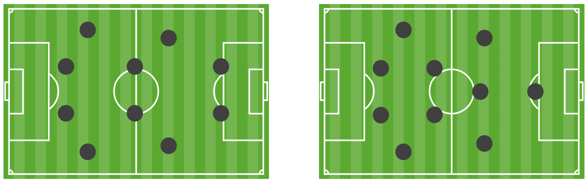
Formation2Vec: Exploring a Representation for Formation Segmentation and Detection in Soccer Games

Abstract

In the soccer game, formation depicts the role of the soccer players on the pitch, revealing the primary tactic of the teams. During a game, a team will switch between offensive and defensive formations as they get or lose the ball. Although the analysis of formation changing is important in soccer, there are few methods to detect and classify the formations automatically. In this paper, we analogize the formation detection to the human action segmentation and classification problem. Referring to the general framework to solve such a problem, we firstly adopt a neighborhood-aware method to model the positional data and obtain the topological structure of the players, which can help represent the position. Our experiments prove that a good parameter selection for our representation can capture the characteristics of the relative positions. Then by a convolutional network, the temporal patterns will be captured, enabling us to segment the game into the periods with specific formations. The accuracy of the method achieves a satisfactory level in the sequence segmentation and classification area. Compared with the existing method in the formation detection, our method supports fine-grained analysis of formation changes and shows a wider application prospect.

1 Introduction

Soccer is one of the most popular sports in the world with huge commercial value. This 11-a-side competition shows a high degree of confrontation and dynamic. The players on the pitch (except goalkeeper) are organized by team **formation**, which is arranged by the coach considering the characteristics of the players and the situation of the game. Formation indicates the role of the players on the pitch. For example, formation 4-4-2 means 4 backwards, 4 midfielders, and 2 forwards, as Figure 1a shows; Figure 1b shows formation 4-2-3-1 with 4 backwards, 2 defensive midfielders, 3 offensive midfielders, and 1 forward. The players will try to keep their relative positions according to the formation. During a game, the formation changes frequently. As the team obtains the ball, the players will move forward and the number of people in the frontcourt will increase. A proper formation helps the team control the midfield and generate fierce attacks. When the opponent gets the ball back and approaches the penalty area, the team will adopt a deep defensive formation, like 6-3-1, to enhance the defense. If the defensive formation is not held tightly, leaving space for the opponents, the backwards will be more likely to make mistakes and lose the game.



(a) $4-4-2$ (b) $4-2-3-1$

Figure 1: The relative position of the players with formation 4-4-2 and 4-2-3-1.

As the formation plays a key role in soccer games, many works tried to analyze the games from the perspective of formation. [Bialkowski *et al.*, 2014a] analyzed the formation of a team when it is at home and away and the influence on the results of the game. [Wei *et al.*, 2013] used the formation as prior knowledge to explore the offensive and conceding patterns of a team. These above works assume that a team generally adopts one or two formations in a game. [Wu *et al.*, 2019] developed a visual analytic system called ForVizor for dynamic formation analysis, and one challenge in this work is the access to fine-grained formation data. Few works in the literature focused on formation detection, and the most approved one is the algorithm proposed by [Bialkowski *et al.*, 2014b]. They used a method equivalent to k-means to aggregate the sequential positional data into Gaussian distributions, and then the agglomerative clustering is adopted for formation detection.

Two major challenges in formation segmentation and detection are feature selections and temporal pattern learning for dynamic data. First, it is difficult to extract an effective high-level feature representation to characterize the highly dynamic and unstructured positional data. Due to the unstructured nature of the positional data, the available feature extraction algorithms in image processing and network embedding cannot be directly applied to this problem. Second, for segmentation, as the formation change of a team is a continuous and evolving process, capturing the transformation of formation and ignoring the subtle changes of the features pose a significant challenge to a segmentation model. The semantic pattern in formation is hard to define.

We address the first challenge by proposing a neighborhood-aware representation. The idea comes from the observation of the players on the pitch. A player cannot keep aware of his position in the formation during the game. To preserve it, he must watch the relative positions of his teammates to determine if he is at the right position and keeps a proper distance from other players. During the game, every player adjusts the position dynamically to hold the formation.

