

Research Article

An Algorithm Using DBSCAN to Solve the Velocity Dealiasing Problem

Wei Zhao ¹, Qinglan Li ¹ and Kuifeng Jin ²

¹Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China

²Guangdong Meteorological Public Service Center, Guangzhou 510650, China

Correspondence should be addressed to Qinglan Li; ql.li@siat.ac.cn

Received 6 May 2021; Revised 17 September 2021; Accepted 15 November 2021; Published 17 December 2021

Academic Editor: Tomeu Rigo

Copyright © 2021 Wei Zhao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Velocity dealiasing is an essential task for correcting the radial velocity data collected by Doppler radar. To improve the accuracy of velocity dealiasing, traditional dealiasing algorithms usually set a series of empirical thresholds, combine three- or four-dimensional data, or introduce other observation data as a reference. In this study, we transform the velocity dealiasing problem into a clustering problem and solve this problem using the density-based spatial clustering of applications with noise (DBSCAN) method. This algorithm is verified with a case study involving radar data on the tropical cyclone Mangkhut in 2018. The results show that the accuracy of the proposed algorithm is close to that of the four-dimensional dealiasing (4DD) method proposed by James and Houze; yet, it only requires two-dimensional velocity data and eliminates the need for other reference data. The results of the case study also show that the 4DD algorithm filters out many observation gates close to the missing data or radar center, whereas the proposed algorithm tends to retain and correct these gates.

1. Introduction

Radial velocity data obtained from the Doppler weather radar can be used for wind retrieval [1], convective cloud detection [2], and warning of severe wind and other disasters [3]. Velocity ambiguities often coincide with disastrous weather. The equation for, V_n , the unambiguous or Nyquist, velocity (i.e., the maximum target velocity that will produce no aliasing) is

$$V_n = \frac{PRF * \lambda}{4}, \quad (1)$$

where PRF is the repetition frequency and λ is the wavelength of transmitted energy. Radial velocities that fall outside the range of $\pm V_n$ will be aliased into the range of $\pm V_n$ [4]. Because the unambiguous range R is also sensitive to PRF , the equation for R is $R = c / (2 * PRF)$, where c is the speed of the electromagnetic wave. Therefore, it is impossible to increase the unambiguous velocity and unambiguous range simultaneously by using uniform PRF [5]. A way to increase the unambiguous velocity based on hardware is to

increase the wavelength λ , but many other factors need to be considered when selecting a wavelength for radar. There will be a tradeoff between the practical constraints of size, weight, and cost and the relationship between the wavelength and the size of the target hydrometeors [6].

In the past 40 years, many software methods have been proposed for velocity dealiasing [7]. The equation for the true velocity V is

$$V = V_m + 2n * V_n, \quad (2)$$

where V_m is the measured radial velocity and the integer n is the Nyquist number, which denotes the aliasing interval.

One-dimensional techniques were first designed because of their simplicity and ease of use. Ray and Ziegler [5] developed a dealiasing technique based on searching the largest gap where there are no velocities below a prescribed probability threshold in a single tilt. This method is difficult to apply in cases where the velocity folding is too severe for the gap to be detected easily. Barga and Brown [8] developed another method that assumes continuity along radials and adds manual controls to allow the user to set

reasonable boundary conditions, but this method is not automated.

Subsequent studies showed that using the information in the azimuth dimension can effectively improve the accuracy of velocity dealiasing. The local environment dealiasing (LED) technique is one of the widely used methods [4]. The LED method incorporates both supplemental wind information and two-dimensional continuity and has proved to be efficient in many cases. There are many empirical thresholds, such as the radial continuity threshold and adaptive threshold, in the LED method. When the algorithm deals with some values close to the threshold, some incorrect aliasing will appear, and if one velocity is placed in the wrong aliasing interval, the following velocities may also be placed in the wrong aliasing interval; so, numerous error check processes are required [9].

In two-dimensional velocity dealiasing algorithms, reference information obtained by the traditional velocity azimuth display (VAD) method [10, 11] is commonly used [12]. The traditional VAD is vulnerable to alias errors in the raw data; so, it needs some other data (such as radiosondes, wind profiler measurements, or numerical weather prediction data) to serve as reference information. The reliance on additional information limits the widespread use of this algorithm. In follow-up research, some modified VAD methods [13, 14] were proposed and were successfully tested with many cases of severely aliased radial velocity. However, those methods rely on high coverage of available data and low levels of noise [15], and some of the failures were reported in the cases with local small-scale wind shear [3]. Then, Xu et al. [16] proposed the alias-robust VAD (AR-VAD) technique and an AR-VAD-based dealiasing method to reduce false dealiasing. Although it corrected some mesocyclone regions, some isolated data areas far away from the radar were rejected [9].

Bergen and Albers [17] added a third dimension of information to a two-dimensional dealiasing scheme [18]. This three-dimensional technique is adequate to resolve isolated data and reduces the need for auxiliary wind information. This idea of providing additional information to the velocity dealiasing algorithm has been further extended to include a fourth dimension, time. James and Houze [15] proposed a four-dimensional dealiasing (4DD) method. This method is available in a widely used open-source Python module called the Python ARM Radar Toolkit (Py-ART) [19] and has been proved to be sufficiently robust to deal with the most severely aliased situation. The reference information of vertical and time continuity used by the 4DD algorithm causes the dealiasing algorithm to perform very similarly to hand-dealiasing [20]. However, it also makes the algorithm difficult to use because three- and four-dimensional methods require at least one “whole volume scan” radar data. If scanning data of some tilts are missing or only one-tilt scan data are given, these methods cannot dealias velocities correctly. Sometimes, researchers prefer to compromise on accuracy in exchange for greater ease of use.

