

Aerosol Plume Detection Algorithm Based on Image Segmentation of Scanning Atmospheric Lidar Data

R. ANDREW WEEKLEY

National Renewable Energy Laboratory, Golden, Colorado

R. KENT GOODRICH

National Center for Atmospheric Research, and Department of Mathematics, University of Colorado Boulder, and Science and Technology in Atmospheric Research, LLC, Boulder, Colorado

LARRY B. CORNMAN

National Center for Atmospheric Research, Boulder Colorado

(Manuscript received 12 June 2015, in final form 1 February 2016)

ABSTRACT


An image-processing algorithm has been developed to identify aerosol plumes in scanning lidar backscatter data. The images in this case consist of lidar data in a polar coordinate system. Each full lidar scan is taken as a fixed image in time, and sequences of such scans are considered functions of time. The data are analyzed in both the original backscatter polar coordinate system and a lagged coordinate system. The lagged coordinate system is a scatterplot of two datasets, such as subregions taken from the same lidar scan (spatial delay), or two sequential scans in time (time delay). The lagged coordinate system processing allows for finding and classifying clusters of data. The classification step is important in determining which clusters are valid aerosol plumes and which are from artifacts such as noise, hard targets, or background fields. These cluster classification techniques have skill since both local and global properties are used. Furthermore, more information is available since both the original data and the lag data are used. Performance statistics are presented for a limited set of data processed by the algorithm, where results from the algorithm were compared to subjective truth data identified by a human.

1. Introduction

A human expert analyzing a movie loop of backscatter lidar images can usually differentiate between aerosol plumes and other similar features resulting from artifacts such as low signal-to-noise ratio (SNR), hard targets, or background fields. Consciously or subconsciously, the human is performing subjective image-processing tasks: clustering regions of like data and then classifying these clusters as either *valid* (i.e., aerosol plumes) or *invalid*. The goal of many image-processing

algorithms is to mimic the clustering and classification processes that the human expert performs. Examples of such algorithms used in the atmospheric sciences include Cornman et al. (1998) and Weekley et al. (2003, 2004, 2010). In the former, the images consisted of Doppler wind profiler radial velocity versus range; in the latter, the images were the time series of anemometer data. These papers described feature detection and quality control algorithms based on image analysis and fuzzy logic methods that mimicked human experts. In fact, many of the techniques described in this work are an outgrowth and generalization of the methods described in Weekley et al. (2010)—that is, image segmentation via clustering, and fuzzy logic classification using properties of the original image data and those in the so-called *lag domain*.

Image segmentation is the decomposition of an image into clusters of pixels that correspond to objects of

 Denotes Open Access content.

Corresponding author address: R. Andrew Weekley, Strategic Energy Analysis Center, National Renewable Energy Laboratory, 15013 Denver West Parkway, Golden, CO 80401.
E-mail: andrew.weekley@nrel.gov

DOI: 10.1175/JTECH-D-15-0125.1

interest—for instance, decomposing a satellite image into clusters of clouds and background features. Methods currently exist to segment images, and they rely on segmenting the image in the original (or physical) domain. The lag domain (sometimes referred to as the *delay domain*) consists of ordered pairs of data offset in either time or space. The motivation behind the use of the lag domain comes from the assumption of spatial and/or temporal correlation in physical processes. For example, it would be expected that the motion of low-inertia particles such as aerosols would be reflective of the wind field within which they are embedded. Therefore, over small space or time offsets, there should be a certain amount of correlation in the backscatter field. On the other hand, data reflective of low SNR would show the opposite character, that is, low correlation. This implies that using information from the lag domain and the physical domain should be useful in differentiating between clusters of aerosols and random clusters found in low-SNR regions. The quantification and use of correlated and uncorrelated fluctuations in lidar data is not new (Lenschow et al. 2000). One of strengths of the proposed algorithm is its use of both the original and lag domains in the classification step. Clusters are found in both domains, and then properties of these clusters are used to differentiate between valid aerosol plumes and clusters emanating from nonplume sources.

