\|_{1,1},$$

which captures differences in *topological structure* rather than absolute positioning. Based on these pairwise distances, agglomerative hierarchical clustering is applied to group formation graphs into a small number of canonical formation types.

2.2 Role CPD

The goal of RoleCPD [2] is to detect long-term tactical changes in player roles (e.g., a winger swapping sides with another winger permanently) while ignoring temporary switches (e.g., overlapping runs or covering defensive duties). This process operates within a single Formation Period (T_i) identified in the previous step. Figure 6 in Appendix B illustrates the ground truth of such role assignments with simulated tactical switches.

2.2.1 Formal Representation

We define the inputs and mathematical framework based on the SoccerCPD protocol:

Input: A sequence of "Temporary Role Permutations" $\{\pi_t\}_{t=1}^{|T_i|}$.

Role Permutation (π_t): At every frame t , the Role Representation step assigns a role X_p to every player p . Since roles are distinct, this assignment is a permutation of the initial canonical role:

$$\beta_t(p) = \pi_t(X_p)$$

Where β_t is the player-to-temporary-role mapping at time t .

2.2.2 The Distance Metric

To detect a change, we must quantify the difference between the team's configuration at time t and time t' . Since the data consists of permutations (non-Euclidean), we cannot use standard Euclidean distance. We use the Hamming Distance normalized by the number of roles ($N = 10$ outfield players):

$$d(\pi_t, \pi_{t'}) = \frac{1}{N} \sum_{p \in P} \mathbb{1}_{\pi_t(X_p) \neq \pi_{t'}(X_p)}$$

This metric represents the "Switch Rate" or the proportion of players whose roles differ between two frames. Figure 7 plots this switch rate over time, showing it fluctuates relative to a dominant permutation.

2.2.3 Change-Point Detection Algorithm

The paper utilizes Discrete g-segmentation, a graph-based change-point detection method effective for repeated observations in non-Euclidean space. The procedure is as follows:

1. **Preprocessing:** Calculate the Switch Rate relative to the dominant permutation. Exclude frames with a switch rate > 0.7 (likely temporary switches during set-pieces or abnormal situations).
2. **Segmentation:** Apply the detection algorithm recursively on the sequence of permutations using the Hamming distance. Figure 8 shows the result of a single segmentation step, while Figure 9 demonstrates the final output after recursive segmentation captures all tactical phases, accurately recovering the ground truth on the synthetic example.
3. **Significance Test:** A change point τ is significant if:
 - The p-value of the scan statistic is < 0.01 .
 - The segmentation duration is sufficient (robustness against noise).
 - The most frequent permutations (Instructed Roles) in the segments before and after τ must be distinct.

3 Data

Our study utilizes two distinct types of spatiotemporal datasets to evaluate the SoccerCPD framework: high-frequency GPS tracking data as used in the original SoccerCPD study, and event-stream data. This section analyzes the properties, quality, and preprocessing requirements of these signals.

3.1 High-Frequency Tracking Data

For the reproduction of the core algorithm and methods, we utilized the sample dataset provided by the authors. Though usually scarce, we found public tracking data for 10 matches from SkillCorner [3], and GPS data of one play from Last Row [4]. The data characteristics are as follows:

Source & Format: The Fitogether data originates from the K-League (South Korea) and consists of player trajectories recorded at 10 Hz. The raw input is a sequence of 2D coordinates $x_{i,t} \in \mathbb{R}^2$ representing the position of player i at frame t . The SkillCorner data is also at 10 Hz, and comes from the Australian A-League, in the .jsonl format. The Last Row data is a .csv at 20 Hz, but only contains a single play segment. Both datasets were converted to the Fitogether format for consistency.

Data Diagnosis & Quality: We performed a preliminary diagnosis of the signal stationarity and completeness. The tracking data is relatively clean, with practically no gaps (missing values < 1%). However, we observed high-frequency jitter in the raw trajectories, consistent with GPS measurement noise.

Preprocessing: To satisfy the model’s assumption that role centers follow a Gaussian distribution, we normalized the pitch coordinates to a centered metric system. Testing the assumption that player positions within a role follow a Gaussian distribution, Shapiro-Wilk tests on the specific role clusters (e.g., Center Back) showed deviations from normality, likely due to behaviours such as tactical pressing or zonal marking which create skewed distributions. However, the clusters remain sufficiently separable for the EM algorithm to converge.

3.2 StatsBomb Event Data

While public high-frequency GPS tracking data is scarce and contains little information besides player locations, datasets such as the StatsBomb Open Data repository present event-stream data. These datasets differ fundamentally, presenting unique challenges for the SoccerCPD pipeline.

3.2.1 Signal Characteristics and Challenges

Unlike the continuous 10 Hz GPS signal, StatsBomb data is event-based, recording location (x, y) only when an on-ball action occurs (e.g., pass, shot, dribble).

Sparsity: The primary diagnosis in Figure 10 reveals extreme sparsity. While GPS data provides ≈ 600 points per minute per player, event data provides fewer than 10 points per minute on average. This challenges the EM algorithm used in the Role Assignment step, which relies on dense spatial clusters to estimate role means μ_k and covariances Σ_k .

Spatial Bias: Event data is inherently biased towards the ball’s location. Defenders engaging in off-ball marking are often unrecorded in the event stream.

