Computers and Electronics in Agriculture 186 (2021) 106180

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

journal homepage: www.elsevier.com/locate/compag

Original papers

Field-road trajectory segmentation for agricultural machinery based on direction distribution

Ying Chen, Xiaoqiang Zhang, Caicong Wu *, Guangyuan Li

College of Information and Electrical Engineering, China Agricultural University, Beijing 100083, China

ARTICLE INFO

Keywords:
Trajectory segmentation
Field-road segmentation
Rule-based inference
DBSCAN
Operational management

ABSTRACT

Field-road segmentation that automatically divides a trajectory into a sequence of field/road segments is an important component in segmentation process for the trajectories of agricultural machinery. A trajectory is a sequence of geospatial coordinates recorded by GNSS receivers during the driving of the machine. The objective of this paper is to develop a field-road segmentation method in the case of the unavailability of field boundary information. The developed method consists of two stages. The first stage uses DBSCAN, a typical clustering algorithm, to do field-road segmentation, and the second stage uses a rule-based inference to correct two types of false segmentation cases from output of DBSCAN-based clustering. Based on the parallel direction distribution that strips in the same field are almost parallel, two inference rules, *Field2Road-Cluster* and *Road2Field-Segment* are performed sequentially. *Field2Road-Cluster* uses the direction distribution difference (parallel in fields vs. not parallel on roads) to correct false field segmentation cases and *Road2Field-Segment* uses the parallel relationship among strips in the same field to correct false road segmentation cases. The developed method was validated by 60 selected trajectories. The results demonstrated that the rule-based inference achieved an increase of 7.95% in F1 scores, where *Field2Road-Cluster* and *Road2Field-Segment* contributed 6.40% and 1.55% increase, respectively.

1. Introduction

When analyzing agricultural machinery operational performance, it is often a case that a trajectory segmentation system, which splits a GNSS (Global Navigation Satellite System) trajectory of the considered machine into a sequence of segments, is carried out. A segment refers to a sub-trajectory of a trajectory where the same operation is performed (e.g., driving on the road, working in fields and turning at headland). Furthermore, trajectory segmentation is often performed based on an operation scheme (e.g., the KTBL time classification scheme, KTBL, 2016). For example, to support field efficiency assessment, two types of segmentation were sequentially carried out (Kortenbruck et al., 2017): field-road segmentation that splits a trajectory into a sequence of field/road segments, and working-turning segmentation that splits a field segment into a sequence of working/turning segments. Compared to a significant amount of research on the working-turning segmentation of a field segment (Grisso et al., 2002; Grisso et al., 2004; Bochtis et al., 2010; Jensen and Bochtis, 2013; Griffel et al., 2020), few studies have been carried out to investigate the field-road trajectory segmentation. Most of existing field-road segmentation methods used field boundary information to separate field and road activities (Kortenbruck et al., 2017; Hanke et al., 2018). However, the information of field boundary is not always available in many countries and areas (Kilic et al., 2017), and the boundary of the considered field cannot be created totally automatically (Kortenbruck et al., 2017; Hanke et al., 2018). Therefore, the field-road segmentation without using field boundary information is required as a new functionality of the developed trajectory segmentation systems.

The required field-road segmentation can be considered as a case of the stop-move segmentation problem (Spaccapietra et al., 2008), a typical research issue in human urban transportation, which automatically splits a trajectory into a sequence of stop/move segments. Although the definitions of "stop" and "move" depend on targeted applications, they have similar characterizations: low-speed for "stop" (e.g., shopping) and high-speed for "move" (e.g., road transportation). Analogously, in our field-road segmentation method, "field" and "road" corresponds to low-speed and high-speed respectively. In general, a stop-move segmentation method consists of two stages: clustering and inference. In the clustering stage, a density-based clustering algorithm is chosen due to low speed of "stop" resulting in high point density. The

* Corresponding author.
E-mail address: wucc@cau.edu.cn (C. Wu).

https://doi.org/10.1016/j.compag.2021.106180
Received 23 November 2020; Received in revised form 3 March 2021; Accepted 20 April 2021
Available online 5 May 2021

point density of a point refers to the number of points in its neighborhood. For a trajectory, the points are classified into clusters ("stop" points) and noise points ("move" points). According to the ways to compute point density, two types of density-based clustering algorithms are used for stop-move segmentation: spatial-closeness-based (e.g., DBSCAN, Ester et al., 1996; OPTICS, Ankerst et al., 1999) and spatio-temporal-closeness-based (e.g., CB-SMoT, Tietbohl et al., 2008; TrajDBSCAN, Tran et al., 2011; SOC, Xiang et al., 2016). In the inference stage, various rule-based inference algorithms have been proposed, which use application-specific information to correct false segmentation cases in the output from the clustering stage. E.g., CB-SMoT used information of Places of Interest (POIs) (Tietbohl et al., 2008), and SOC used the measurement information of straightness and centered-distance (Xiang et al., 2016). Since most existing approaches to the stop-move segmentation problem target the applications in human urban transportation, their rule-based inference algorithms cannot be directly applied to the field-road segmentation.

