Correcting the trajectory of the player in Soccer game

Kai Seno s1250240 Supervised by Prof.Maxim Mozgovoy

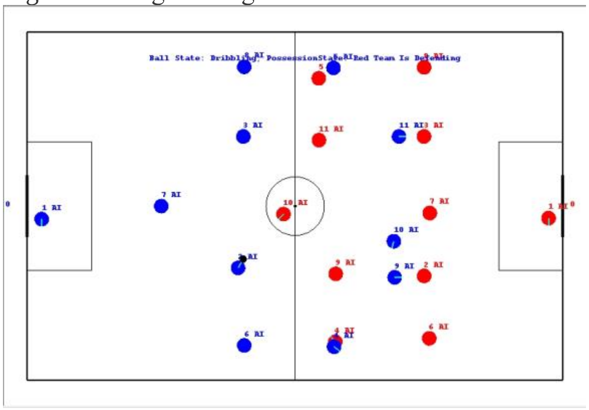
Abstract

In the soccer game engine I'm studying [1], player movements are simulated based on coordinate information (Fig.1). This is information obtained from observing the actual game. However, the data obtained at this time is not perfect, and there is a problem that the coordinate information obtained from the tracking of the player is not accurate.

My research is to improve existing coordinate information, including jitter, and provide coordinate information for players that move more smoothly. This time I use a filter called the least squares method [2] to measure the improvement of coordinate information. This is a method to derive the function with the least deviation from multiple coordinates based on the coordinates. This removes the coordinates that are obviously jitter and optimizes the data.

It also evaluates whether the modified coordinates are good or bad. In order to evaluate the effect of the above filtering, we will examine and evaluate the "similarity" with the original data and the "smoothness" of the modified coordinates. This evaluation determines what is optimal for the order of the function applied by the least squares method, and confirms its reliability as a filter.

Fig.1. Soccer game engine “TSG simulator”



1.Motivation

In soccer games, it is common for teams to collect and use match data to win. Not only the number of shots and the control rate, but also the mileage and trajectory of the athlete are used as data. In other words, match data is very important in soccer in recent years, and the accuracy of this data will have a great impact on the strategy of the team. But unfortunately, the match data is not perfect. My research is to correct this inaccurate data and provide more valuable data.

The modern method of observing trajectories is to bring an observation aircraft to the site and automatically track the movements of the ball, players, and referees in real time to acquire data. Regarding auto-tracking, first, two camera units (each unit is equipped with his three cameras) are installed at a high position in each stadium to ensure a view of the entire pitch (Fig.2). However, this observation is not perfect, and tracking becomes difficult when players cross each other or when many players are crowded. In addition, the observation aircraft may misidentify due to the movement of the player's hand. Inaccurate information (jitter), as I say, is a record of coordinates that are slightly off from the original position for that reason. This appears in the simulation as if the player's position suddenly jumps when the player is running. I prepare a filter to correct such player’s jitters.

Fig.2. Image of observer

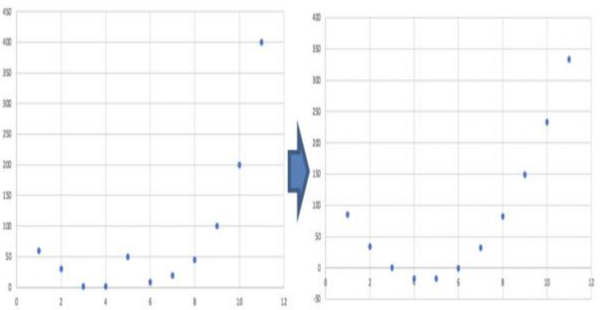
2.Method

2.1 Data correction process

In the first place, the least squares method is the sum of the residual sums so that the function assumed when approximating the set of numerical values obtained by the measurement using the function assumed from an appropriate model is a good approximation to the measured value. This is a method of determining a coefficient that minimizes the sum of squares. When a locus as shown in the figure below (Fig.3) exists as data, there are coordinates that clearly deviate from the locus. By applying this data to a quadratic function using the least squares method, we have succeeded in eliminating the outlier coordinates. This time, we will correct the data in this way.