3.2.2 Adaptation and Preprocessing

To adapt this data for the SoccerCPD framework, we applied the following transformations:

Coordinate Transformation: StatsBomb coordinates are given in a $[0, 120] \times [0, 80]$ system with the origin at the top-left corner. We projected these to the centered metric system used by the Fitogether model to ensure consistency.

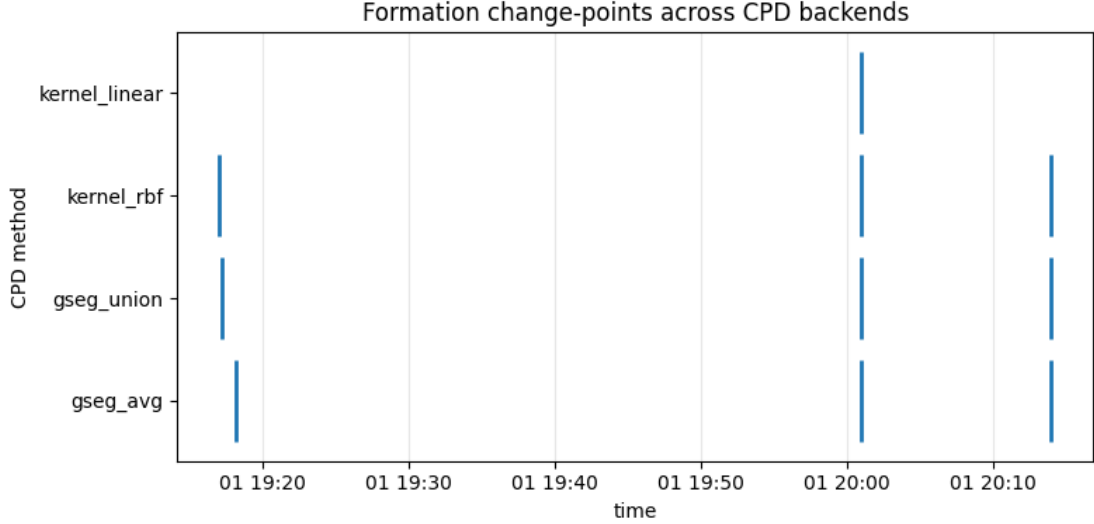


Figure 1: Formation change-points detected by different CPD backends on a common timeline.

Pseudo-Trajectory Generation: To mitigate sparsity, we aggregated events over temporal windows. Instead of instantaneous frame-by-frame analysis, we treated the collection of event locations within a time window as a sampling of the underlying spatial distribution \mathcal{F}_i .

Role Proxy: Since we lack continuous tracks for all 10 players simultaneously, we modified the input feature matrix $A(t)$. Instead of a frame-wise Delaunay graph, we constructed aggregate adjacency matrices based on average positions of players over 5-minute segments, serving as a proxy for the mean role-adjacency matrix.

4 Results

Comparison across CPD backends (single match). We compare several formation CPD backends (gseg_avg, gseg_union, kernel_rbf, kernel_linear) on a single reference match, using identical preprocessing and hyperparameters. The goal is not to rank methods, but to assess whether different CPD formulations yield compatible formation change-point timelines.

Figure 1 shows the detected formation change-points for each backend on a common timeline. Despite differences in underlying detection principles, most methods identify change-points at similar temporal locations, indicating a shared high-level segmentation of the match. Minor discrepancies mainly correspond to additional or missing change-points rather than large temporal shifts, suggesting differences in sensitivity rather than fundamentally different interpretations of the match dynamics.

Overall, this experiment indicates that formation change-points are largely consistent across CPD backends for this match. Since the analysis is restricted to a single game, these observations should be interpreted qualitatively and primarily serve to illustrate methodological robustness rather than general performance.

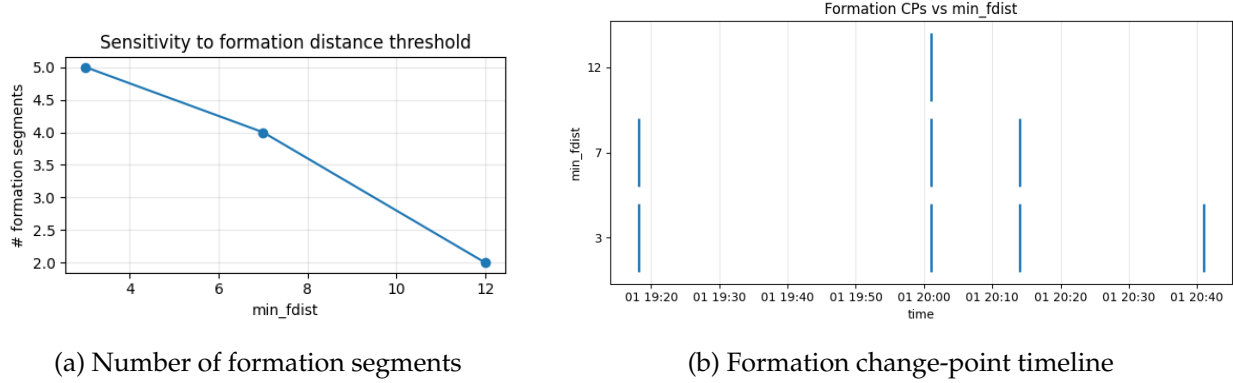


Figure 2: Sensitivity of formation change-point detection to the formation distance threshold `min_fdist`. Increasing the threshold reduces the number of detected segments while preserving the temporal locations of the most salient change-points.