To address the second challenge, we adopt a temporal convolutional network (TCN), a supervised manner, to learn the temporal patterns. TCN is a variant of convolutional network designed for sequential data. As the convolutional networks achieve good performance in image segmentation and classification, [Gehring *et al.*, 2017] from Facebook and [Kalchbrenner *et al.*, 2016] from Google successfully applied CNNs to machine translation. [Lea *et al.*, 2017] proposed TCN and achieved better results than the previous state of the art in the field of human action detection and segmentation. [Bai *et al.*, 2018] evaluated TCNs in wider range of problems and demonstrated the strengths of TCNs, but the application of TCNs regarding formation detection have not been discussed.

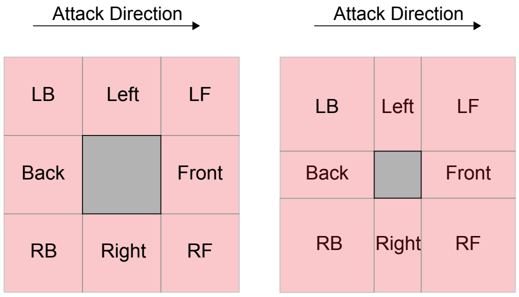
Our main contributions are as follows. Firstly, we define the problem of formation detection as a sequence segmentation and detection problem. Secondly, we propose a representation suitable for the formation segmentation and classification. Thirdly, we apply the TCN to formation data and obtain satisfactory results compared to the baseline method.

2 Problem Definition

2.1 Data Preprocessing

Our data are collected from several panoramic videos of soccer games. We use a semi-automatic collecting method to label the position of the players in the videos. To preserve the efficiency and precision, we run a particle filter-based tracking algorithm, developed by [Dearden *et al.*, 2006], under the supervision of the annotators. When occlusion happens and the tracker losses the target, the annotators will manually stop, correct the position of the tracker, and make sure the tracking data are accurate. Data smoothing is adopted to avoid drifting and reduce the noise.

With the acquired position data of the players in the panoramic videos, we further use an affine projection to map the position data to a two-dimensional soccer pitch. According to the latest *Law of the Game 2018/19*[1] by the International Football Association Board (IFAB), the length of the pitch (touchline) is between 100m and 110m and the width (goal line) is between 64 and 75m. However, in many important international competitions, such as 2018 Russia World Cup, the pitch dimension is 105m by 68m. To keep the degree of accuracy, we map the positional data into a 1050 × 680 area.



(a) Range 1

(b) Range 2

Figure 2: An illustration of Moore Neighborhood.

The events on the pitch, such as foul, out of line and goal, are important for game analysis. The offensive or defensive state of a team can be interrupted by the events as the game will stop and restart. With the events, we segment the game into small periods, and the time between the periods is stoppage time. Each period is independent to other periods. As a result, the position change within a period can be viewed as a sequential process.

To label the formation information of the game, we closely collaborate with the domain experts in soccer, two of whom were previous professional soccer players in the Chinese Football Association Super League, the first-class league in China. They watch the soccer game videos and label the time when formations exist with an assistant system. The rest of the time is labeled with no formation.

2.2 Problem Definition and Parameters Description

With the labeled events on the pitch, the stoppage time is eliminated and the whole game is divided into representation sequences $\{S_i\}(i = 1, 2, \cdots, N)$, and N is the amount of the sequences. The sequence $S_i = [r^i_1, r^i_2, \cdots, r^i_{N_i}], r^i_j \in \mathbb{R}^k$, where N_i is the length of the sequence and k is the dimension of the representation (or feature) space. Each sequence S_i is assigned with the labels $l_i = [l^i_1, l^i_2, \cdots, l^i_{N_i}]$ and the label $l^i_j \in L$, where label set $L =$ [No Formation, 4-4-2, 4-2-3-1, 2-3-2-3, $\cdots$]. Given a sequence, the classification function is defined as $f : \mathbb{R}^{t\times k} \to L^t$. The formation segmentation and detection is defined as an optimization problem

$$\arg\min_{f} \sum_{i=1}^{N} \|f(S^i) - l_i\|. \tag{1}$$

3 Proposed Method

Modeling the player connections in a team is not a new problem. Ball passing is the most direct relation between players on the pitch. With ball passing, [Gonçalves *et al.*, 2017] analyzed the ball passing network of the players. [Wang *et al.*,