The data was collected by approximating with a 2nd to 6th order function without approximating with a logarithmic curve.

Fig.3. Example of Least squares method



2.2 Evaluation of the filter

There are two factors that were used to evaluate the filter this time: "smoothness" and "similarity" [3].

"Smoothness" is the value of the amount of change in the slope of the function obtained by the filter. The difference in slope before and after when the value on the x-axis changed was calculated and used as the sum of the absolute values of the values. That is, the closer this value is to 0, the smoother it is and the better the evaluation is. This evaluation method is used because it is considered that the data in which jitter is present loses smoothness due to jitter. Expressed as an equation, assuming that smoothness is calculated with n data for the modified coordinates y, the following formula is used. (n is number of coordinate)

$$Smoothness = \sum_{i}^{n} \left(\frac{\mathrm{dy_i}}{\mathrm{dx}} - \frac{\mathrm{dy_{i-1}}}{\mathrm{dx}}\right)$$

"Similarity" is the value of the difference between the y-axis value of the function obtained by the filter and the y-axis value that can be placed in the original data. The sum of the absolute values of the differences between the above two points in a certain data was calculated and used as the similarity. That is, the closer this value is to 0, the similarity is and the better the evaluation is. Even if the filter succeeds in removing the jitter, if the difference from the original data is large, it will be different data, so this evaluation method is used. Expressed as an equation, assuming that the evaluation is calculated for n data with the y coordinate of the original data as y and the modified coordinate y as Y, the following formula is used. (n is number of coordinate)

$$Similarity = \sum_{i}^{n} (y_i - Y_i)$$

3.Result

3.1 Result of filtering

The data used is partially modified by a filter. The data used is a part of the data 1 in the evaluation section. Modify with a quadratic function and a cubic function, and compare before and after modification.

The coordinates are coordinated as shown in the figure (Fig. 4) With the center of the field as the origin, there are coordinates in three directions, x, y, and z. This time, we are focusing on the x and y coordinates, not the z coordinate.

Fig.4. Field coordinates

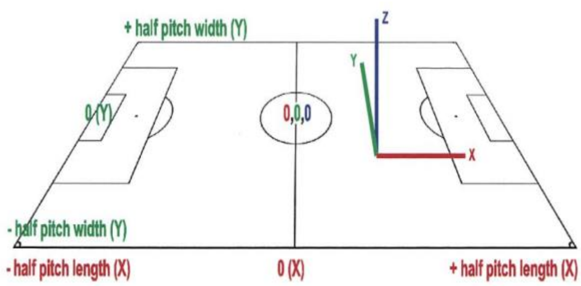


Fig.5-1. Correct the trajectory with quadratic function

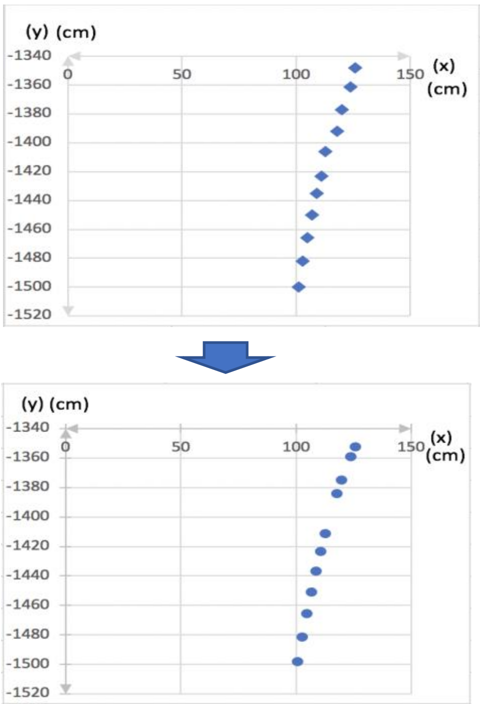
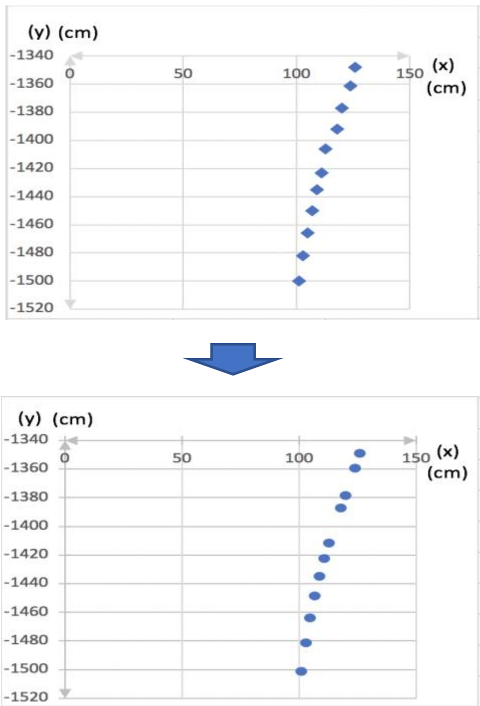