4.1 Experiments using SoccerCPD

Sensitivity analysis of formation CPD (single match). We evaluate the sensitivity of the formation change-point detection (CPD) pipeline to its main hyperparameters on a single reference match. All experiments use the same CPD backend (`gseg_avg`), and one parameter is varied at a time while keeping the others fixed.

We first study the impact of the formation distance threshold `min_fdist`, which controls the minimum Manhattan distance between mean formation graphs required to validate a change-point. As `min_fdist` increases, the number of detected formation segments decreases monotonically, indicating that higher thresholds progressively filter out minor structural variations. Importantly, the remaining change-points occur at consistent temporal locations, suggesting that `min_fdist` mainly acts as a pruning mechanism rather than inducing temporal instability.

We then analyze sensitivity to the minimum period duration `min_pdur`, which enforces a lower bound on segment length. Over the tested range, both the number and locations of detected change-points remain unchanged, indicating that the formation changes identified in this match are temporally well separated and not driven by short-lived fluctuations.

Overall, these results suggest that, for this match, formation change-points are robust to reasonable variations in temporal resolution and structural significance thresholds. Since the analysis is conducted on a single match, these findings should be interpreted as illustrative rather than representative, motivating a broader multi-match evaluation.

4.2 Extensions to SoccerCPD

To enhance the interpretability of the tactical segments identified by SoccerCPD, we introduced two novel analytical layers: a contextual separation of formations based on possession and a quantifying metric for role fluidity.

4.2.1 Possession Context: Attacking vs. Defending Shapes

A limitation of the standard FormCPD output is that it produces a single "average" formation for a detected time segment. However, modern soccer teams adopt vastly different shapes depending

on possession status. We extended the pipeline to filter frames based on the possession context (Attack vs. Defense, based on a centroid threshold since possession event data is unavailable for GPS tracking) before computing the formation centroids.

- **Attacking Formation:** Computed using only frames where the team is in possession.
- **Defensive Formation:** Computed using frames where the opponent is in possession.

The results, visualized in Figure 11 (Appendix D) for Match ID 1925299 (A-League, SkillCorner) and Figure 12 (FCBarcelona, StatsBomb), reveal significant structural differences that are lost in a global average. For both matches, the attacking shape is wider and higher up the pitch, whereas the defensive shape compresses, with wingers dropping deeper to form a compact block. This granular analysis confirms that a single "stable" phase detected by SoccerCPD actually comprises two distinct, alternating tactical configurations. The StatsBomb FCBarcelona match illustrates this particularly well, with the team shifting from a high-pressing 4-3-3 when attacking to a more conservative 4-4-2 block when defending. It must be noted, however, that the sparsity of event-stream data limits the precision of these formation estimates compared to high-frequency tracking data. This is why we see a player such as Pjanic, who typically plays as a central midfielder, being detected in a very deep defensive position in both the attacking and defensive formations of Phase 1. The events involving him mostly occur when the team is defending, leading to a biased estimation of his average position. That is not the case in phase 2, where he is detected in a more reasonable midfield position.

4.2.2 Player Stationarity Metric

While the original SoccerCPD model assigns a Gaussian mean (μ_k) and covariance (Σ_k) to each role, it does not explicitly quantify how strictly a player adheres to that role. We defined a Player Stationarity metric, calculated as the standard deviation of the Euclidean distance between a player's actual position and their assigned role's center (μ_k) over a formation phase. A lower value indicates a "fixed" positional role (e.g., a center-back holding the line), while a higher value indicates a "fluid" or "roaming" role. Applying this to the sampled StatsBomb FC Barcelona match, we observed distinct behavioral patterns. As shown in Figure 14 in Appendix D, midfield roles exhibited higher stationarity (lower deviation) compared to defensive fullbacks or attacking roles. For instance, during Phase 1, the average spatial deviation was approximately 9.8m, but specific attacking and even defensive roles showed deviations exceeding 12m, quantifying the "free-roaming" nature of Barcelona players compared to their midfielders, who distribute the ball and keep possession. These results can also be compared to phase 2 in Figure 15, where the average spatial deviation increased, as players were much more fluid in both attack and defense.

We also validated this metric on the A-League tracking data (Match ID 1925299), as illustrated in Figure 13. The metric successfully captures the variance in player movement within a stable tactical phase, providing coaches with a concrete measure of positional discipline.

5 Conclusion

This report presented a comprehensive reproduction and extension of the SoccerCPD framework for detecting tactical formation changes in soccer using spatiotemporal data. We successfully implemented both stages of the pipeline, FormCPD and RoleCPD, verifying their effectiveness on synthetic datasets and real-world matches. Our contributions included the introduction of a

player stationarity metric to quantify role fluidity, as well as the refinement of formation analysis by separating frames based on possession context (attack vs. defense). These extensions provided deeper insights into team tactics and individual player behaviors. We finally applied the adapted framework to event-stream data from the StatsBomb repository, demonstrating its versatility across different data types. Overall, our work validates the SoccerCPD approach while enhancing its interpretability and applicability in football analytics.