The objective of this paper is to develop a field-road segmentation method that can automatically segment a trajectory of agricultural machinery into a sequence of field/road segments. In the clustering stage, a typical density-based clustering algorithm (DBSCAN) is chosen, and then in the inference stage, based on the parallel direction distribution of strips in a field, a rule-based inference algorithm is developed. Specifically, the rule-based inference attempts to correct the following two types of false segmentation cases in the output of DBSCAN-based clustering.

- False field points: It is often a case that a machine drives slowly or even stops on the road (e.g., passing a traffic light, making a turn). Since the point density of such a "road" segment is similar to the one of a "field" segment, the points in the "road" segment are considered as "field" by mistake during DBSCAN-based clustering (namely false field points, as shown in Fig. 1a).
- False road points: Due to GNSS signal loss, points in a "field" segment sometimes are sparse, and thus, they are treated as "road" by mistake during DBSCAN-based clustering (namely false road points, as shown in Fig. 1b).

2. Materials and methods

2.1. Overview

Straight trips pattern is one of the most commonly used fieldwork patterns in China (Miao and Wang, 2011; Xiang and Noguchi, 2014). Compare with a *road* segment, a *field* segment has the following two characteristics: i) high point density because both in-field driving speed is rather low and the distance between consecutive strips is closed (as shown in Fig. 2a); ii) parallel direction distribution because consecutive strips are almost in opposite direction (as shown in Fig. 2b).

Based on these two characteristics, a field-road segmentation method was developed to distinguish *road* segments and *field* segments of a trajectory. The method involves the following three stages, as shown in Fig. 3.

1) Data cleaning. Smoothing of noise points and filtering of duplicated points are performed to provide a high-quality trajectory for the field-road segmentation (presented in Section 2.2).
2) DBSCAN-based clustering. DBSCAN, which is based on the high point density in fields, is used to segment a trajectory (presented in Section 2.3).
3) Direction-distribution-based inference. Two inference rules, *Field2-Road-Cluster* and *Road2Field-Segment*, are sequentially used to correct false segmentation cases in the result of DBSCAN-based clustering. Moreover, the two rules are based on the parallel direction distribution of strips in a field (presented in Section 2.4).

2.2. Data acquisition and cleaning

To evaluate the effectivity of the field-road segmentation method, a number of 60 daily trajectories are used as our experimental data, which are spatially distributed across nine provinces of China, from north regions to south regions (as shown in Fig. 4). These trajectories were collected using agricultural equipment from Apr. 2019 to Nov. 2019 in China by BO CHUANG LIAN DONG Inc and the equipment consisted of tractors mounted with low-accuracy (approximately 2 m–5 m) and low-cost (approximately $40) GNSS receivers. Each GNSS record includes tractor ID, timestamp (YYYY:MM:DD-hh:mm:ss), geospatial coordinate (including longitude and latitude, WGS84), speed (m/s), and direction (°). To validate the field-road segmentation method, these trajectories were annotated manually as ground truth data. During human annotation, a trajectory is divided into a sequence of segments, and the operation of each segment is given as either "*field*" or "*road*".

For each trajectory in the data, two types of data cleaning processes are performed as follows.

- Smoothing of noise points: GNSS signal noise is inevitable during the collecting of GNSS trajectories. To alleviate the impact of the signal noise, noise points are detected by using the maximum speed of the considered machine (Marketos et al., 2008). Then, for every noise point, the nearest-neighbor smoothing method is used, which replaces its geospatial coordinate by the mean of the geospatial coordinates of its adjacent points (Schuessler and Axhausen, 2009; Zheng, 2015).

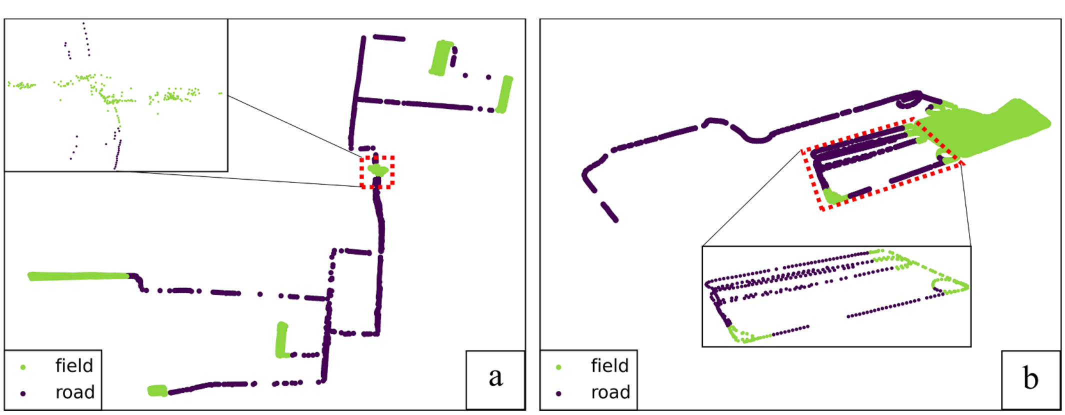

Fig. 1. Two types of false segmentation cases in the output of DBSCAN-based clustering. (a) False field points: the points in the dashed red rectangle (green color) are actually "road". (b) False road points: the points in the dashed red rectangle (brown color) are actually "field". (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