In the following, a detailed description of the plume detection algorithm is provided. It should be noted that while the current application of the algorithm is the detection of aerosol plumes in backscatter lidar data, the methods are easily adapted to other data and sensors, where a correlated field is present.

2. Motivation and example

Consider a constant elevation lidar scan at one time t_m . The backscatter value at the range/azimuth location (r_i, θ_j) is denoted $y_{i,j}(t_m)$, and so we can think of the original image as the collection of points, $p_{i,j}(t_m) = [r_i, \theta_j, y_{i,j}(t_m)]$. Next, consider the scan at a previous time t_n $p_{i,j}(t_n) = [r_i, \theta_j, y_{i,j}(t_n)]$. The image in the (temporal) lag domain is then the set of points $[y_{i,j}(t_m), y_{i,j}(t_n)]$ —that is, the ordered pair of backscatter values at the same range/azimuth location but at two different times. It is important to note, in this paper, a full scan of lidar data $p_{i,j}(t_m)$ is treated as a snapshot of the atmosphere at a single point in time. In fact, a full scan of data is collected over multiple seconds, in the case of the lidar data described in this paper, on the order of 20 s. It has been shown in previous work (Sasano et al. 1982) that such a full scan of lidar data is not a perfect snapshot, but is distorted from the fact the

atmosphere changes during the time it takes to collect a full scan of data. Such effects are not addressed in this paper. To simplify the notation, in the following we will denote $y_{i,j}(t_m) = y_{i,j}$ and $y_{i,j}(t_n) = y'_{i,j}$. Typically, spatial structures share similar statistics. For example, the backscatter from hard targets is usually much greater than those any of the other aforementioned categories. One can imagine an algorithm that would cluster data in an image based on self-similar statistics. A simple example of such an algorithm is one that finds all the data above the given threshold. Another way to find adjacent data with similar intensity values is to consider the data in the lag domain. The set of all points of the form (r_i, θ_j) is called the physical domain, the set of all points of the form $(r_i, \theta_j, y_{i,j}, y'_{i,j})$ is called the lag space, and the set of all points of the form $(y_{i,j}, y'_{i,j})$ is called the lag domain. The points in the lag space contain information of not only the data values but also indicate the point in the physical domain where these data values occurred. In the lag domain, the information about where the data values occurred has been lost, but it is easier to visualize points in two dimensions than in four.

Wishart (1969) used the idea of a lag domain in 1969 to find outliers in scatterplots. The lag domain, also called time lag space, is used in dynamics to find attractors (Rosenstein and Cohen 1998) and novelties in time series (Ma and Perkins 2003). Other techniques for image segmentation include the level set method discussed in Airouche et al. (2009). Mathematical morphology is discussed in Shih (2009). Various clustering techniques are compared in Kettaf et al. (1996) and could be used to find plumes in lidar images as well. Some morphology techniques are used in the present paper to classify certain features associated with hard targets. Plumes can be found as a range (statistical) anomaly using hyperspectral signal processing techniques (Ben-David et al. 2007). In the Ben-David paper, each shot is processed one at a time; anomalies may also be detected in time as well using the lag domain. A statistical model for each shot is calculated and assumes the first few range gates consist only of background signal with no anomalies. A probability model is assumed and statistical anomalies are then detected using this model.

In the present paper, fuzzy maps are used in the classification. Once the classification fuzzy maps have been defined in the original space, the largest value of the maps is computed. If this maximum occurs for, say, the plume map, then the point is classified as a plume point. Every point in the entire two-dimensional image is classified, using both the lag domain and the original image, as *dead zone*, noise, persistent, background, or plume points. The dead zone is a region of the backscatter waveform that begins at the origin and tapers off