References

- [1] Alina Bialkowski, Patrick Lucey, Peter Carr, Yisong Yue, Sridha Sridharan, and Iain Matthews. Large-scale analysis of soccer matches using spatiotemporal tracking data. In *Proceedings of the IEEE International Conference on Data Mining (ICDM)*, pages 725–730. IEEE, 2014.
- [2] Hyunsung Kim, Bit Kim, Dongwook Chung, Jinsung Yoon, and Sang-Ki Ko. Soccercpd: Formation and role change-point detection in soccer matches using spatiotemporal tracking data. In *The 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 3146–3156. ACM, 2022.
- [3] SkillCorner. SkillCorner Open Data: Broadcast tracking data. <https://github.com/SkillCorner/opendata>, 2020. Accessed: 2025-01-04.
- [4] Ricardo Tavares and Friends of Tracking Data. Last-Row: Sample tracking data and code. <https://github.com/Friends-of-Tracking-Data-FoTD/Last-Row>, 2020. Accessed: 2025-01-04.
- [5] Jingru Zhang and Hao Chen. Graph-based two-sample tests for data with repeated observations, 2019.

A Appendix: Visual Examples for FormCPD

This appendix gathers all visual material related to the FormCPD pipeline (Role assignment, De-launay encoding, and adjacency representations) referenced in Section 2.1.

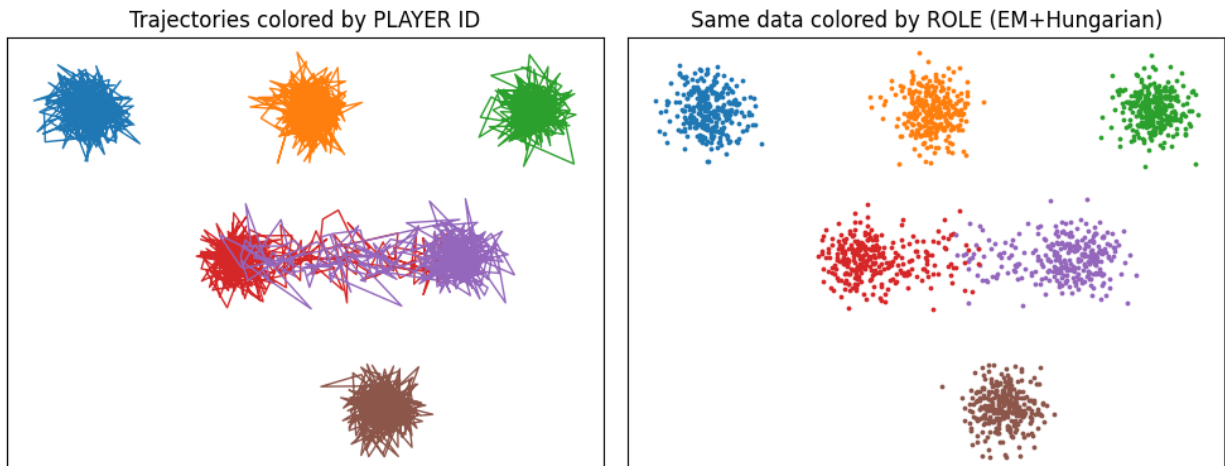


Figure 3: Player versus role-based coloring of trajectories during a simulated position swap.

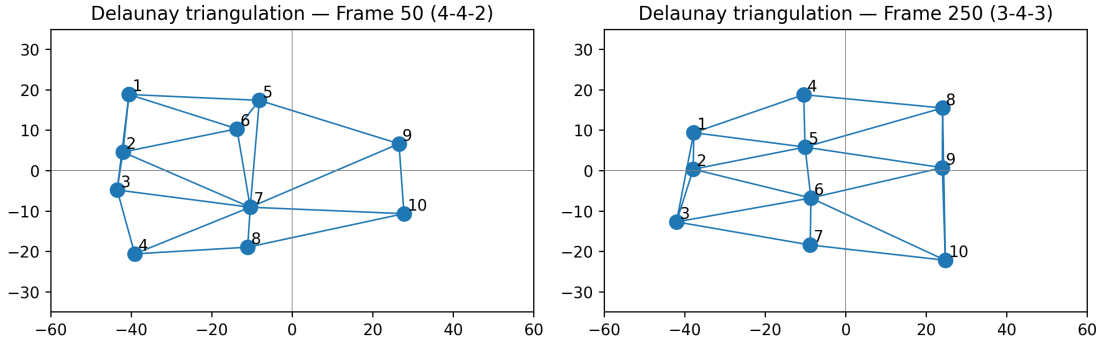
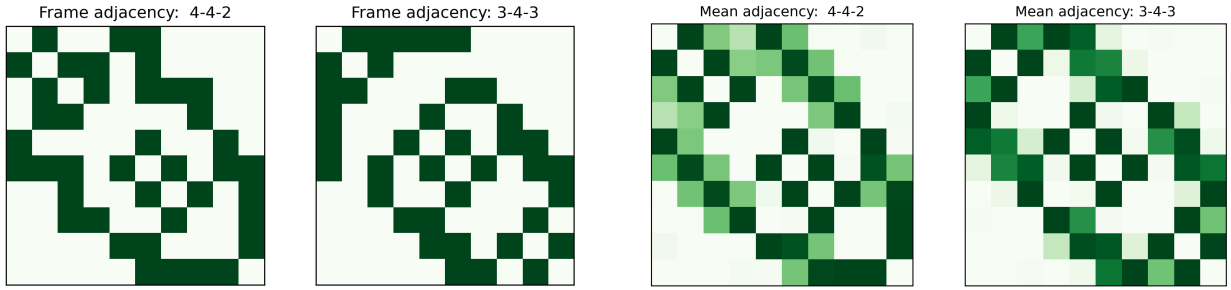