Cluster analysis is an unsupervised learning method that constitutes a cornerstone of many data analysis processes [21], and it has been widely used in radar data analysis

[22, 23]. Unlike the K-means algorithm, which is a widely used clustering algorithm, density-based spatial clustering of applications with noise (DBSCAN) is a clustering algorithm that can obtain clusters with any arbitrary shape and size [24]. Because the region with unambiguous velocity is not always approximate to a convex hull, arbitrariness of the clustering shape is necessary. Another advantage of DBSCAN is that it is robust when dealing with noisy data.

In this study, we propose a dealiasing algorithm with few empirical parameters and ensure that the occasional incorrect aliasing will not affect the dealiasing of other gates. We convert the velocity dealiasing problem into a general clustering problem so that we can bypass the problem of calculating the reference velocity. To solve the clustering problem, the DBSCAN method is introduced in this case study. This new dealiasing algorithm needs only three experimental parameters and may achieve a similar accuracy as the 4DD algorithm.

2. Methodology

2.1. Data Preprocessing. Because the DBSCAN method can treat outliers as noise points [25], the noise removal or filtering process in many previous algorithms is not necessary. The data preprocessing in this study mainly involves dealing with the problem of data normalization [26] before the application of DBSCAN.

DBSCAN is not very sensitive to the normalization method; so, the only requirement is to ensure that the order of magnitude of each variable is consistent. The X and Y coordinates of gates are normalized by the following equations:

$$\begin{aligned} X^* &= \frac{X}{(M \cdot S)}, \\ Y^* &= \frac{Y}{(M \cdot S)}, \end{aligned} \quad (3)$$

where X^* and Y^* are the normalized coordinates, X and Y are the original coordinates, M is the number of gates in a single radius, and S is the radial spacing of the gates. Thus, the normalized coordinates change in the range of -1 to 1 .

In addition to the coordinates, the radial velocity also needs to be normalized by the following equation:

$$RV^* = \frac{RV}{(2 \cdot V_n)}, \quad (4)$$

where RV^* is the normalized radial velocity, RV is the original radial velocity, and V_n is the unambiguous velocity of a radar. This normalization is used not only for the measured radial velocities but also for the true radial velocity. The measured radial velocity after normalization changes in the range of -0.5 to 0.5 . The true radial velocity after normalization changes in a larger range than the measured velocity. For example, when the integer n changes in the set of $\{-1, 0, 1\}$, the true radial velocity after normalization is in the range of -1.5 to 1.5 . The more the set elements of the unknown integer n , the larger the true radial

velocity range after normalization. In general, the normalized true radial velocity is still of the same order of magnitude as the normalized coordinates.

2.2. Definition. To convert the dealiasing problem to a general clustering problem, the definitions of “object,” “position,” and “distance” are needed [27]. The number of objects to be clustered is related to the range of the Nyquist number. Assume there are N gates (equal to the total number of observations gates minus the number of missing measurements and distance ambiguity) in a single tilt; note that the capital N here is different from the previous Nyquist number n in equation (2), and if the Nyquist number n is restricted to values of ± 1 and 0, then there are $3N$ objects waiting to be clustered, and if the Nyquist number n in equation (2) is restricted to values of ± 2 , ± 1 , and 0, then there are $5N$ objects waiting to be clustered. Because the X coordinates, Y coordinates, and radial velocity have different units, the normalized data defined above are more suitable to represent the positions of objects. The set of objects relating to gates is

$$G_0 = \{(X_i^*, Y_i^*, RV_i^*) | i = 1, 2, \dots, N\}, \quad (5)$$

where X_i^* is the normalized X coordinate of the gate “ i ,” Y_i^* is the normalized Y coordinate of the gate “ i ,” and RV_i^* is the normalized radial velocity of the gate “ i .”

If dealiasing can be completed with n restricted to values of ± 1 and 0, there are two (corresponding to the number of nonzero values of the Nyquist number) sets of objects to be clustered in addition to G_0 :

$$\begin{aligned} G_{+1} &= \{(X_i^*, Y_i^*, RV_i^* + 2V_n^*) | i = 1, 2, \dots, N\}, \\ G_{-1} &= \{(X_i^*, Y_i^*, RV_i^* - 2V_n^*) | i = 1, 2, \dots, N\}, \end{aligned} \quad (6)$$

where V_n^* is the normalized unambiguous velocity, which equals 0.5. The total set of objects to be clustered is G , which is given by

$$G = G_0 \cup G_{+1} \cup G_{-1}. \quad (7)$$

Another important definition of DBSCAN is the distance used to measure the similarity of any two objects p and q . For simplicity, it is defined as the Euclidean distance [28]:

$$\text{dist}_{p,q} = \sqrt{(X_p^* - X_q^*)^2 + (Y_p^* - Y_q^*)^2 + (RV_p^* - RV_q^*)^2}, \quad p, q \in G, \quad (8)$$

where (X_p^*, Y_p^*, RV_p^*) is the position of object p and (X_q^*, Y_q^*, RV_q^*) is the position of object q .

2.3. The DBSCAN Method. DBSCAN is a popular density-based clustering algorithm that can discover clusters of any arbitrary shape and size in databases containing even noise and outliers [29]. Traditional DBSCAN has only two parameters: the minimum number of neighbors and the neighborhood radius ε [30]. Objects with more than minPts neighbors within ε are considered to be a core point. If one core point is one of the neighbors within ε of another core

point (called direct density reachable), the two core points are considered to be in the same cluster. The pseudocode of DBSCAN is shown in Figure 1 [24].