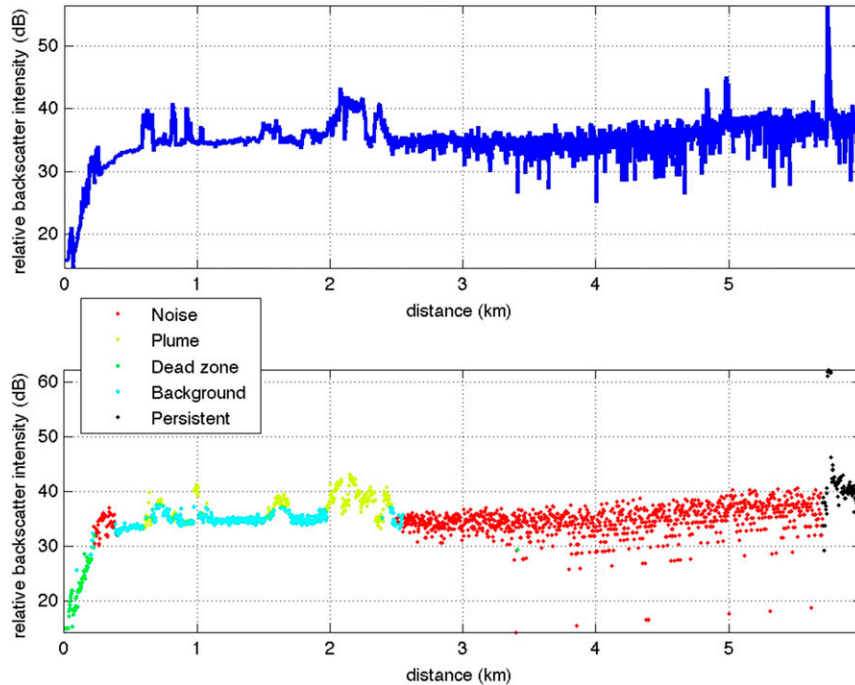


FIG. 1. (top) Backscatter as a function of distance for a given azimuth. (bottom) Backscatter at the same azimuth from the next scan; the color indicates the classification assigned to the data by the algorithm.

gradually due to geometrical overlap of the laser beam and the receiver field of view. In the Raman-Shifted Eye-Safe Aerosol Lidar (REAL), full overlap should be achieved by the 500-m range (Mayor and Spuler 2004; Mayor et al. 2007). In addition the persistent classification will use information from a larger number of previous scans.

Before defining the classifications, a motivating example is presented. This example utilizes two individual shots of the lidar, whereas in the subsequent discussion, the techniques of the algorithm are applied to full PPI scans of data. In this paper a PPI scan is radial data collected at a fixed elevation angle and multiple azimuths. Figure 1 illustrates two radial shots from the same azimuth, at two sequential times, from a horizontal-pointing scanning lidar, that is, the REAL.¹ The horizontal scanning lidar transected a total azimuth of about 71.5° ; shots are equally spaced in time and were assigned to a regularly spaced grid in azimuth every 0.25° , and radial range gates had 3-m spacing from 0 to roughly 6 km, and the elevation angle was near zero. Each scan is roughly 21 s in duration. Two radial shots are shown with the same azimuthal value and separated in time by about 20 s. The first shot in Fig. 1a is shown as a solid blue line.

Notice several features in this data. The data form a ramplike feature, where the intensity increases from 0 to above 30 dB (see the appendix for details on the conversion) in the first third of a kilometer (the dead zone). The data have a flat region with several spikes from 0.5 to about 2.5 km. At distances greater than 3.5 km, the data have a low SNR. A human expert inspecting the entire two-dimensional image determined that the spike in the data at about 5.75 km is a hard target. The other elevated intensity values in the 0.5–2.5-km range were classified as plumes, again from inspecting the two-dimensional image, and the remainder of the points in the flat region, where the intensity was not changing much, were classified as background. The second shot is shown color-coded by the classification algorithm. The green data near the origin in Fig. 1b denote a dead zone, the bright red data are noise, the yellow data are plume, and the sky blue data are classified as background. The black data at about 5.75 km are classified as persistent. The classifications shown in Fig. 1b were found using the current algorithm that uses two-dimensional data from the two scans adjacent in time and the two-dimensional data from the lag domain. The previous 100 two-dimensional scans were used to find the persistent data.

To classify points in the original domain, it is necessary to develop scores in that domain. Some of these scores are transferred from the lag domain by defining a

¹The data presented and analyzed in this paper are the property of STAR LLC.