Figure 4: Delaunay triangulations of role locations at two frames before and after a synthetic formation change (4-4-2 \rightarrow 3-4-3).



(a) Single-frame binary adjacency matrices before and after the simulated formation change (4-4-2 \rightarrow 3-4-3) (Green = 1).

(b) Mean role-adjacency matrices averaged over each formation period (Green = 1).

Figure 5: Delaunay-based adjacency representation on a synthetic formation change.

B Appendix: Visual Examples for RoleCPD

This appendix gathers all visual material related to the RoleCPD pipeline (Hamming distance, Change Point Detection, and Segmentation) referenced in Section 2.2.

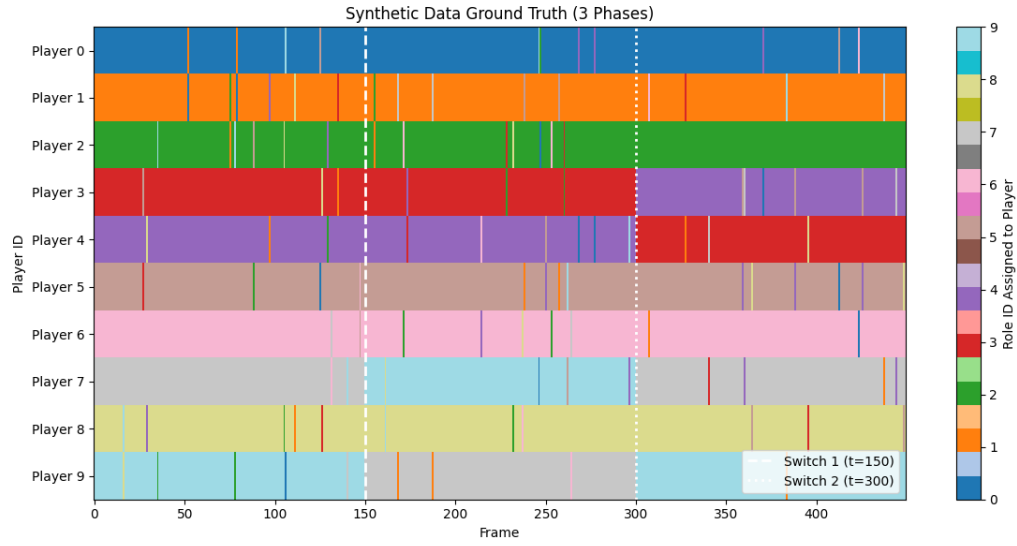


Figure 6: Ground truth role assignments over time with two tactical switches at frames 150 and 300.

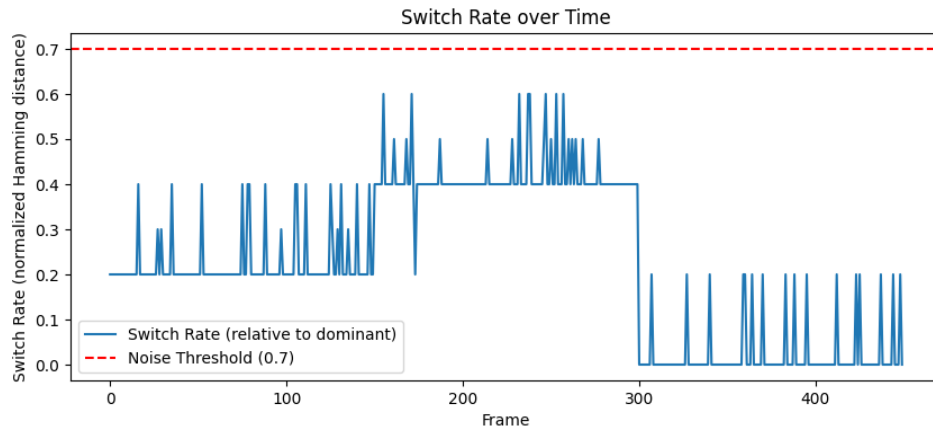


Figure 7: Hamming distance-based switch rate time series used for change point detection.