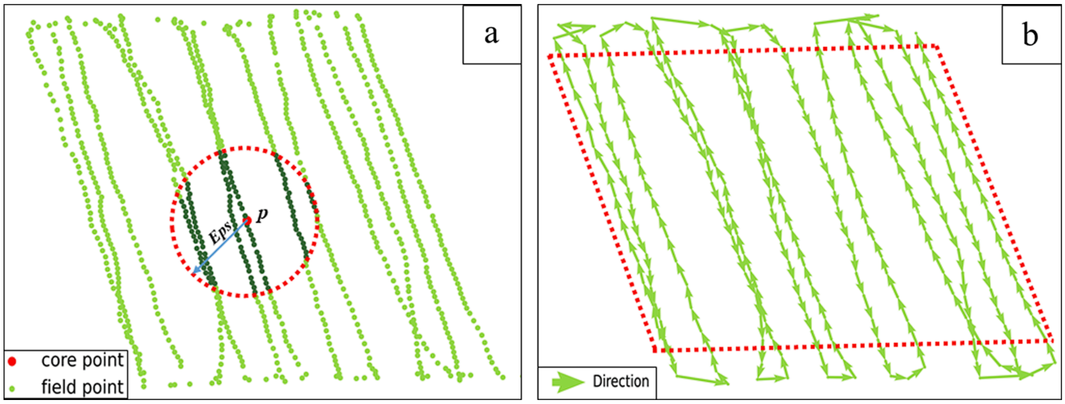

Fig. 2. The two characteristics of a field segment. (a) High point density: the point density of the red point p is high. Point p is a core point because the number of points in its *Eps-neighborhood* (i.e., the dark green points in the dashed red circle) is high; (b) Parallel direction distribution: the strips in the dashed red rectangle are generally parallel. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

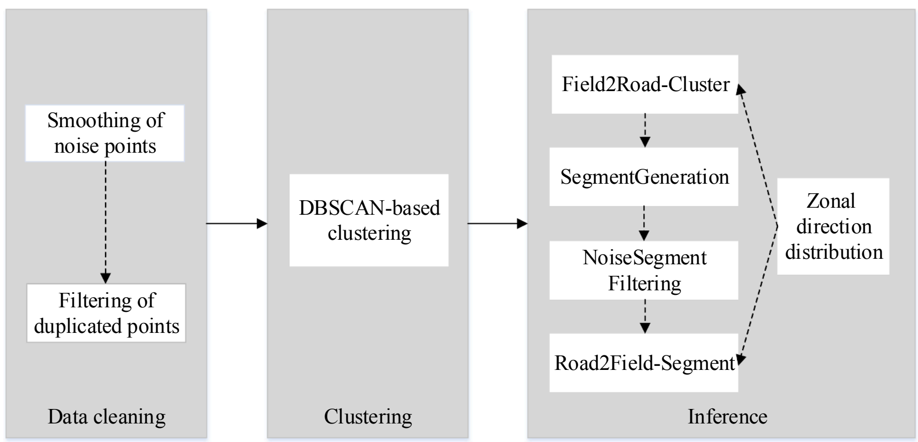

Fig. 3. The workflow of the field-road segmentation method.

- Filtering of duplicated points: Agricultural machines often perform a stop operation (e.g., lifting/lower of implements during turns at the headland, stops for traffic lights). Such a stop operation generates duplicated points that have the same geospatial coordinate and results in an abnormal high point density of the geospatial coordinate. Thus, for consecutive duplicated points, only the first one has remained, and others are deleted.

2.3. DBSCAN-based clustering

DBSCAN (Ester et al., 1996), which is one of the most commonly used density-based clustering algorithms, is chosen to do the field-road segmentation. The key idea of DBSCAN is that a cluster mainly comprises core points whose point densities are high (see Definition 1-2). E.g., as shown in Fig. 2a, the red point p is a core point and the dashed red circle is its *Eps-neighborhood*. For DBSCAN, the two input parameters must be specified: the neighborhood radius *Eps* and the minimum number of points in the neighborhood *MinPts*. Moreover, the appropriate values of the two input parameters should reflect the difference in point density: high in fields vs. low on roads.

Definition 1. (*Eps-neighborhood of a point*) : The Eps-neighborhood of point p, denoted by $N_{Eps}(p)$, is defined by $N_{Eps}(p) = \{q \in T | dist(p,q) \leq Eps\}$, where T is a set of points in the given trajectory, and $dist(p,q)$ represents the Euclidean distance between points p and q.

Definition 2. (*Core point*) : If the number of points in the *Eps-neighborhood* of point p is larger than *Minpts*, it is a core point.

DBSCAN accomplishes the clustering process by extracting clusters sequentially. Starting from an arbitrary point p, its *Eps-neighborhood* is checked. If p is a core point, a new cluster C is created with the points in $N_{Eps}(p)$. Then, for each point q in C whose *Eps-neighborhood* has not yet been checked, if q is a core point, the points in $N_{Eps}(q)$ which are not already contained in C are added to the cluster. The expansion of cluster C is repeated until no new point can be added to the cluster. The clustering process terminates when no new cluster is created. After clustering, the points in a cluster (namely a field cluster) are predicted as "*field*", and the noise points are predicted as "*road*". E.g., as shown in Fig. 5b, there are six field clusters.