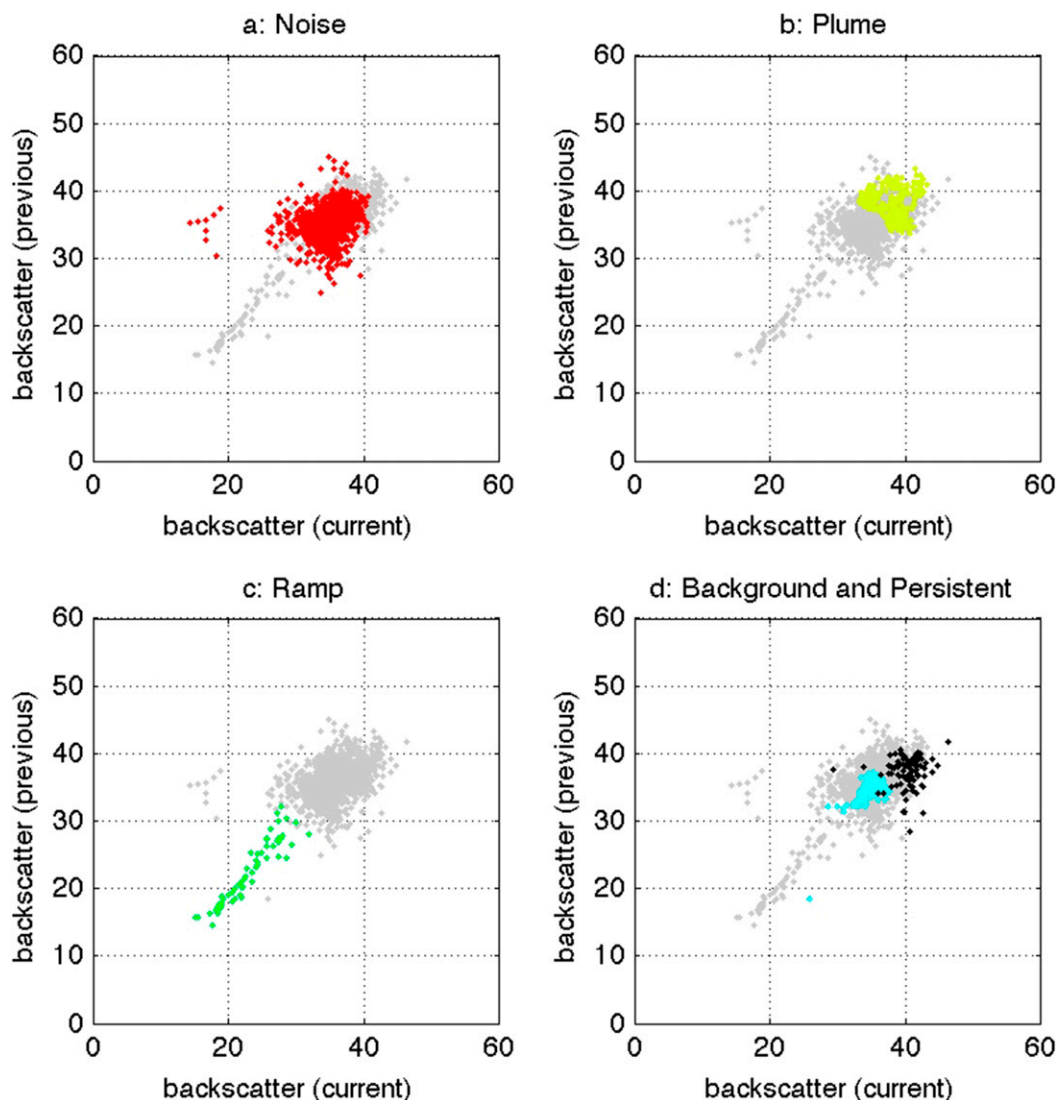


FIG. 2. Scatterplots of the data classified by the algorithm in the lag domain, where the color indicates the classification assigned by the algorithm.