The input feature set G and the distance function dist of the function RangeQuery in Figure 1 are defined as above. The value of minPts is related to the number of radials in a single tilt scan and the number of gates in a radius. Based on experience, minPts can be taken in the range of 50–125. The greater the number of radials in a single tilt scan or the greater the number of gates in a radius, the greater the minPts. In this study, minPts equals 80 because it is experimentally suitable for the observation data of China Next Generation Weather Radar (CINRAD). In our research, the clustering result is not sensitive to minPts. According to previous research [30], one strategy for estimating the value of the neighborhood radius ε is to generate a top- k distance graph for the input data G . For each object in G , we find the distance to the k th nearest point and plot sorted points against this distance. The graph usually contains a knee, as shown in Figure 2. The distance that corresponds to the knee is generally a good choice for ε because it is the region where points start tailing off into outlier (noise) territory. In many cases of using data obtained from CINRAD, we find that the knee is usually in the range of 0.05–0.10, and when minPts = 80, $\varepsilon = 0.08$ is relatively robust.

2.4. Conversion of Clustering Results to Dealiasing Result. After being clustered by DBSCAN, each object in set G has a label indicating that the object belongs to a particular group or noise. The next step is to choose reasonable groups to reflect the true radial velocity. An example of the preclustered set G and the clustered result of DBSCAN can be seen in Figure 3. Generally, there is a “major group” (denoted as M) among the clustered groups, which can be defined as the group with the smallest mean in the set of largest groups. The condition “in the set of largest groups” ensures that the number of objects in the major group is the maximum, and the condition “the smallest mean” ensures that the group reflects the true radial velocity because there is less possibility that winds blow towards (or away from) the radar from all directions. By this definition, group 2 in Figure 3(b) is suggested to be the major group.

Once the major group G_M is selected, the groups containing objects with the same X and Y coordinates as objects in the major group should be removed. If there are still some groups remaining (this situation is not very common), we can select a minor group G_s from those remaining groups according to the following rules:

$$\arg \min_{G_s} \sum_i \frac{(A \cdot X_i^* + B \cdot Y_i^* + C \cdot RV_i^* + D)^2}{\sqrt{A^2 + B^2 + C^2}}, \quad i \in G_s, \quad (9)$$

where A , B , C , and D are the plane regression coefficients of the major group and (X_i^*, X_i^*, X_i^*) are the coordinates of the objects in group G_s . Once the minor group G_s is selected, the groups containing objects with the same X and Y coordinates as the objects in G_s should be removed. If there are still

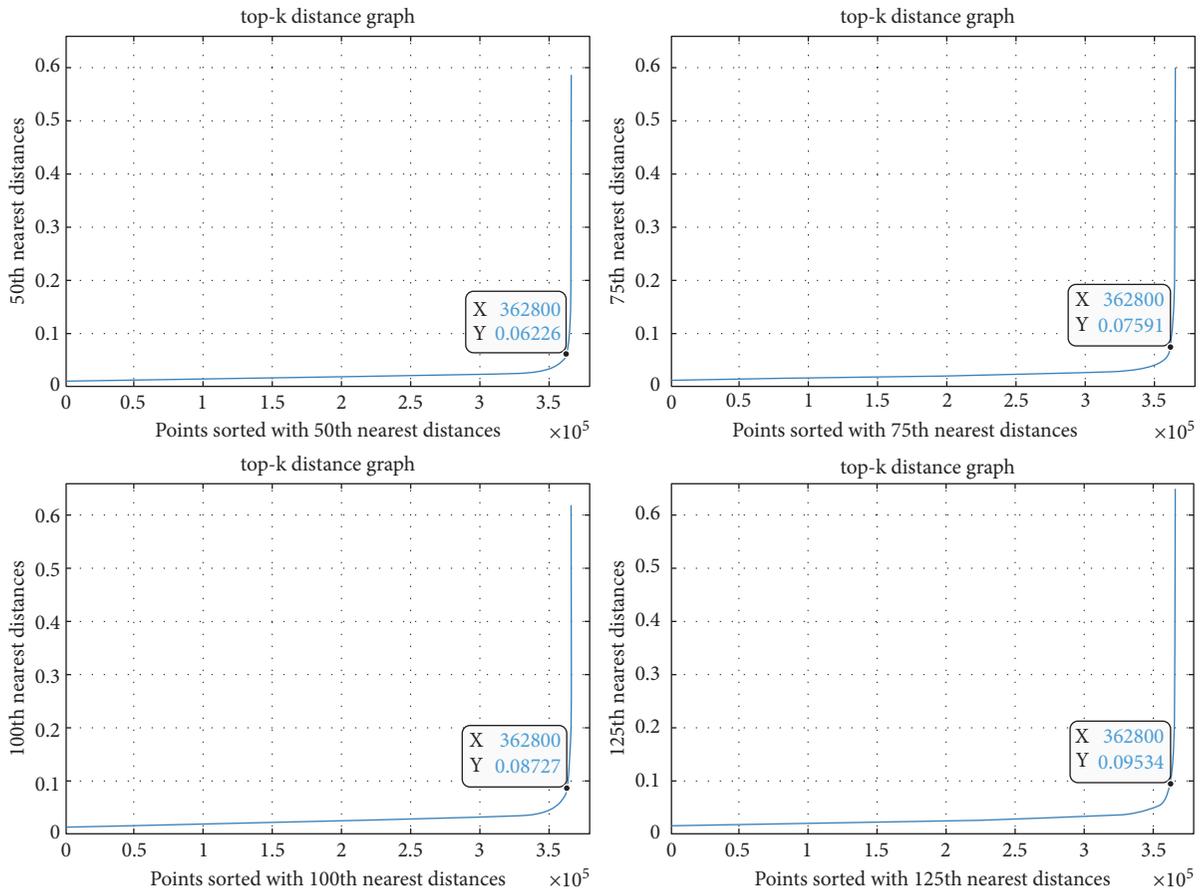
Algorithm 1: DBSCAN

```

Input:  $G$  : Set of objects to be clustered
Input:  $\varepsilon$  : Neighborhood radius
Input:  $minPts$  : Minimum number of neighbors
Input:  $dist$  : Distance function
Data:  $label$  : Point labels, initially undefined
1 foreach object  $o$  in set  $G$  do
2   if  $label(o) \neq undefined$  then
3     continue // Skip processed points
4   set of neighbors  $N_o \leftarrow RangeQuery(G, dist, o, \varepsilon)$  // Find initial neighbors
5   if  $\|N_o\| < minPts$  then
6      $label(o) \leftarrow Noise$ 
7     continue
8    $c \leftarrow$  next cluster label // Start a new cluster
9    $label(o) \leftarrow c$ 
10  Seed set  $S \leftarrow N_o \cup \{o\}$  // Expand neighborhood
11  foreach  $q$  in  $S$  do
12    if  $label(q) = Noise$  then
13       $label(q) \leftarrow c$ 
14    if  $label(q) \neq undefined$  then
15      continue
16    set of neighbors  $N_q \leftarrow RangeQuery(G, dist, q, \varepsilon)$ 
17     $label(q) \leftarrow c$ 
18    if  $\|N_q\| < minPts$  then
19      continue
20     $S \leftarrow S \cup N_q$ 