[1] http://www.theifab.com/laws

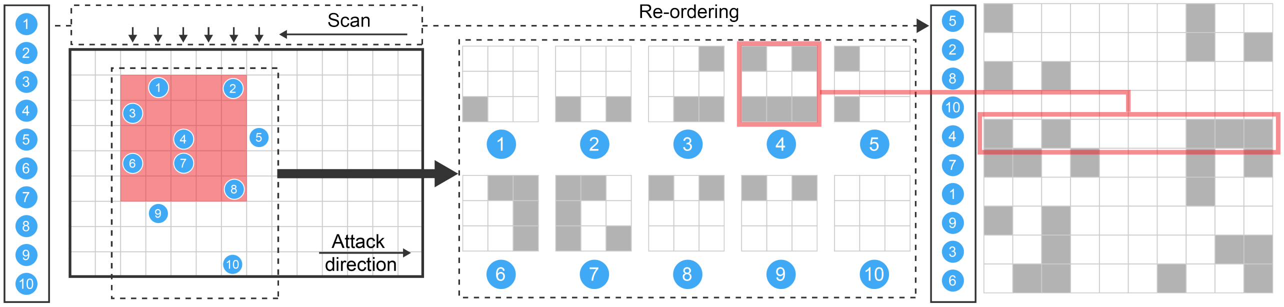


Figure 3: An illustration of formulating the neighborhood-aware representation

2015] developed a specific topic model and explored the passing patterns of Barcelona. [Decroos *et al.*, 2018] defined the phases of events and used hierarchical agglomerative clustering to extract the attacking patterns. However, these methods cannot be directly applied to the formation analysis. According to the domain experts, the players in the same line, such as backward line, do not necessarily pass a ball frequently during a game. In contrast, they are required to pass the ball forward. Therefore, we focus on the position of players, which is the most relevant factor to formation.

Based on the observation, we propose a novel framework with two major components, namely, position representations and temporal convolutional networks. The position representation component extracts the relative position of players as features. With the features, a temporal convolutional network is employed to further extract the temporal patterns. At the end of this section, we introduce a baseline model ([Bialkowski *et al.*, 2014b]) for this problem, against which we compare our model.

3.1 Position Representation

Position Scaling

Given the preprocessed position of the players, the most direct and intuitive way to represent the position of the players is using a parameterized pitch. The soccer pitch is parameterized into $(W + 1) \times (H + 1)$ grids and the position of the players are normalized into a $[0, W] \times [0, H]$ area of integer. However, with the data dimension of $(W+1)\times(H+1)$, only 10 grids have value one, and the rest of them are filled with zeros. The sparsity of the representation affects the result of segmentation and classification. With a scaling rate s, the size of the pitch is scaled into $\lceil W/s \rceil \times \lceil H/s \rceil$, while the position of the players are scaled into $(\lceil x/s - 0.5 \rceil, \lceil y/s - 0.5 \rceil)$ (rounding). The scaling rate depicts the occupancy area of a player, and a larger scaling rate means a larger occupancy area of the players. In our experiment, we set $W = 106$ and $H = 69$ as a baseline, and explore how the scaling rate s affects the results.

Neighborhood-aware Representation

To capture the relative position of the players, we borrow the idea of Moore neighborhood in cellular automata theory. The original definition of the Moore neighborhood is the cellular itself and the surrounding eight cells, as Figure 2 shows. The mathematical definition of the Moore neighborhood with a radius k is

$$\{c_{ij} | \|c_{ij} - c\|_{L_\infty} < k\}, \tag{2}$$

where c_{ij} is the cell on the i-th row and the j-th column ,c is the center of the Moore neighborhood and L_∞ is the Chebeshev distance.

Combined with the traditional representation of different scaling rate, we further develop a neighborhood-aware representation (NAR), which also takes the offensive direction of the team into consideration. As Figure 3 shows, assuming the team is attacking from left to right, the ranged 2 Moore neighborhood indicates 9 relative positions of the player 4. The player 1 (and 3), 2, 6, 7 and 8 are at the LB, LF, RB, Right and RB positions of the player 4 respectively. After extracting the NAR of all players, we form a 10×9 neighboring graph. As the formation depicts the relative position along the touchline of the players (the backwards will stay closer to the back-court than the forwards), the layout of neighboring graph should be consistent with this characteristic. An $x-$coordinate first sorting is operated on the position of the players and the neighboring graph is re-organized by the order, as Figure 3 shows. The NAR builds the relationship between the players indirectly, which is suitable for modeling the unstructured positional data for formation detection.