transformation F from the original domain to the lag domain as $F(r_i, \theta_j) = (y_{ij}, y'_{ij})$. Suppose the point $(y_{ij} + y'_{ij})$ has been assigned the score I_{ij} ; then all of the points in $F^{-1}(y_{ij} + y'_{ij})$ will also be assigned the score I_{ij} . In this way, a score is defined for every point in the original domain. Transferring a score from the original domain to the lag domain is more problematic. If the point (r_i, θ_j) has been assigned the score t_{ij} , then we assign the same score to (y_{ij}, y'_{ij}) if $F(r_i, \theta_j) = (y_{ij}, y'_{ij})$. It is possible that the point (y_{ij}, y'_{ij}) could be assigned more than one score this way. This could happen if two different points in the original domain have the same datum in the present and the previous scan. Thus, this assignment is multivalued, and in fact the number of scores could be as large as

the number of elements in $F^{-1}(y_{ij}, y'_{ij})$. In spite of this difficulty, it is still useful to study the location of points in the lag domain. For example, if the point (r_i, θ_j) has been assigned the classification of a noise point, then assign the classification of a noise point to the point (a, b) if $F(r_i, \theta_j) = (y_{ij}, y'_{ij}) = (a, b)$ for some (r_i, θ_j) .

Figure 2 illustrates the classification for noise (Fig. 2a), background, plume (Fig. 2b), dead zone (Fig. 2c), background, and persistent (Fig. 2d) from Fig. 1 in the lag domain. The colors used in Figs. 2a–d correspond to the colors used in Fig. 1. In Figs. 2a–d the horizontal axis is associated with the data from the present scan and the vertical axis is associated with the data from the previous scan. Notice that the classifications shown in Fig. 1 map to coherent clusters in the lag

domain in Fig. 2. The gray data points in each of the figures indicate data with classifications other than the classification under consideration. These figures indicate how clusters with different classifications may overlap in the lag domain.

For example, the noise points overlap or nearly overlap all of the other point types (see Figs. 2b–d). This indicates that backscatter values alone cannot produce a unique classification. Consequently, an independent classification score for noise points was developed, based on certain statistical measures of correlation. Persistent and background points often overlap in the lag domain (Fig. 2d). Similarly, an independent score is required to separate persistence points from the other point types. For example, a hard target such as a building has a large backscatter, but it is also persistent over time. The background points are separated from the other point types by a score based on density in the lag domain. Many points in the background region have a low local spatial and temporal variance and are clustered around a backscatter value near 35 dB. This means that the background points in the original domain will map into a small, densely populated region centered around the point (35,35) in the lag domain. The classification of dead zone data also uses a lag density approach. The remaining classification to consider is plume points, which are classified using a score derived from relative backscatter intensity.

Moving on from classification of radial data, consider the classification of points in a two-dimensional lidar image. The image to be segmented is shown in Fig. 3a and is a scan of relative backscatter intensity (dB) consisting of multiple radial shots. Notice the plumes between 0.5 and 3.5 km. Noise begins to dominate the scan between about 3.5 and 6 km, and the bright data—specifically, values greater than 40 and farther than 3.5 km—are mostly hard targets. A human analyst classified the data by watching time-lapse animations of the data. Figure 3b is the associated lag domain image for the data shown in Fig. 3a and the previous scan (not shown). The $y = x$ line is shown for reference. Notice the structure of full scan data is similar to that for the single shot shown in Fig. 2: there is a well-correlated region between roughly 10 and 30, a broad region with less dense and less correlated data (the arrowhead shape) between 30 and 40, and a second, less dense and sparser region greater than 40.