```

FIGURE 1: Pseudocode of DBSCAN.

FIGURE 2: Top- k distance graph with $k = 50, 75, 100,$ and 125 .

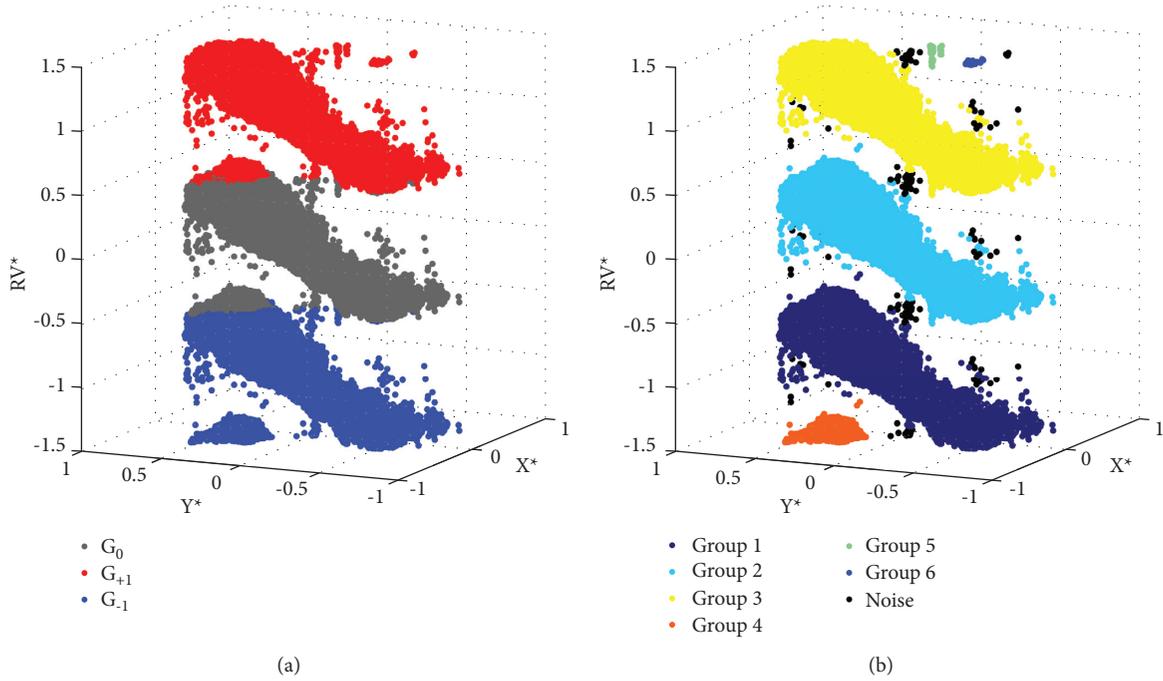


FIGURE 3: The preclustered set G and the clustered result of DBSCAN. (a) The preclustered set G . (b) The clustered result of DBSCAN.

some groups remaining, another minor group should be selected, and this process should be repeated until no groups remain.

The number of iterations of this algorithm is less than the number of cluster groups divided by the number of subsets on the right side of equation (7). Since DBSCAN does not produce too many clusters in this case study (the number of clusters is usually less than 20), such iterations will soon end. Objects with X and Y coordinates outside the major and minor groups are noisy. Then, the objects in the major group and minor group(s) can be used to reflect the estimated velocities and their positions by the following equations:

$$\begin{aligned} RV &= 2 \cdot V_n \cdot RV^*, \\ X &= M \cdot S \cdot X^*, \\ Y &= M \cdot S \cdot Y^*. \end{aligned} \quad (10)$$

The meanings of the symbols in these equations are the same as previously defined.