3.2 Temporal Segmentation and Detection

We explore a new method to segment and detect the formation given the representation of the player positions. Formation is a kind of spatio-temporal data, not only related to the relative position of the players on the pitch, but also concerning the evolution of their positions. According to the domain experts, when a team obtains the ball at the back court, the defensive formation will change into an advancing formation, trying to control the midfield and seeking a chance to pass the ball forward. If the team goes deep into the front court, the formation will become more aggressive, even disappear, because the strikers have much more freedom to choose their positions for tearing off the defense of the opponents and creating chaos in front of the goal. The boundary between two formations is consequently vague, with uncertainty in some way. Therefore, the model should be able to learn the temporal patterns of the features but should not divide the time into two segments when the position of the players changes slightly, like the text segmentation according to semantic differences instead of the variation in description.

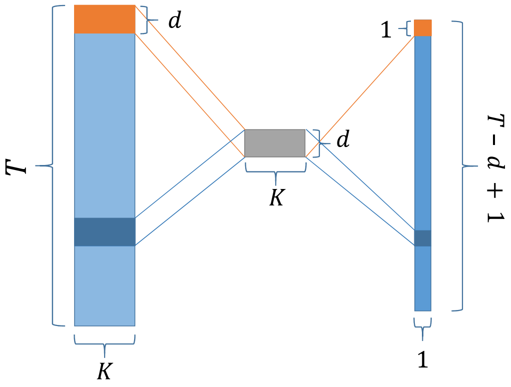


Figure 4: Temporal convolution: given an input sequence with length T and K features, convolved by a kernel with duration d, an output in shape $(T - d + 1) \times a$ is obtained (withoud padding).

The problem is similar to the human action segmentation and detection problem but is more challenging. On the one hand, the boundaries between human actions are clearer. On the other hand, many works on image feature extraction and physical detection have laid the foundation for the derived works. Referring to the general framework in this area, the feature sequences extracted from the videos will be fed into a set of classifiers. Traditionally, Recurrent Neural Networks (e.g. LSTMs and GRUs) achieve good performances on the sequential data, benefiting from their recurrent structure. Recently, [Bai *et al.*, 2018; Lea *et al.*, 2017] showed that temporal convolutional networks have the potential to compete with RNNs on sequential data.

Temporal Convolution

Temporal convolution is a kind of 1D convolution, which means that the filters only convolve temporally. As Figure 4 shows, after convolving by a $d \times l$ filter with stride one and zero padding, the input sequence in shape $t \times d$ is transformed into a $t \times 1$ sequential feature. Note that d is the dimension of input features and l is the perception area. The filter can capture the features within l units temporally.

In our case, given a sequence S_i, we have the representation $r_j^i \in \mathbb{R}^{10 \times 9}$. With the dot product nature of convolution, the representations can be flattened into vectors and the form becomes the same as example.

Application of ED-TCN

In our paper, we adopt the Encoder-Decoder temporal convolutional network (ED-TCN) by [Lea *et al.*, 2017] for segmentation and classification, whose architecture is as Figure 5 shows.

The encoder with n layers convolves and captures the temporal patterns of the representation. The decoder has the same number of layers, upsampling the high level embedding and classifying the formation. Every layer contains a set of temporal convolutional filters. The perception area of the model is $l(2^n - 1) + 1$. The model shows good performance on segmenting and classifying the human action using appearance features extracted from the videos.