Sets of scores are to be defined in the original and lag domains to indicate if a point belongs to a plume, noise, background, persistent, or is in the dead zone. The scores are calculated using methods in either the physical domain, the lag domain, or via a combination. For instance, the noise score is calculated using fuzzy logic

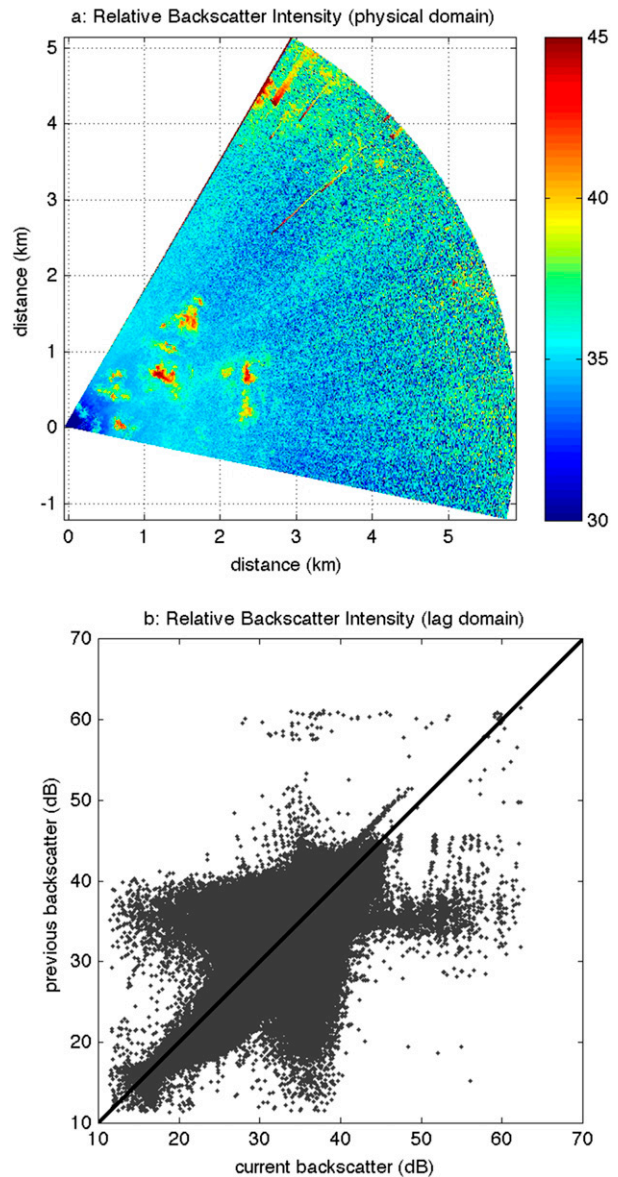


FIG. 3. (a) Backscatter as a function of distance (km) from the lidar located at the origin. (b) Lag domain for the above-mentioned image and the previous image.

techniques and statistics in the physical domain, whereas the background score is based on a density statistic calculated in the lag domain. One might start by calculating a score in the physical domain and then translate it to the lag domain. A new score may be calculated for all the points in the lag domain and then translated back to the physical domain. This back and forth calculation—physical domain to lag domain and back to the physical domain—will be used to calculate a score for the dead zone region of the lidar image. Once all the scores have been calculated, each data point in

the original domain is classified as noise, plume, persistent, dead zone, or background according to the maximum score for each classification type.

3. Fuzzy logic

The classification of the lidar image uses a fuzzy logic approach; that is, scores are calculated for each of the data types, which are then used to determine the final classification assigned to the data. The basic idea behind fuzzy logic is to assign a value that indicates the degree a condition is satisfied rather than a certainty, as would be done with a Boolean zero or one value. For instance, suppose one has a time series with outliers. A fuzzy score might be created to indicate the overall confidence in a given data point, as opposed to a discrete *good-or-bad* assignment (Weekley et al. 2010). Typically, the fuzzy score is in the interval $[0,1]$. Fuzzy scores are often normalized in this way to facilitate combining the scores by fuzzy rules. A membership function can be applied to a statistic (or score) to create a membership value. This can also be seen as a way of classifying data—for example, to determine whether the statistic is small, medium, or large. In the case of an outlier score one might apply a membership function to the score to classify the large outliers. Membership values may be further combined to create additional scores. Statistics and scores for plumes, noise, background, the dead zone, and persistence (which includes hard targets) are developed and discussed. Membership functions are defined and membership values are found and used to classify the data. The statistics themselves are calculated from a number of methods applied in the physical domain, the lag domain, or both.