3. Results and Discussion

3.1. Data Description. On September 17, 2018, a tornado occurred in Foshan, Guangdong Province, China, and caused extensive damage along its northern movement path to Zhaoqing [31]. At the same time, a strong southerly wind was caused by the super typhoon Mangkhut in the north of Zhaoqing [32]. Doppler velocity data obtained from the CINRAD in Zhaoqing during the 1.5 h period from 00:30 UTC to 02:00 UTC are selected to test the method. The frequency of CINRAD is 2700–3000 MHz; its maximum observation range of the radial wind speed is 150 km, and its resolution is 250 m. CINRAD takes 6 min to complete a

volume scan. The Nyquist velocity of CINRAD in Zhaoqing is 26.60 m/s, and there are 920 gates in a radius. The azimuthal increment is not quite uniform but is around 1° or less. The volume scan data coverage changes with nine tilt angles (0.5° , 1.5° , 2.4° , 3.4° , 4.3° , 6.0° , 9.9° , 14.6° , and 19.5°). This poses a challenging task because large-scale aliased velocities and tornado-induced aliased velocities both occur in the selected time and place.

3.2. Results of the Application. Figure 4 shows the raw radial velocities scanned at 1.5° tilt by the CINRAD at 00:49 UTC on September 17, 2018. The missing data (MD), which are marked in black, include data affected by missing reflectivity and range folding. It can be seen that the observed radial velocities were severely aliased in the two main areas, a large area caused by the tropical cyclone Mangkhut and a small area caused by the tornado near Zhaoqing (marked by the two white frames in Figure 4). In addition to these two main areas, there were some isolated areas (marked by capital letters in Figure 4) that were also caused by Mangkhut.

To test the robustness of the algorithm, where robustness means that the continuous dealiasing of gates will not be affected by the occasional wrong aliasing, we design a special algorithm application flow. Firstly, the gates in a tilt are assigned to three sets. The gates in the same set are radially discontinuous. Secondly, the radial velocities in these three sets are dealiased separately. Finally, the dealiased velocities of the three sets are combined. If the final result is good, it is proved that the algorithm can be used to dealias velocities in one radially discontinuous set without depending on the information of the other two sets. If one wrong dealiasing occurs in one set, the results of the other two sets are not affected.

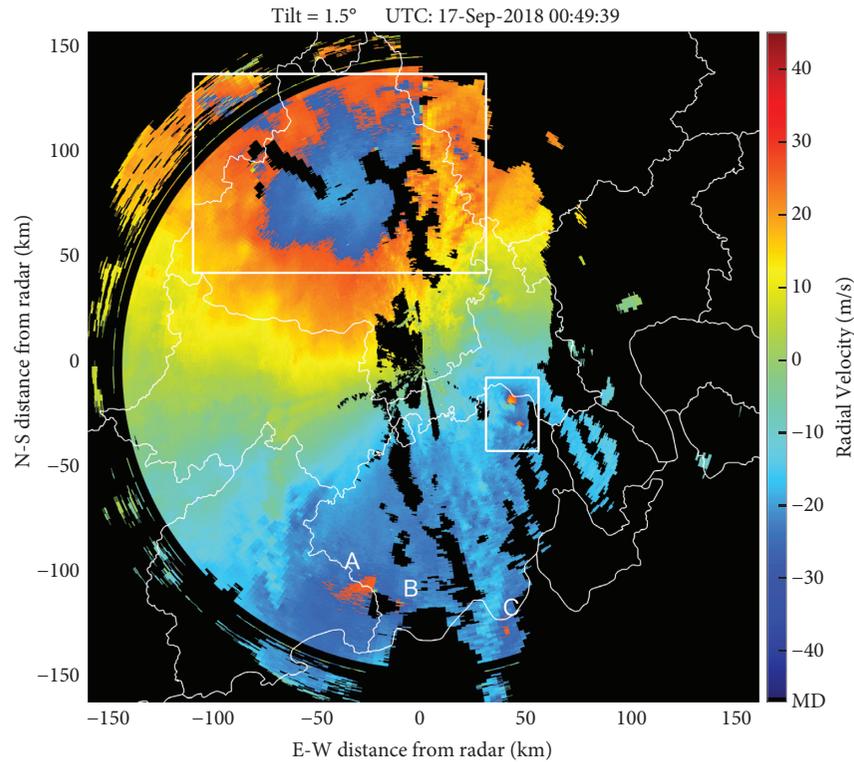


FIGURE 4: Radial velocity measured on 1.5° tilt from CINRAD in Zhaoqing at 00:49 UTC on September 17, 2018.

The results of dealiasing each set separately are shown in Figures 5(a), 5(b), and 5(c). It can be seen that discontinuous gates in each set can be dealiased well, regardless of whether they are in one of the two main areas or isolated areas. Unlike a point-to-point continuity check, DBSCAN uses all objects in radius ϵ when judging the classification of an object. This mechanism brings robustness and allows the algorithm to deal with cases in which many continued velocities are missing. The number of gates in the unit area is only about one third of the original number because all gates are divided into three sets. In this case, minPts is still taken as 80, but good dealiasing results can be obtained. It means that although minPts is related to the number of radials in a single tilt scan and the number of gates in a radial, this dealiasing algorithm is not very sensitive to this parameter.

The results of combined dealiasing are shown in Figure 5(d). It can be seen that the radial velocity field is corrected by the new method without any false dealiasing. As shown in Figure 5(e), the measured velocities are quite noisy in the immediate vicinity of the radar within the 25 km range. The radial velocities at some gates in Figure 5(e) are 20–30 m/s or –30 to –15 m/s, whereas the radial velocities at adjacent gates are close to 0. In equation (2), $2 \times V_n$ is 53.2 m/s; so, the Nyquist number n has no appropriate integer value. Discontinuous gates that cannot be corrected by equation (2) are considered as noise and are marked in gray in Figure 5(f). The strategy of classifying velocity as noise is relatively conservative, and there is no need to select any velocity threshold. Only when the objects (X_i^*, Y_i^*, RV_i^*) , $(X_i^*, Y_i^*, RV_i^* + 2V_n^*)$, and

$(X_i^*, Y_i^*, RV_i^* - 2V_n^*)$ do not belong to the major group or any minor group will the corresponding gate be regarded as noise.