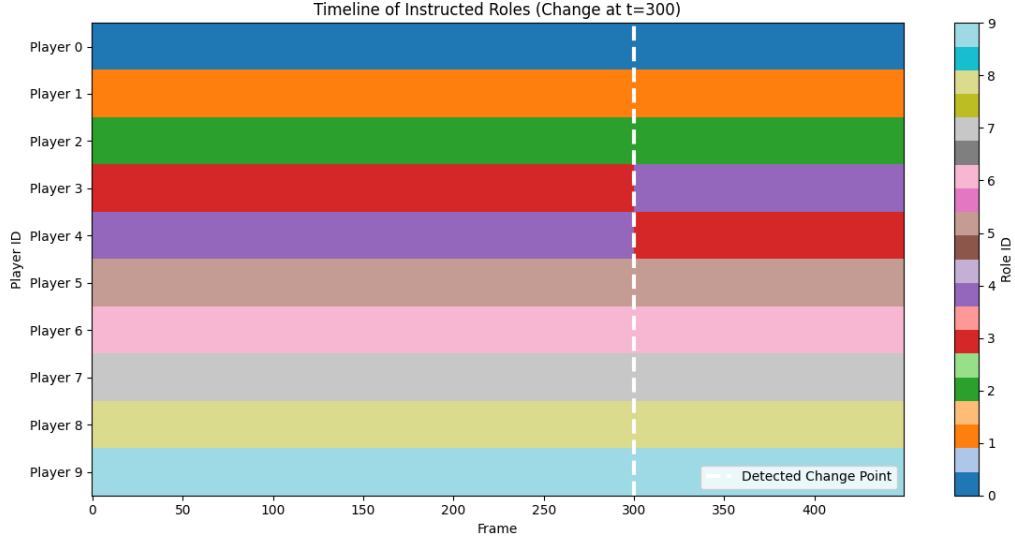


Figure 8: Visualization of one detected tactical phase with dominant role assignments. This shows the need for recursive segmentation to capture multiple phases.

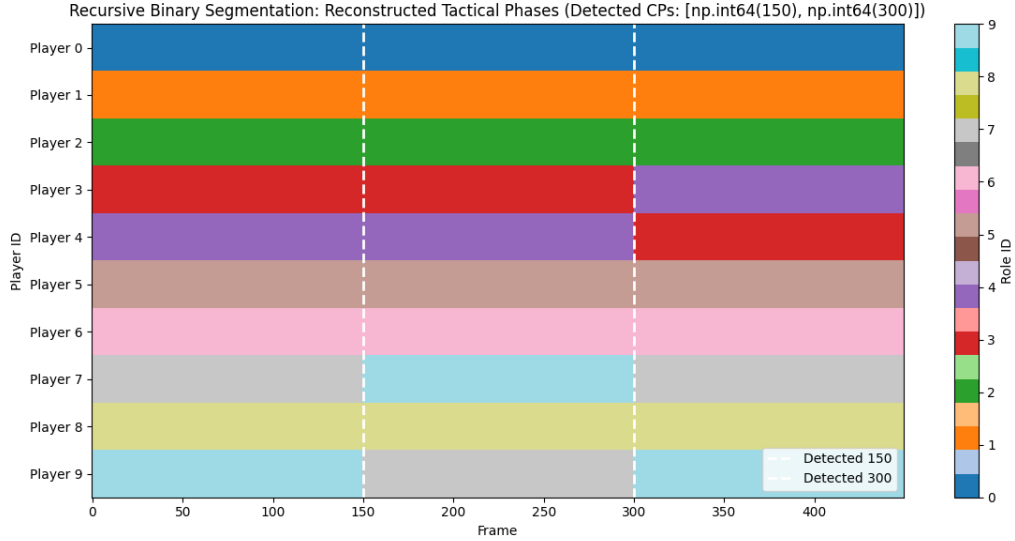


Figure 9: Visualization of all detected tactical phases after recursive binary segmentation, showing accurate recovery of the ground truth phases.

C Appendix: Visual Results for Data Analysis

This appendix gathers all visual material related to the data analysis referenced in Section 3.

Spatial Density and Signal Structure Comparison

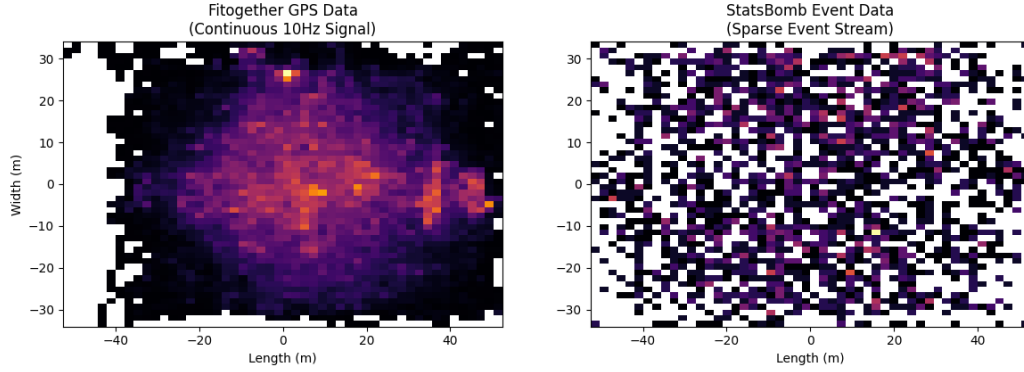


Figure 10: Spatial density comparison between continuous Fittogether GPS tracking data and StatsBomb event-based data (right). The GPS data shows continuous coverage of player positions, while the event data is sparse and concentrated around ball actions.

D Appendix: Visual Results for Experiments

This appendix gathers all visual material related to the experiments referenced in Section 4.