2.4. Direction-distribution-based inference

A direction-distribution-based inference is developed to correct two types of false segmentation cases in the result of DBSCAN-based clustering: false field points and false road points. The key idea of inference is that strips in the same field are almost parallel. Based on the idea, two inference rules (*Field2Road-Cluster* and *Road2Field-Segment*) are developed from the perspective of clusters and segments, respectively.

As shown in Fig. 3, there are four steps in the direction-distribution-based inference. Firstly, rule *Field2Road-Cluster* (Section 2.4.2) corrects false field points in a cluster. Secondly, *SegmentGeneration* generates a list of segments according to the output of *Field2Road-Cluster*, where consecutive points with the same operations in the considered trajectory are linked into a segment. Thirdly, *NoiseSegmentFiltering* filters out noise

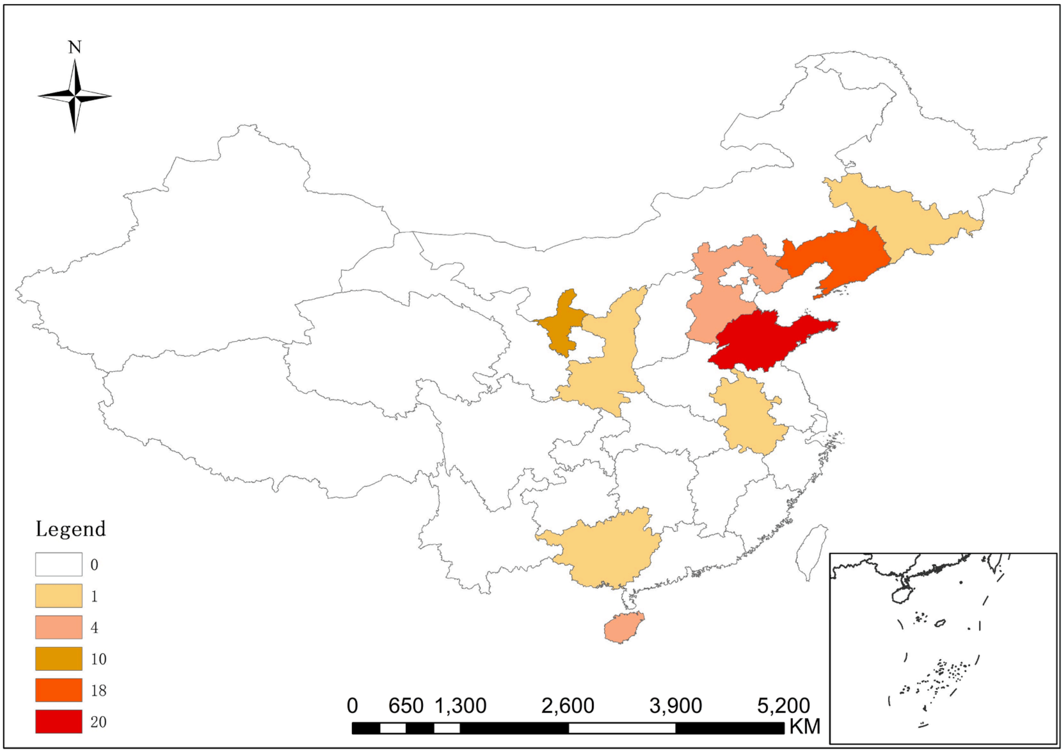

Fig. 4. The regional distribution of the trajectories in the data. Legend: the number of trajectories located in the considered province.

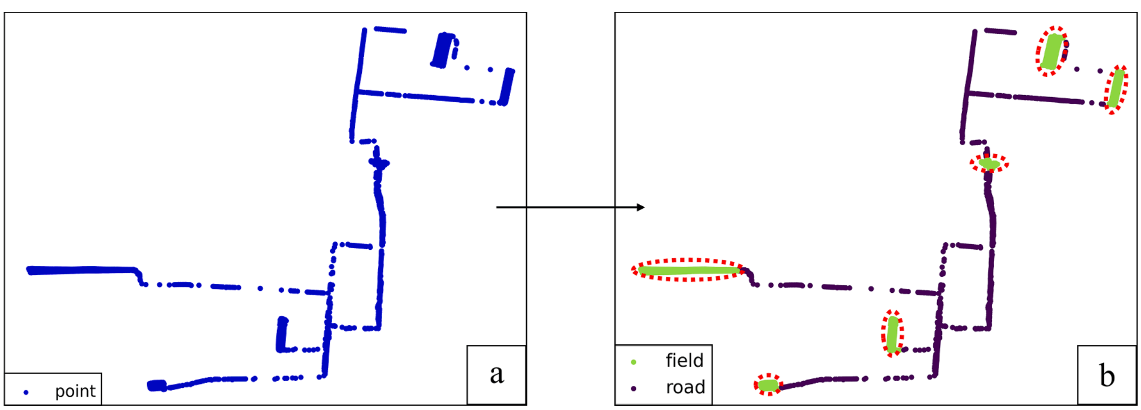

Fig. 5. The examples of DBSCAN-based clustering. (a) Input trajectory: the blue points are points in a trajectory; (b) Output trajectory: the green points are "*field*", the brown points are "*road*", and the dashed red circles are field clusters. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

segments. If the number of points in a segment is less than a threshold, it is considered as a noise segment. Both *SegmentGeneration* and *Noise-SegmentFiltering* attempt to provide high-quality segments. Lastly, rule *Road2Field-Segment* (Section 2.4.3) corrects false road points in a segment. Moreover, during *Field2Road-Cluster* and *Road2Field-Segment*, computation of zonal direction distribution (Section 2.4.1) is performed, which provides the direction distribution of a cluster or a segment.