3.3. Comparison with the 4DD Method. There are 144 tilts contained in the 16 volume scans for the 1.5 h period. Due to the body scan setting of CINRAD in Zhaoqing, the reflectivity field and radial velocity field are not observed synchronously at the elevation angles of 0.5° and 1.5°. Thus, the 4DD method does not work well on these two angles. We compare the DBSCAN-based algorithm with the 4DD algorithm in the remaining tilts (i.e., seven elevation angles above 1.5°) and count the two methods' analysis results of 10740221 gates in these 112 tilts. A comparison is presented in Table 1. In Table 1, the value of each cell represents the number of occurrences of a certain situation. For example, the first row of the first column indicates 8297427 occurrences of the following situation: the DBSCAN-based algorithm determines that the Nyquist number n at one gate should be 0, and the 4DD method also determines that n at this gate should be 0.

It can be seen that most of the values of integer n obtained by the two methods are the same. This means that the DBSCAN-based algorithm achieves an accuracy that is close to that of the 4DD algorithm. A total of 2152822 measured radial velocities are filtered by the 4DD algorithm. Most of these points are concentrated near the center of the radar or near the margin of the missing value (figure omitted). On the contrary, only 5733 gates are treated as noise gates by the DBSCAN-based algorithm. This means that the DBSCAN-

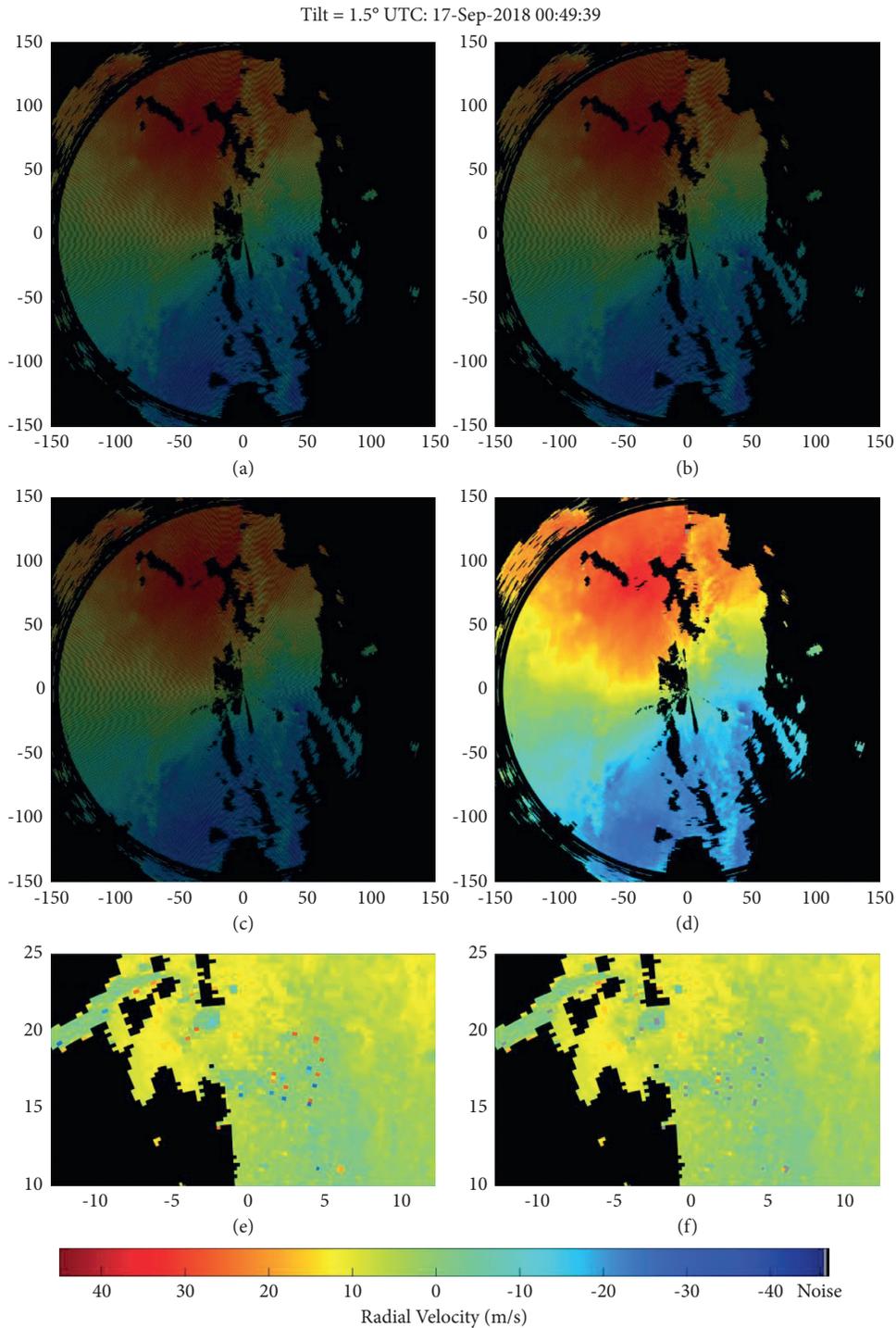


FIGURE 5: (a), (b), and (c) The dealiasing results of three radially discontinuous sets. (d) The combined dealiasing results. (e) Measured radial velocities near the radar within 25 km. (f) Dealiasied radial velocities near the radar within 25 km.

based algorithm tends to retain as much observation information as possible and attempts to revise it instead of discarding it directly.