Fuzzy logic was first used in engineering (Zadeh 1965) and has been used in the atmospheric sciences for some time—for example, the identification and tracking of gust fronts (Delanoy and Troxel 1993) and to improve moment estimation for Doppler wind profilers (Cornman et al. 1998; Morse et al. (2002)). Fuzzy logic has also been used extensively in image processing (see Chi et al. 1996; Blackledge and Turner 2001; Nachtgael et al. 2007). In Weekley et al. (2010) fuzzy logic was applied to both the physical and lag domains to study time series, and it was noted therein that such techniques could be applied to images.

4. The fuzzy power statistic

The REAL measures digitizer count values related to the backscattered intensity as a function of range and azimuth, and the backscatter intensity is large (after preprocessing described in appendix A) for plumes,

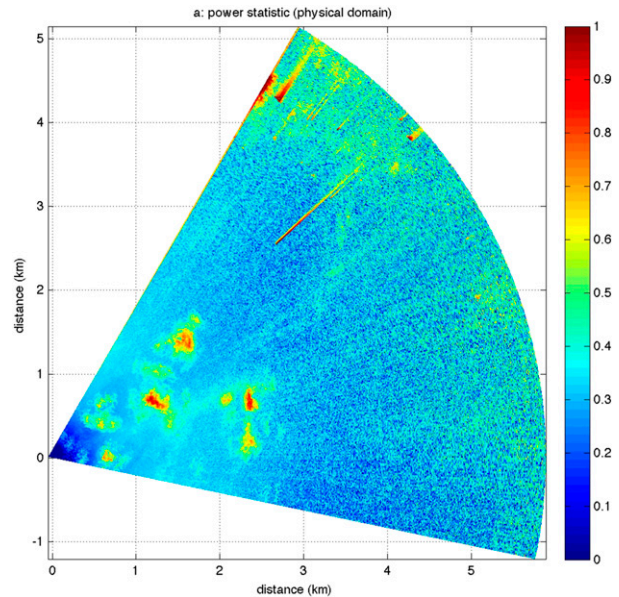


FIG. 4. Backscatter rescaled to the interval $[0,1]$ by a piecewise linear map.

noise, and hard targets. Consequently, the backscatter is a good measure for each of these features. However, if the backscatter is unnormalized, the scaling can change as a function of time and range and could be detrimental to the processing in both the physical and lag domains. A preprocessing (or normalization) algorithm was provided to the authors by the manufacturer of the lidar and by STAR LLC. The normalized backscatter data are referred to as the relative backscatter intensity (also referred to as simply backscatter, for the sake of space, in the figures). This quantity is shown in Figs. 1 and 3. Figure 3b is the lag domain for the relative backscatter intensity.

The relative backscatter is converted to a fuzzy statistic—that is, a quantity on the closed interval $[0,1]$ —by applying a piecewise linear function to the relative backscatter, where $f(y) = 0$ for $0 \leq y \leq 30$, $f(y) = (y - 30)/15$ for $30 < y < 45$, and $f(y) = 1$ for $45 \leq y$. The power statistic is defined as the arithmetic mean of the rescaled backscatter statistic for the present time and for the previous time $p_{i,j} = \{[f(y_{i,j}) + f(y'_{i,j})]/2\}$ and is shown in Fig. 4a, where a warm color indicates a value close to one and a cool color indicates a value close to zero. Figure 4 is very similar to Fig. 3a, but we now have a quantity whose value ranges between zero and one, which is needed when assigning a classification using fuzzy logic; that is, a decision is made regarding what classification to assign to a point based on score, so all the scores need to have similar scaling. Notice that the plumes and hard targets are bright relative to the noise