2.4.1. *Computation of zonal direction distribution*

Given a set of points (a cluster or a segment), to alleviate the impact of GNSS signal noise, a zonal direction distribution is chosen to represent its overall direction distribution. For each point, according to its direction, a direction zone is selected from 36 direction zones (i.e., [0,10], [10,20]…[10,360]). For example, the zone of direction 6° is [0, 10]. Then, each direction zone has attached to a value of *PointNum*, which counts the number of points located in the zone. Lastly, 36 direction zones are ordered according to their *PointNum* values. For example, Fig. 6(a-b) illustrates the zonal direction distributions of two clusters, respectively.

2.4.2. *Field2Road-Cluster*

After DBSCAN-based clustering, if the points in a field cluster are actually "*road*", the cluster is a false one. E.g., the dashed red cycle in Fig. 7a is a false field cluster because the considered machine is on the road at that time. Although a true field cluster and a false field cluster are similar in their point densities, they are different in their direction distributions (parallel in a true one vs. not parallel in a false one). E.g., Fig. 6(a-b) shows the zonal direction distributions of a true field cluster and a false field cluster, respectively. Taking the advantage of the difference in direction distribution, rule *Field2Road-Cluster* is performed as follows. For each field cluster, the *PointNum* value of each direction zone of the cluster is examined. If every *PointNum* value is less than a

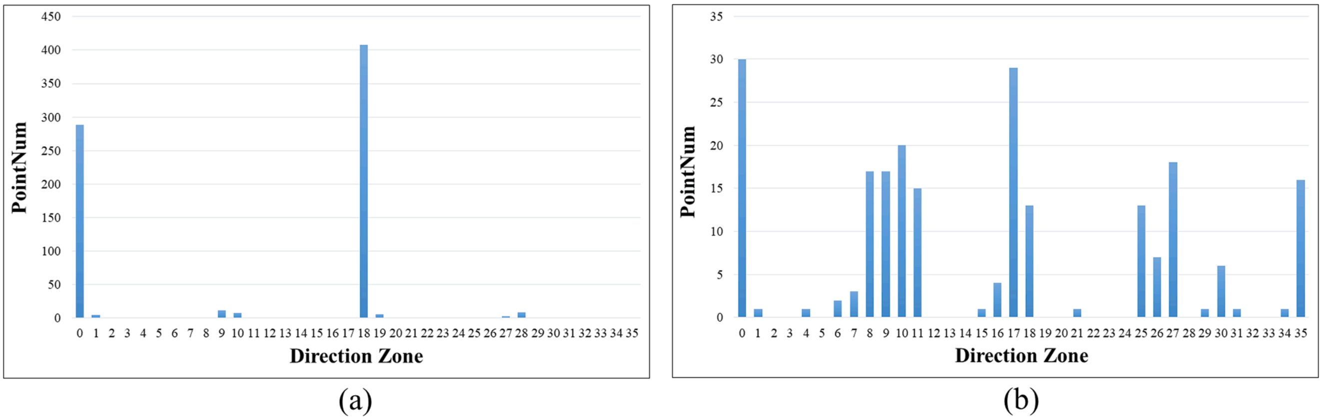

Fig. 6. The examples of zonal direction distribution. (a) A true field cluster: points mainly located in two parallel zones (i.e., [0,10] and [180,190]) and the *PointNum* value of either zone is high (above 250); (b) A false field cluster: points spread out across several zones and the *PointNum* value of each zone is low (below 30).