3.4. Discussion. Following the conversion into a clustering problem, this problem can not only be solved by DBSCAN but also by any clustering method meeting the following

conditions: (1) the number of clusters does not need to be known in advance, (2) the shape of the class can be arbitrary, and (3) the noise points can be identified. The purpose of this paper is not to improve the accuracy of velocity dealiasing but to improve the robustness of the algorithm and reduce the requirement of data integrity.

The primary limitation of this algorithm is that clustering is time intensive. In this case study, it takes about

TABLE 1: The comparison results of the DBSCAN-based algorithm and the 4DD algorithm.

		DBSCAN-based algorithm			
		$n = 0$	$n = 1$	$n = -1$	Noise
4DD algorithm	$n = 0$	8297427	4	5	1462
	$n = 1$	14	209326	1	125
	$n = -1$	2	5	78664	364
	Filtered	2115429	22630	10981	3782

60–95 s to complete an algorithm run on the 3.40 GHz quad-core computing platform. Although this method works well for the research environment, it is not suitable for real-time applications. We hope to improve this method by reasonably designing subsets containing part of the data to be clustered and choosing a better clustering method in the next study.

4. Conclusion

In this paper, a DBSCAN-based algorithm is proposed for Doppler radar velocity dealiasing. After normalizing the coordinates and radial velocities, constructing the set of objects to be clustered G , and defining the “position” and “similarity distance,” the velocity dealiasing problem is converted into a clustering problem. Because clusters can be of any arbitrary shape and size and the databases contain noise and outliers, we choose the DBSCAN method to solve the clustering problem. After the clustering problem is solved, we choose one major group and some other minor groups from the clustering result and reconstruct the radial velocity field.

The process only involves three parameters, i.e., the Nyquist velocity V_n , the minimum number of neighbors, minPts , and the neighborhood radius, ϵ . In this study, minPts is set to 80 because it is experimentally suitable for CINRAD. The top- k distance graph shows that the neighborhood radius ϵ is not sensitive to minPts , and the knee point of the top- k distance is usually in the range of 0.05–0.10. The value of minPts is set to 0.08 in the following case study.

As a case study, the Doppler velocity data obtained from the CINRAD in Zhaoqing during a 1.5-h period from 00:30 UTC to 02:00 UTC are selected to test the DBSCAN-based algorithm. During this period, a tropical cyclone caused large-scale aliased velocities and a tornado caused small-scale aliased velocities near Zhaoqing. The results of the case study show that the DBSCAN-based algorithm can be used to deal with data with radial discontinuities and can achieve an accuracy similar to that of the 4DD algorithm; yet, it only requires one-tilt scanning radial velocity data. The 4DD algorithm filters out a large number of observation gates close to the missing data or radar center, whereas the proposed algorithm is more inclined to retain and correct them.

Data Availability

The Doppler weather radar data used to support the findings of this study may be released upon application to the China

National Meteorological Information Centre, which can be obtained at <https://data.cma.cn/>.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

This study was supported by the Science and Technology Department of Guangdong Province (Grant 2019B111101002) and the Innovation of Science and Technology Commission of Shenzhen Municipality Ministry (Grants JCYJ20170413164957461, JCYJ20180305180905450, and GGF2017073114031767).

References

- [1] S. G. Park and D. K. Lee, “Retrieval of high-resolution wind fields over the southern Korean peninsula using the doppler weather radar network,” *Weather and Forecasting*, vol. 24, no. 1, pp. 87–103, 2009.
- [2] N. Mahavik and S. Tantanee, “Spatio-temporal analysis of convective cloud properties deriving from weather radar reflectivity during the decaying stage of tropical storm over the lower northern Thailand,” *International Journal of Geoinformatics*, vol. 15, no. 3, 2019.
- [3] Y. Yuan, P. Wang, D. Wang, and J. Shi, “A velocity dealiasing scheme based on minimization of velocity differences between regions,” *Advances in Meteorology*, vol. 2020, Article ID 6157636, 12 pages, 2020.
- [4] M. D. Eilts and S. D. Smith, “Efficient dealiasing of doppler velocities using local environment constraints,” *Journal of Atmospheric and Oceanic Technology*, vol. 7, no. 1, pp. 118–128, 1990.
- [5] P. S. Ray and C. Ziegler, “De-aliasing first-moment doppler estimates,” *Journal of Applied Meteorology*, vol. 16, no. 5, pp. 563–564, 1977.
- [6] M. Montopoli and F. S. Marzano, “Meteorological radar systems,” in *Integrated Ground-Based Observing Systems*, pp. 33–57, Springer, Berlin, Germany, 2011.
- [7] V. Louf, A. Protat, R. C. Jackson, S. M. Collis, and J. Helmus, “UNRAVEL: a robust modular velocity dealiasing technique for doppler radar,” *Journal of Atmospheric and Oceanic Technology*, vol. 37, no. 5, pp. 741–758, 2020.
- [8] D. W. Barga and R. C. Brown, “Interactive radar velocity unfolding,” in *Proceedings of the 19th conference on Radar Meteorology*, pp. 278–285, Miami Beach, FL, USA, April 1980.
- [9] Q. Xu, K. Nai, S. Liu, C. Karstens, T. Smith, and Q. Zhao, “Improved doppler velocity dealiasing for radar data assimilation and storm-scale vortex detection,” *Advances in Meteorology*, vol. 2013, Article ID 562386, 10 pages, 2013.
- [10] R. M. Lhermitte, “Precipitation motion by pulse doppler,” in *Proceedings of the 9th Weather Radar Conference*, pp. 218–223, Boston, MA, USA, October 1961.
- [11] K. A. Browning and R. Wexler, “The determination of kinematic properties of a wind field using doppler radar,” *Journal of Applied Meteorology*, vol. 7, no. 1, pp. 105–113, 1968.
- [12] J. Gong, L. L. Wang, and Q. Xu, “A three-step dealiasing method for doppler velocity data quality control,” *Journal of Atmospheric and Oceanic Technology*, vol. 20, no. 12, pp. 1738–1748, 2003.