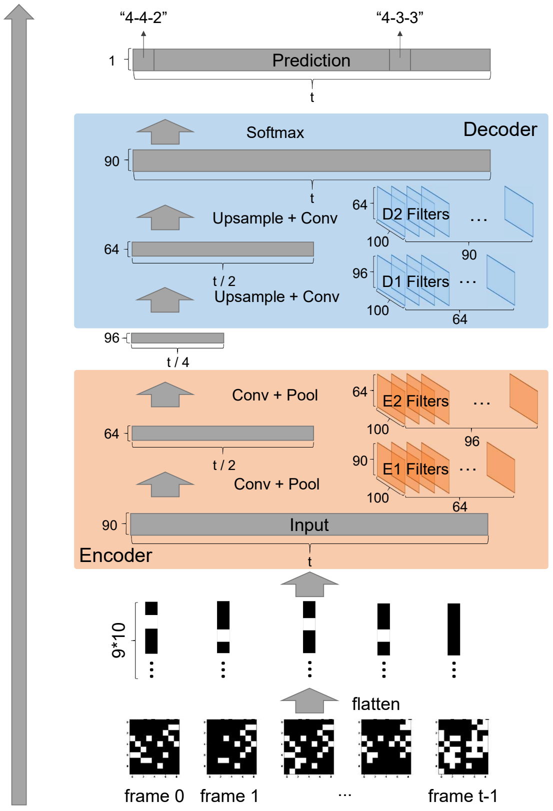


Figure 5: The architecture of the ED-TCN

3.3 Baseline Method

We adopt the model developed by [Bialkowski *et al.*, 2014b] as the baseline method. The model assumes that a team uses a stable formation within a period of time. Therefore, the roles on the pitch are stable and every moment each role is played by a player. A role is modeled with a 2D gaussian distribution. To obtain the final clustering, the model adopts a k-Means equivalence framework to compute the distributions. Iterating through the temporal data, player positions at each frame are assigned to the distributions bijectively using the Hungarian algorithm. After several iterations, the distributions come to convergence.

With a representation of distributions, the position of the players within a period is aggregated. The authors labeled the formation of the aggregations and performed agglomerative clustering on the them, with the metric of EMD distance. The final agglomerations are marked as specific formations respectively, and with the results, formation detection is feasible given an unlabeled positional aggregation of the team within a period. The Figure 6 shows the framework of the model. The baseline method is used for macro analysis of the formation. The authors get 6 agglomerations representing the formation 4-2-3-1, 4-4-2, 3-4-3, 4-3-3, 4-1-4-1 and others, without 2-3-2-3. The formation 2-3-2-3 is used by a team when it tries to increase the people in the frontcourt during the transition phase. The formations with relatively

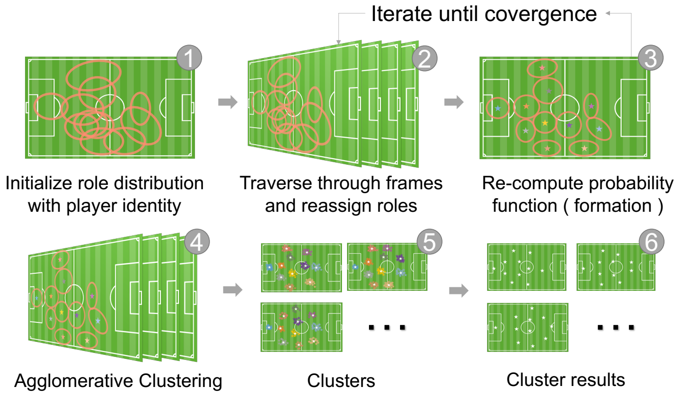

Iterate until covergence
1
2
3
Initialize role distribution with player identity
Traverse through frames and reassign roles
Re-compute probability function (formation)
4
5
6
Agglomerative Clustering
Clusters
Cluster results

Figure 6: The framework of the baseline method.

lower frequency would be elliminated. The clustered distributions are classified as the formations balanced in attack and defense. In the whole game, the formations of frontcourt offense and deep back court defense are averaged as a single balanced formation.

4 Experiments

In the experiments, we firstly evaluated the parameter selection for feature extractions, including the scaling rate, the range of the neighborhood, and the number of the filters on each layer. We then compared our model with the baseline method to demonstrate the advantages of our model.

4.1 Implementation

The ED-TCN was implemented with Keras [Chollet and others, 2015] by [Lea *et al.*, 2017]. Based on the provided code, we made some modifications to support our positional data. We trained the ED-TCN on an NVIDIA 1080Ti.