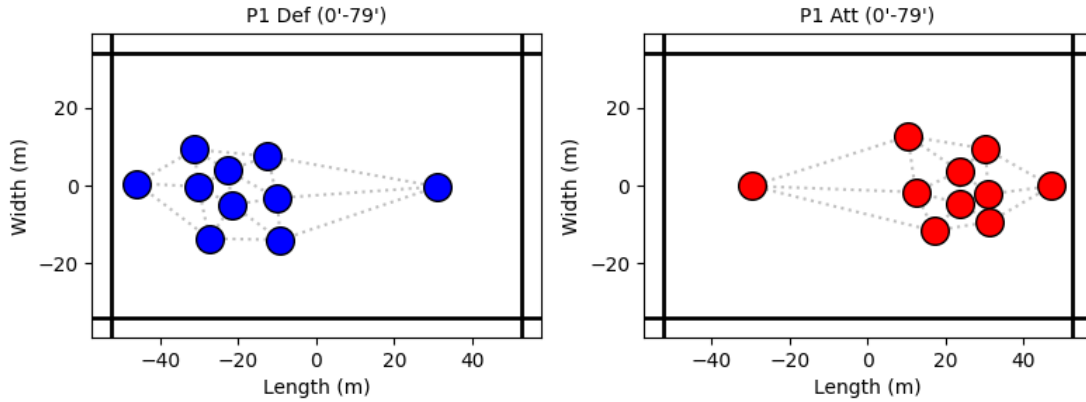


Figure 11: Detected formations for match "1925299" (A-League, SkillCorner) separated by phase of play: defending (left) vs. attacking (right). This highlights tactical adjustments based on possession context.

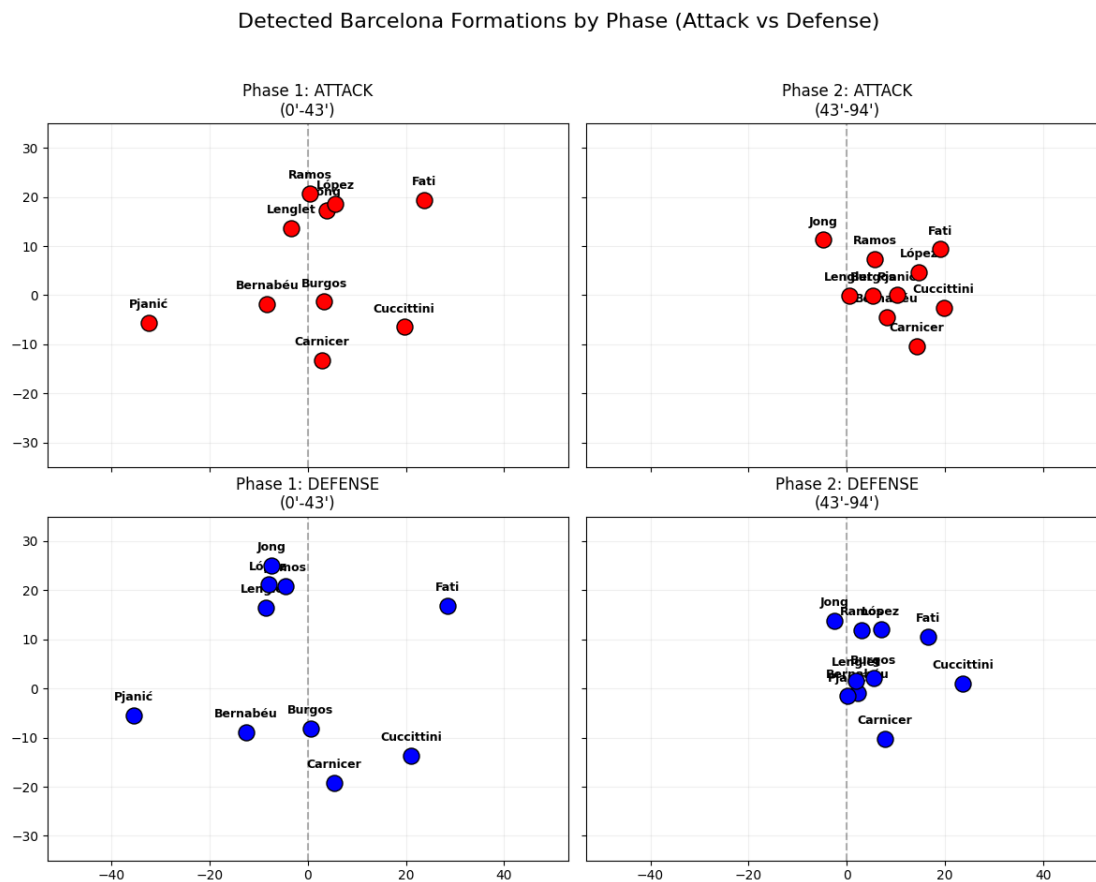


Figure 12: Detected formations for FC Barcelona using event-stream StatsBomb data separated by different tactical phases within the match. Each phase reflects a distinct formation strategy employed by the team.