Fig.5-2. Correct the trajectory with cublic function



3.2 Result of evaluation

The following is an evaluation of player trajectories randomly extracted from the data. This locus is a part of the locus that the player has moved during the match, and contains about 50 coordinates. Prepare seven of these and use them as data 1-7.
The results corrected by the 2nd to 6th order functions by the filter are shown below. As explained earlier, the smaller the size, the better.

The result of the data is rounded to the first decimal place. Also, since the value of the 6th-order function data is very large, it is truncated to a small number.

●Table: Evaluation value in each data

Data1		
degree	Smoothness	Similarity
2	5.4	27.2
3	6.9	19.9
4	9.0	15.3
5	319.8	192.2
6	52276803048	41658389880

Data2		
degree	Smoothness	Similarity
2	0.3	124.8
3	1.3	76.3
4	3392.6	364.6
5	638.8	24476.1
6	11922011838	47766403626

Data3		
degree	Smoothness	Similarity
2	0.9	122.7
3	0.7	61.5
4	0.8	60.7
5	3.1	60.9
6	7027512	5440712

Data4		
degree	Smoothness	Similarity
2	1.2	232.3
3	2.7	136.9
4	2870.3	10362.1
5	40723.3	65317.2
6	786789065	71186555777

Data5		
degree	Smoothness	Similarity
2	0.9	190.7
3	1.4	194.1
4	2529.7	22479.5
5	2530.0	8492.0
6	5507183056	44236697630

Data6		
degree	Smoothness	Similarity
2	2.5	334.4
3	4.1	148.4
4	128.9	109.5
5	412.1	109.3
6	38104440	559958300

Data7		
degree	Smoothness	Similarity
2	0.2	323.9
3	2.4	323.3
4	532.4	15922.0
5	10415.0	118532.6
6	6225542446	77316963760

4.Analysis

Looking at the result of the correction by the filter, the point that seemed to be jitter due to the original data was corrected. However, the result of the quadratic function is very different from the original data, and the filter of the cubic function seems to be better. Quadratic filters are excellent in terms of smoothness, but lack practicality in terms of similarity. It seemed that the higher the order, the better the correction by the filter. However, looking at the results of the evaluation, it can be seen that the values larger than the sixth-order function become very large and do not function as a filter. Up to the quintic function, it depends on the data, but there are some that give not bad numerical values as evaluation.

A possible reason is that high-order filters give unnecessary curves. A high-order filter would make unnecessary corrections to the data that would have been optimal if the order was 3 in the original trajectory. As a result, the similarity value is expected to be worse. Also, regarding smoothness, this value will increase as the number of unnecessary curves increases. The expectation that a high order would give a flexible modification was wrong, and it was found that a degree of 3 gives the best rating.

Reference

[1] M.Buckland , Programming Game AI By Example (Wardware Game Developers Library),2004

[2] Keniti Kanatani, “Renormalization for Computer Vision” Information Processing Society of Japan Journal, vol.35 No.2, Feb 1994 (Japanese thesis)

[3] Zhilong XU, Masakazu Higuchi and Takashi Komuro ,“Comparison of Real-time Smoothing Filters for Spatial Operating Interfaces” ITE Winter Annual Convent on 2013 (Japanese thesis)