We implemented the baseline method by Python, with NumPy for matrix computations. In our experiments, the data was acquired from a U15 game between Argentina and Brazil. The sequential positional data was truncated by the events during the game and the number of the sequences is 294, the volume of which is comparable to the 50 Salads [Stein and McKenna, 2013]. To accelerate the training process, the sequences were downsampled temporally by 3. After downsampling, the longest sequence contains 892 frames, with a feature dimension k = 90 (9 neighbors * 10 players). Before being fed into the ED-TCN, each sequence has been zero padded with mask.

4.2 Parameter Selection

The first experiment evaluates two aspects of the parameter. The first aspect is about the feature extraction, which is correlated to scaling rates and the range of NAR. In our experiment, the scaling rate ranges from 2 to 10 and NAR ranges 1 and 2. The second aspect is the number of the filters. The domain experts indicate that a formation usually sustains for 20 $\sim$ 30 seconds and the players are required to transform from one formation to another in about 5 $\sim$ 8 seconds. Therefore we set the duration of the convolution kernel to 100, equivalent to 12 seconds. Referring the setup of the ED-TCN, we set two combinations of the filters of the layers, (64, 96) and (96, 128). Our exploration on deeper networks did not get good feedbacks. To evaluate the effect of the scaling, we defined a variable to measure the information density of the representation. For a representation $r \in \mathbb{R}^k$, information density is defined as:

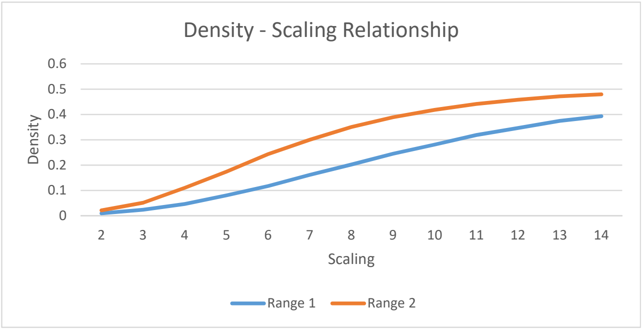
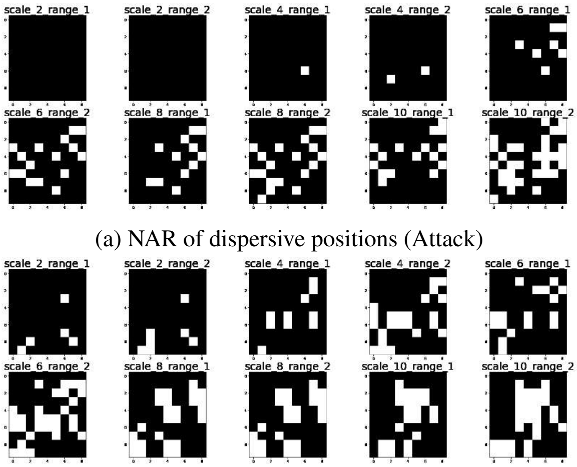


Figure 7: Information density versus scaling rate.



(a) NAR of dispersive positions (Attack)

(b) NAR of tight positions (Defense)

Figure 8: The NAR under different density of player positions.

$$d(r) = \frac{\sum_{i=0}^{k} r_i}{k \cdot Max(r)}, \tag{3}$$

where $Max(r)$ means the maximum value in representation r. We followed [Lea *et al.*, 2017] and employed the segmental $F1$ score as the metric to evaluate the model.

The results are shown in Table 1. The information density in the table is the average density of the NAR with relative sampling rate and range. As the sampling rate increases from 2, the information density increases, but as the sampling rate gets larger, the increase of the information density gets slower (Figure 7). From the table, we can find that the representations with higher scaling rates get better results. When the information density reaches 0.24, the accuracy increases obviously. From the table, we notice that when the information density is in $0.24 \sim 0.30$, the F1 scores are relatively higher.