- [13] Y. Yamada and M. Chong, "Vad-based determination of the nyquist internal number of doppler velocity aliasing without wind information," *Journal of the Meteorological Society of Japan. Ser. II*, vol. 77, no. 2, pp. 447–457, 1999.
- [14] P. Tabary, G. Scialom, and U. Germann, "Real-time retrieval of the wind from aliased velocities measured by doppler radars," *Journal of Atmospheric and Oceanic Technology*, vol. 18, no. 6, pp. 875–882, 2001.
- [15] C. N. James and R. A. Houze Jr., "A real-time four-dimensional doppler dealiasing scheme," *Journal of Atmospheric and Oceanic Technology*, vol. 18, no. 10, pp. 1674–1683, 2001.
- [16] Q. Xu, K. Nai, and L. Wei, "Fitting VAD winds to aliased doppler radial-velocity observations: a global minimization problem in the presence of multiple local minima," *Quarterly Journal of the Royal Meteorological Society*, vol. 136, no. 647, pp. 451–461, 2010.
- [17] W. R. Bergen and S. C. Albers, "Two-and three-dimensional de-aliasing of doppler radar velocities," *Journal of Atmospheric and Oceanic Technology*, vol. 5, no. 2, pp. 305–319, 1988.
- [18] M. W. Merritt, "Automatic velocity de-aliasing for doppler weather radar," Doctoral dissertation, Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, Cambridge, MA, USA, 1984.
- [19] J. J. Helmus and S. M. Collis, "The python arm radar toolkit (Py-ART), a library for working with weather radar data in the python programming language," *Journal of Open Research Software*, vol. 4, 2016.
- [20] S. M. Stough and L. D. Carey, "Observations of anomalous charge structures in supercell thunderstorms in the southeastern United States," *Journal of Geophysical Research: Atmospheres*, vol. 125, no. 17, Article ID e2020JD033012, 2020.
- [21] S. Kotsiantis and P. Pintelas, "Recent advances in clustering: a brief survey," *WSEAS Transactions on Information Science and Applications*, vol. 1, no. 1, pp. 73–81, 2004.
- [22] J. Jun Liu, J. P. Y. Lee, L. Lingjie Li, Z. Q. Zhi-Quan Luo, and K. M. Wong, "Online clustering algorithms for radar emitter classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1185–1196, 2005.
- [23] C. Kuang, C. Wang, B. Wen, and W. Huang, "An applied method for clustering extended targets with UHF radar," *IEEE Access*, vol. 8, pp. 98670–98678, 2020.
- [24] E. Schubert, J. Sander, M. Ester, H. P. Kriegel, and X. Xu, "DBSCAN revisited, revisited: why and how you should (still) use DBSCAN," *ACM Transactions on Database Systems*, vol. 42, no. 3, pp. 1–21, 2017.
- [25] L. Ertöz, M. Steinbach, and V. Kumar, "Finding clusters of different sizes, shapes, and densities in noisy, high dimensional data," in *Proceedings of the 2003 SIAM International Conference on Data Mining*, pp. 47–58, Society for Industrial and Applied Mathematics, San Francisco, CA, USA, May 2003.
- [26] P. Trebuña, J. Halčinová, M. Fil'o, and J. Markovič, "The importance of normalization and standardization in the process of clustering," in *Proceedings of the 2014 IEEE 12th International Symposium on Applied Machine Intelligence and Informatics (SAMII)*, pp. 381–385, IEEE, Herl'any, Slovakia, January 2014.
- [27] A. Rodriguez and A. Laio, "Clustering by fast search and find of density peaks," *Science*, vol. 344, no. 6191, pp. 1492–1496, 2014.
- [28] K. L. Elmore and M. B. Richman, "Euclidean distance as a similarity metric for principal component analysis," *Monthly Weather Review*, vol. 129, no. 3, pp. 540–549, 2001.
- [29] K. Khan, S. U. Rehman, K. Aziz, S. Fong, and S. Sarasvady, "DBSCAN: past, present and future," in *Proceedings of the Fifth International Conference on the Applications of Digital Information and Web Technologies (ICADIWT 2014)*, pp. 232–238, IEEE, Bangalore, India, February 2014.
- [30] M. Ester, H. P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," *Kdd*, vol. 96, no. 34, pp. 226–231, 1996.
- [31] L. Bai, Z. Meng, K. Sueki, G. Chen, and R. Zhou, "Climatology of tropical cyclone tornadoes in China from 2006 to 2018," *Science China Earth Sciences*, vol. 63, no. 1, pp. 37–51, 2020.
- [32] J. Y. He, Y. C. He, Q. S. Li et al., "Observational study of wind characteristics, wind speed and turbulence profiles during super typhoon mangkhut," *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 206, Article ID 104362, 2020.