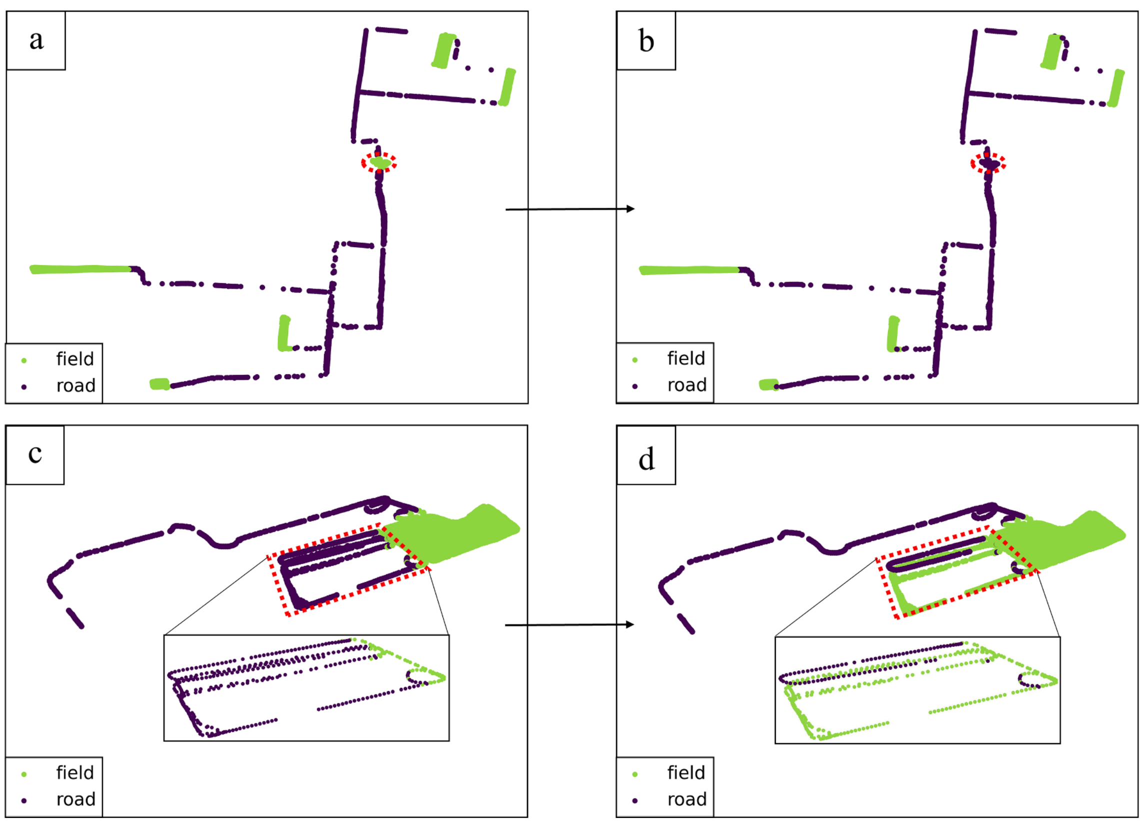

Fig. 7. The examples of the two inference rules. (a-b) Input and output trajectory of Field2Road-Cluster: the points in the dashed red cycle are changed from "*field*" (green color) to "*road*" (brown color); (c-d) Input and output trajectory of Road2Field-Segment: the points in the dashed red rectangle are changed from "*road*" (brown color) to "*field*" (green color). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

threshold (e.g., Fig. 6b), the field cluster is predicted as a false one. Then, the points in the cluster are changed from "*field*" to "*road*" (e.g., the points in the dashed red cycle from Fig. 7a to Fig. 7b).

2.4.3. Road2Field-Segment

Because of the problem of GNSS signal loss (e.g., a wet weather day), points in a field maybe be sparse, and are predicted as "*road*" by DBSCAN-based clustering. E.g., as shown in Fig. 7c, the road segments in the dashed red rectangle are actually in a field, and thus they are false ones. Moreover, it is obvious that these false road segments are parallel. Taking the advantage of the parallel relationship among false road segments which locate in the same field, rule *Road2Field-Segment* is performed as follows. Each *RRR* case is checked, where a *RRR* case comprises three consecutive road segments (seg_{k-1}, seg_k, seg_{k+1}). If the following two conditions are satisfied, the points in the three road segments are changed from "*road*" to "*field*" (e.g., the points in the dashed red rectangle from Fig. 7c to Fig. 7d).

Condition (1) (*Speed*): If the three segments (seg_{k-1}, seg_k, seg_{k+1}) are in the same field, their speed should be close to each other. Thus, both $\mathrm{speedDiff}(seg_{k(k-1)})$ and $\mathrm{speedDiff}(seg_{k(k+1)})$ (see Eq. (2)) need to be less than a threshold.

$$\mathrm{speed}(seg_j) = \frac{\sum_{p_i \in seg_j} speed(p_i)}{|seg_j|} \tag{1}$$

$$\text{speedDiff}(seg_{ij}) = |\text{speed}(seg_i) - \text{speed}(seg_j)| \quad (2)$$

where$speed(p_i)$ is the speed of point p_i, $|seg_j|$ is the number of points in segment seg_j ($j = k\text{-}1, k, k+1$), and $\text{speedDiff}(seg_{ij})$is the speed difference between two segments: seg_i and seg_j ($i, j = k\text{-}1, k, k+1$).

Condition (2) (*Direction*): If the three segments (seg_{k-1}, seg_k, seg_{k+1}) are in the same field, their direction should generally be parallel. Thus, the number of *Shared Parallel Zone* (see Definition 3) between segment seg_k and either of the two segments (seg_{k-1} and seg_{k+1}) should be greater than a threshold. In this paper, the value of parameter *n* in Definition 3 is chosen 3, because the top three zones usually can represent the overall direction of a segment.

Definition 3. (*Shared Parallel Zone*) : For a zone in the top *n* zone of segment seg_i, if either the zone itself (e.g., [30,40]) or its opposite zone (e.g., [210,220]) is in the top n zones of segment seg_j, it is a shared parallel zone between seg_i and seg_j.