Soccer is a highly dynamic sport, and the relative distance between the players change intensely. A proper scaling rate

Scaling	Range	Density	64,96	96,128
2	1	0.009	44.86(40.01)	41.85(38.07)
	2	0.021	50.41(53.33)	33.11(33.27)
3	1	0.023	48.06(45.07)	49.15(52.09)
	2	0.051	37.84(44.58)	39.16(31.47)
4	1	0.046	39.05(41.67)	39.69(37.44)
	2	0.110	54.9(52.36)	60.24(55.22)
5	1	0.080	42.48(42.72)	27.45(29.79)
	2	0.174	41.14(40.09)	34.45(34.86)
6	1	0.117	41.44(43.75)	55.15(55.21)
	2	0.243	52.07(51.53)	58.41(**60.69**)
7	1	0.161	59.51(57.55)	43.01(41.94)
	2	0.300	63.8(**57.6**)	50.32(49.31)
8	1	0.202	32.15(36.34)	38.54(38.19)
	2	0.350	48.63(47.92)	52.8(54.31)
9	1	0.245	66.8(**65.28**)	59.24(**61.25**)
	2	0.389	52.95(55.14)	49.0(48.96)
10	1	0.281	57.83(53.5)	53.08(50.28)
	2	0.418	46.05(47.38)	55.8(56.53)

Table 1: Accuracy (F1 score) of parameter selection. The table shows the accuracy and F1 score of the model with NAR with different scaling rates and ranges.

and an effective range are crucial to capture the relative position of the players. As Figure 8 shows, the NARs with a low scaling rate are unable to capture the relative position of the players when they get far away from each other, especially under the attack situation. However, it is not the fact that the larger the scaling rate, the larger the information density and the better the effect of the representation.

4.3 Comparison with The Baseline Method

We further compared our model with the baseline method. The baseline method is based on the stability assumption of formation. It generates only one representation of the formation every half game and the model evaluation was conducted with a full season data of the English Premier League. To compare the two models, we subdivided the sequential data into 330 smaller multiples, and each multiple with a label for the ease of clustering. To make sure the volume of data for the distribution representation is large enough, we did not perform downsampling on the data. The shortest sequence here is of 27 frames, about 1 second on the video. To obtain the prediction, all the data was used for clustering. Given the agglomerations, the result is shown in Figure 9 assuming that the corresponding relation was built by maximizing the total correctness. The vertical labels represent the ground true and the horizontal labels represent the result of agglomerative clustering. Cluster 0 was the agglomeration of formation 4-4-2, with an accuracy of 76%, but the accuracy of other agglomerations was lower than 50%. The accuracy of the baseline method on the test set is 29.79% and the $F1$ score is 23.47%, which is relatively smaller than the accuracy.

Discussion: The baseline method does not work for fine-grained formation detection in a dynamic situation. Though the sequences are subdivided into smaller multiples, the data is not clean enough for unsupervised learning. To eliminate the spatial variation, the baseline method normalizes the average position of raw data to the origin by a frame-wise mean subtraction. The temporal variation is modeled by 2D Gaussian distribution. Due to the human variation, the dominant formation of the labeled data within a period is correct, but abrupt changes exist. When a team loses the ball, the player nearby will snatch the ball on the spot. At the same time, the formation will start to transform into a defensive one, which takes time. The player may successfully get the ball back and the transformation will stop suddenly and resume to an offensive state. The variation is ignored by the annotators, which is unfriendly to the unsupervised clustering. The distribution representation is not robust to the variation of the sequence length as a longer sequence usually have a larger temporal variation. In our experiment, the largest standard variance of the distribution reaches 179.5, but the smallest one is just 0.23, corresponding to the shortest sequence. The baseline method can achieve good performance on the formation for the whole game. However, for fine-grained analysis, it is difficult for the baseline method to obtain long positional sequences with consistent formation labels in soccer games.

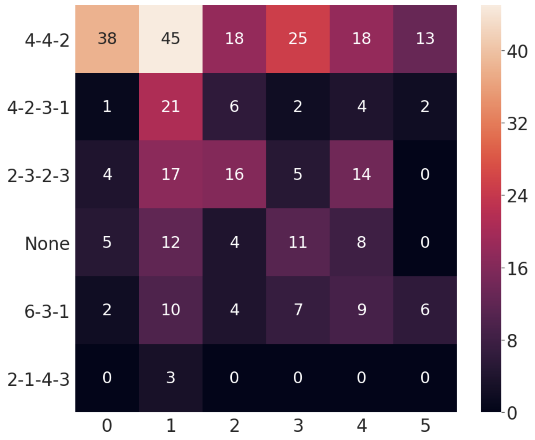


Figure 9: Confusion Matrix: The detection result of the baseline method. The horizontal labels show the clusters and the vertical labels show the most possibel classes. The accuracy of detection is 0.76, 0.19, 0.33, 0.22, 0.17, and 0.