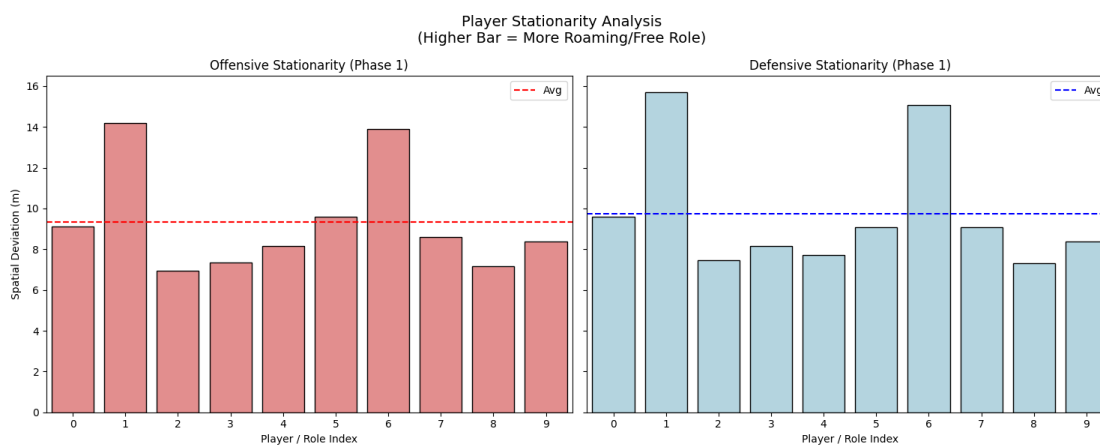


Figure 13: Player stationarity during Phase 1 of match "1925299" (A-League, SkillCorner), illustrating the consistency of player positions within the tactical phase.

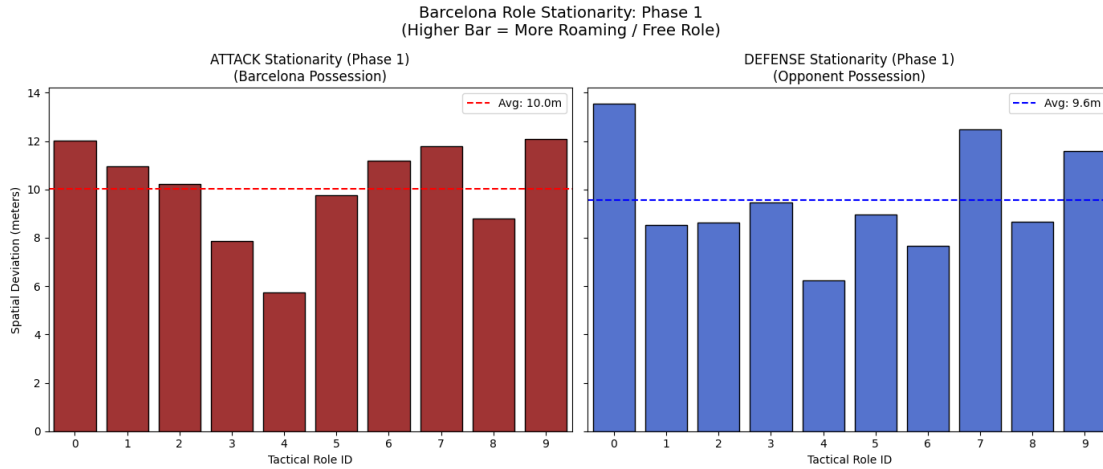


Figure 14: Player stationarity during Phase 1 of FC Barcelona match, illustrating the consistency of player positions within the tactical phase, even with event-stream data.

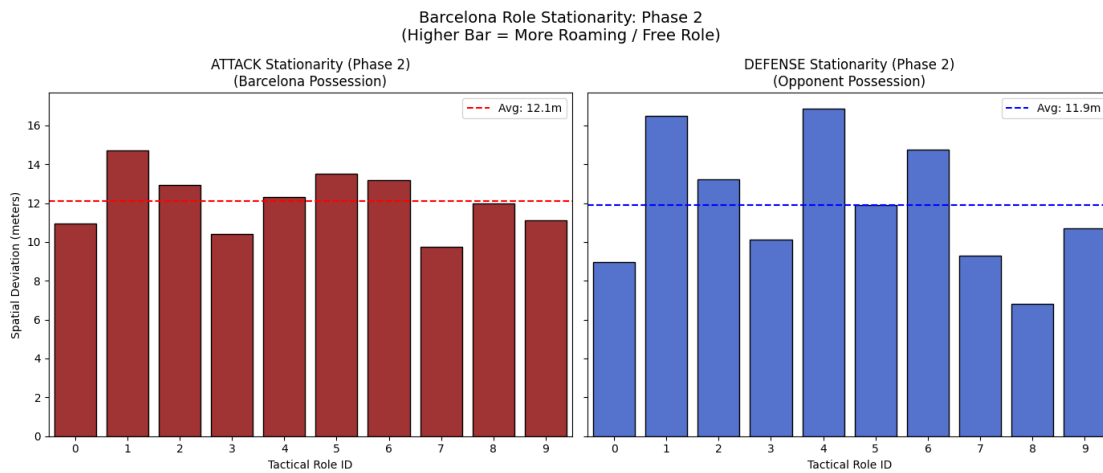


Figure 15: Player stationarity during Phase 2 of FC Barcelona match, illustrating the consistency of player positions within the tactical phase, even with event-stream data.