3. Results and discussion

3.1. Performance metrics

To evaluate the effectivity of the developed method, three time-based performance metrics (precision, recall and F1 score) are used, which combine three metrics commonly used for the stop-move segmentation (Prelipcean et al., 2016) and a typical time-based metric used for field efficiency (Hunt, 2001). In a trajectory $P = \{p_1, p_2 \cdots p_n\}$, there are n points $p_i (i = 1, \cdots, n)$ and *n-1* intervals $interval_j = \overline{p_j p_{j+1}} (j = 1, \cdots, n-1)$ Each interval inherits its operation from the segment where the interval locates. For operation $oper_k$("*field*" or "*road*"), precision, recall and F1 score are defined in Eq. (3)–(5), respectively, where $correct\_intervals(oper_k)$ are the intervals whose operations are *correctly* predicted as $oper_k$, $predicted\_intervals(oper_k)$ are the intervals whose operations are predicted as $oper_k$, $ground\_truth\_intervals(oper_k)$ are the intervals whose *ground truth* operations are $oper_k$, and $\text{time}(\cdot)$ is the summation of the cost time of the considered intervals.

$$\text{Precision}(oper_k) = \frac{Time(correct\_intervals(oper_k))}{Time(predicted\_intervals(oper_k))} \quad (3)$$

$$\text{Recall}(oper_k) = \frac{Time(correct\_intervals(oper_k))}{Time(ground\_truth\_intervals(oper_k))} \quad (4)$$

$$\text{F1 score}(oper_k) = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (5)$$

3.2. Comparisons

3.2.1. Method comparisons

The comparison between the developed method and DBSCAN (a typical method for trajectory segmentation) shows that the developed method can do the field-road segmentation with sufficient accuracy. Table 1 presents the overall performances of two methods, where the developed method is DBSCAN + Field2Road-Cluster + Road2Field-Segment. As shown in Table 1, compared to DBSCAN, the developed method achieved 7.95% increase in F1 scores (from 87.65% to 95.60%).

Table 1
The overall performances of different methods.

Method	P	R	F1
DBSCAN	91.99	89.10	87.65
DBSCAN + Field2Road-Cluster	94.93	95.59	94.05
DBSCAN + Field2Road-Cluster + Road2Field-Segment	***96.01***	***96.29***	***95.60***

P: Precision score; R: Recall score; F1: F1 score.

This indicates that the direction-distribution-based inference can effectively use the characteristic of the parallel direction distribution in fields to correct the two types of false segmentation cases predicted by DBSCAN.

3.2.2. Inference comparisons

To investigate the effects of the two inference rules (*Field2Road-Cluster* and *Road2Field-Segment*), the rules are examined one by one, and the overall and detailed results are listed in Tables 1 and 2, respectively. First of all, as shown from Table 1, the incorporating of *Field2Road-Cluster* (i.e., DBSCAN + Field2Road-Cluster) resulted in 6.40% increase in F1 scores (from 87.65% to 94.05%). Furthermore, as shown in Table 2, this performance gain mainly comes from the significant improvement both on the recall score of "*road*" (13.56%) and on the precision score of "*field*" (7.90%). This indicates that the correction of false field clusters through *Field2Road-Cluster* is very effective, which leads to more true "*road*" points and less false "*field*" points in segmented trajectories. For example, from Fig. 7a to Fig. 7b, the false "*field*" points in the dashed red cycle are corrected by *Field2Road-Cluster*.

Secondly, as shown in Table 1, after incorporating *Road2Field-Segment* (DBSCAN + Field2Road-Cluster + Road2Field-Segment), the F1 scores increased by 1.55% (from 94.05% to 95.60%). Furthermore, as illustrated in Table 2, this performance improvement comes from the increasing both on the precision score of "*road*" (2.15%) and on the recall score of "*field*" (1.53%). This indicates that *Road2Field-Segment* can effectively correct false "*road*" segments, and result in less false "*road*" points and more true "*field*" points in segmented trajectories. For example, from Fig. 7c to Fig. 7d, most false "*road*" points in the dashed red rectangle are corrected by *Road2Field-Segment*.

4. Conclusions

In this paper, a field-road segmentation method that uses the combination of DBSCAN-based clustering and a direction-distribution-based inference to segment trajectories of agricultural machinery was presented. To correct the two types of false segmentation cases in the output of DBSCAN-based clustering, two inference rules (*Field2Road-Cluster* and *Road2Field-Segment*) were proposed. Specifically, *Field2Road-Cluster* uses the difference in direction distribution (parallel in fields vs. not parallel on roads) to correct false field points in a cluster, and *Road2-Field-Segment* uses the parallel relationship among strips in the same field to correct false road points in a segment. According to the test results on 60 selected trajectories, the direction-distribution-based inference achieved sufficient accuracy and the two inference rules increased the overall performance by 7.95%.

This developed field-road segmentation method can work as functionality in a trajectory segmentation system which provides a sequence of segments for agricultural machinery based on its operations. Based on the segmented trajectory, different types of quantitative analysis of operations can be carried out.

CRediT authorship contribution statement

Ying Chen: Methodology, Writing - original draft, Writing - review &

Table 2
The performances of different methods on "field" and "road" operations.

	Field			Road		
Method	P	R	F1	P	R	F1
DBSCAN	88.11	97.32	90.36	95.16	81.62	84.92
DBSCAN + Field2Road-Cluster	96.01	95.74	94.89	93.61	***95.18***	92.91
DBSCAN + Field2Road-Cluster + Road2Field-Segment	***96.02***	***97.27***	***95.93***	***95.76***	95.15	***95.01***

editing. **Xiaoqiang Zhang:** Software, Writing - review & editing. **Caicong Wu:** Supervision, Project administration. **Guangyuan Li:** Data curation.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The research was financially supported by National Development and Reform Commission (Grant/Contract number: JZNYYY001), Ministry of Science and Technology of the People's Republic of China (Grant/Contract number: 2016YFB0501805).