5 Conclusion

This work studies a challenging fine-grained formation detection problem in soccer analysis. To solve the problem, we transform it to a sequence segmentation and detection problem. With this novel transformation, a deep learning approach can be effectively applied on the problem. We then propose a neighborhood-aware representation (NAR) to characterize the topological structure of the players. The experiment on parameter selection in Section 4.2 shows that the NAR can capture the features of player positions with a proper level of information density, which is around $0.24 \sim 0.30$. A new framework based on temporal convolutional network is proposed to segment a soccer game into a set of periods with specific formations. The comparative evaluation in Section 4.3 demonstrates the significant advantages of our method over the baseline method.

References

[Bai *et al.*, 2018] Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *CoRR*, abs/1803.01271, 2018.

[Bialkowski *et al.*, 2014a] Alina Bialkowski, Patrick Lucey, Peter Carr, Yisong Yue, and Iain Matthews. Win at home and draw away: Automatic formation analysis highlighting the differences in home and away team behaviors. In *Proceedings of 8th annual MIT sloan sports analytics conference*, pages 1–7. Citeseer, 2014.

[Bialkowski *et al.*, 2014b] Alina Bialkowski, Patrick Lucey, Peter Carr, Yisong Yue, Sridha Sridharan, and Iain Matthews. Large-scale analysis of soccer matches using spatiotemporal tracking data. In *2014 IEEE International Conference on Data Mining (ICDM)*, pages 725–730. IEEE, 2014.

[Chollet and others, 2015] François Chollet et al. Keras. https://keras.io, 2015.

[Dearden *et al.*, 2006] Anthony Dearden, Yiannis Demiris, and Oliver Grau. Tracking football player movement from a single moving camera using particle filters. 2006.

[Decroos *et al.*, 2018] Tom Decroos, Jan Van Haaren, and Jesse Davis. Automatic discovery of tactics in spatio-temporal soccer match data. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery on Data Mining*, pages 223–232. ACM, 2018.

[Gehring *et al.*, 2017] Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. Convolutional sequence to sequence learning. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1243–1252. JMLR. org, 2017.

[Gonçalves *et al.*, 2017] Bruno Gonçalves, Diogo Coutinho, Sara Santos, Carlos Lago-Penas, Sergio Jiménez, and Jaime Sampaio. Exploring team passing networks and player movement dynamics in youth association football. *PloS one*, 12(1):e0171156, 2017.

[Kalchbrenner *et al.*, 2016] Nal Kalchbrenner, Lasse Espeholt, Karen Simonyan, Aäron van den Oord, Alex Graves, and Koray Kavukcuoglu. Neural machine translation in linear time. *CoRR*, abs/1610.10099, 2016.

[Lea *et al.*, 2017] Colin Lea, Michael D Flynn, Rene Vidal, Austin Reiter, and Gregory D Hager. Temporal convolutional networks for action segmentation and detection. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 156–165, 2017.

[Stein and McKenna, 2013] Sebastian Stein and Stephen J McKenna. Combining embedded accelerometers with computer vision for recognizing food preparation activities. In *Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computing*, pages 729–738. ACM, 2013.

[Wang *et al.*, 2015] Qing Wang, Hengshu Zhu, Wei Hu, Zhiyong Shen, and Yuan Yao. Discerning tactical patterns for professional soccer teams: An enhanced topic model with applications. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 2197–2206. ACM, 2015.

[Wei *et al.*, 2013] Xinyu Wei, Long Sha, Patrick Lucey, Stuart Morgan, and Sridha Sridharan. Large-scale analysis of formations in soccer. In *Digital Image Computing: Techniques and Applications (DICTA), 2013 International Conference on*, pages 1–8. IEEE, 2013.

[Wu *et al.*, 2019] Yingcai Wu, Xiao Xie, Jiachen Wang, Dazhen Deng, Hongye Liang, Hui Zhang, Shoubin Cheng, and Wei Chen. Forvizor: Visualizing spatio-temporal team formations in soccer. *IEEE transactions on visualization and computer graphics*, 25(1):65–75, 2019.