References

Ankerst, M., Breunig, M.M., Kriegel, H.P., Sander, J., 1999. OPTICS: Ordering Points to Identify the Clustering Structure. SIGMOD Rec. (ACM Spec. Interes. Gr. Manag. Data). https://doi.org/10.1145/304181.304187.

Bochtis, D.D., Sørensen, C.G., Green, O., Moshou, D., Olesen, J., 2010. Effect of controlled traffic on field efficiency. Biosyst. Eng. https://doi.org/10.1016/j.biosystemseng.2009.10.009.

Ester, M., Kriegel, H.-P., Sander, J., Xu, X., 1996. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining.

Griffel, L.M., Vazhnik, V., Hartley, D.S., Hansen, J.K., Roni, M., 2020. Agricultural field shape descriptors as predictors of field efficiency for perennial grass harvesting: An empirical proof. Comput. Electron. Agric. https://doi.org/10.1016/j.compag.2019.105088.

Grisso, R.D., Jasa, P.J., Rolofson, D.E., 2002. Analysis of traffic patterns and yield monitor data for field efficiency determination. Appl. Eng. Agric.

Grisso, R.D., Kocher, M.F., Adamchuk, V.I., Jasa, P.J., Schroeder, M.A., 2004. Field efficiency determination using traffic pattern indices. Appl. Eng. Agric.

Hanke, S., Trösken, L., Frerichs, L., 2018. Development and parameterization of an object-oriented model for describing agricultural process steps. Landtechnik. https://doi.org/10.15150/lt.2018.3179.

Hunt, D., 2001. Farm Power and Machinery Management. IOWA Univ. Press I, pp. 77–93.

Jensen, M.A.F., Bochtis, D., 2013. Automatic recognition of operation modes of combines and transport units based on GNSS trajectories. In: IFAC Proceedings Volumes (IFAC-PapersOnline). https://doi.org/10.3182/20130828-2-SF-3019.00059.

Kilic, T., Zezza, A., Carletto, C., Savastano, S., 2017. Missing(ness) in Action: Selectivity Bias in GPS-Based Land Area Measurements. World Dev. https://doi.org/10.1016/j.worlddev.2016.11.018.

Kortenbruck, D., Griepentrog, H.W., Paraforos, D.S., 2017. Machine operation profiles generated from ISO 11783 communication data. Comput. Electron. Agric. https://doi.org/10.1016/j.compag.2017.05.039.

KTBL, 2016. Betriebsplanung Landwirtschaft 2016/17. Darmstadt, Kuratorium für Technik und Bauwesen in 685 der Landwirtschaft e.V.

Marketos, G., Pelekis, N., Frentzos, E., Raffaetà, A., Ntoutsi, I., Theodoridis, Y., 2008. Building real-world trajectory warehouses. In: MobiDE 2008 - Proceedings of the 7th ACM International Workshop on Data Engineering for Wireless and Mobile Access. https://doi.org/10.1145/1626536.1626539.

Miao, Y., Wang, M., 2011. Research on Path Planning Strategy in Agricultural Vehicle Guidance System. J. Agric. Mech. Res. 12–15.

Prelipcean, A.C., Gidofalvi, G., Susilo, Y.O., 2016. Measures of transport mode segmentation of trajectories. Int. J. Geogr. Inf. Sci. https://doi.org/10.1080/13658816.2015.1137297.

Schuessler, N., Axhausen, K., 2009. Processing raw data from global positioning systems without additional information. Transp. Res. Rec. https://doi.org/10.3141/2105-04.

Spaccapietra, S., Parent, C., Damiani, M.L., de Macedo, J.A., Porto, F., Vangenot, C., 2008. A conceptual view on trajectories. Data Knowl. Eng. https://doi.org/10.1016/j.datak.2007.10.008.

Tietbohl, A., Bogorny, V., Kuijpers, B., Alvares, L.O., 2008. A clustering-based approach for discovering interesting places in trajectories. In: Proceedings of the ACM Symposium on Applied Computing. https://doi.org/10.1145/1363686.1363886.

Tran, L.H., Nguyen, Q.V.H., Do, N.H., Yan, Z., 2011. Robust and Hierarchical Stop Discovery in Sparse and Diverse Trajectories. Tech. Rep. EPFL.

Xiang, L., Gao, M., Wu, T., 2016. Extracting stops from noisy trajectories: A sequence oriented clustering approach. ISPRS Int. J. Geo-Information. https://doi.org/10.3390/ijgi5030029.

Xiang, Y., Noguchi, N., 2014. Development and evaluation of a general-purpose electric off-road robot based on agricultural navigation. Int. J. Agric. Biol. Eng. https://doi.org/10.3965/j.ijabe.20140705.002.

Zheng, Y., 2015. Trajectory data mining: An overview. ACM Trans. Intell. Syst. Technol. https://doi.org/10.1145/2743025.