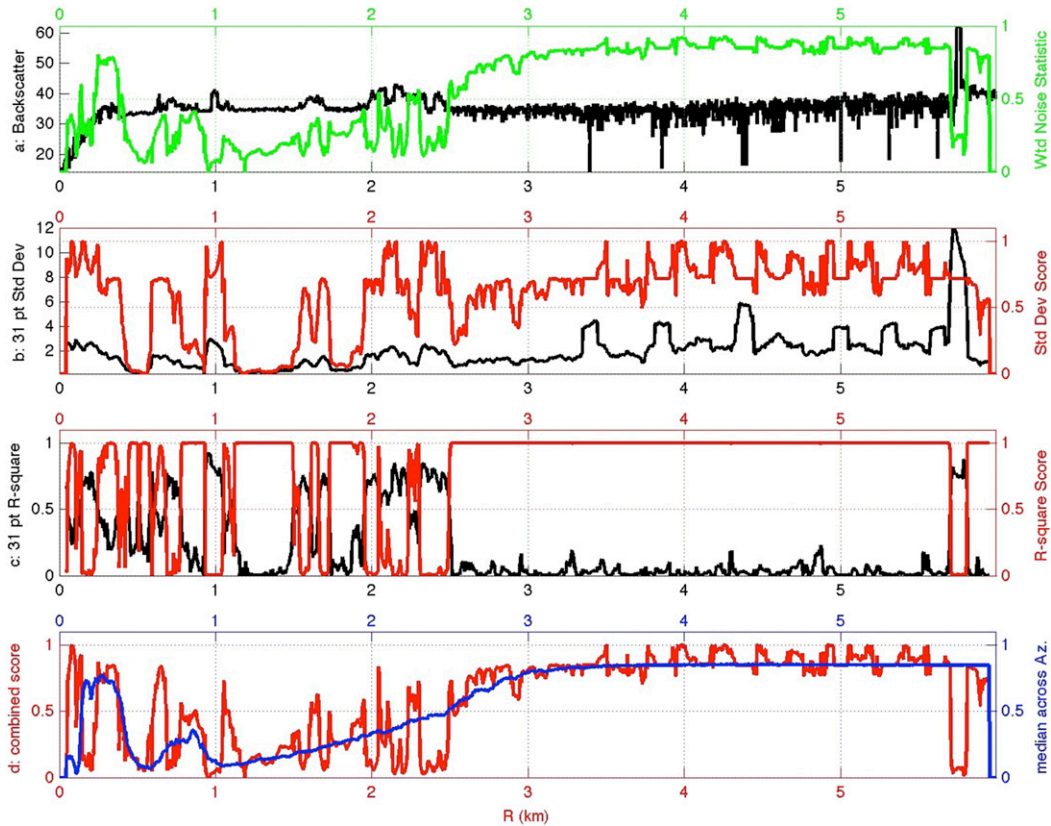


FIG. 5. Intermediate statistics and scores used to calculate the weighted noise statistic. (a) Backscatter as a function of distance (black) and the final weighed noise statistic (green). (b),(c) Intermediate statistics (black) and scores (red) after the application of a membership function (sigmoid). (d) Geometric mean of the scores from (b) and (c) (red) and the average noise score over an entire scan (blue).

and background, but the power statistic by itself does not separate plumes from hard targets; that is, the fuzzy power score for the hard targets found at distances greater than 3.5 km are similar to those in the brighter plumes found between 1 and 3 km, and hence further classification is required to distinguish plumes from hard targets.

5. The fuzzy noise statistic

As mentioned previously, a statistic is developed to find noise in the lidar image. Figure 5a (black curve) is a plot of a single shot of backscatter data from the lidar as a function of range. One can see from the plot that the noise in the black curve increases as a function of range [as expected as the signal-to-noise ratio decreases with range (r_i, θ_j)] and begins to appear more noticeable around 3 km. The green curve in Fig. 5a is the final noise statistic assigned to each point in this radial shot, and the intermediate statistics used to calculate the final noise statistic are shown in Figs. 5b–d. Our statistical measure

of noise is defined as data that have a large sample standard deviation (Miller and Miller 2003) over an interval and a small R -square value, both calculated over 31 range gates. The number of range gates was chosen to be large enough to calculate a meaningful statistic but small enough to resolve features in the data on a scale of about 100 m (the range gates are about 3 m in width). In general, noisy data are not well correlated. This can be quantified to some degree by the R -square statistic or the sample correlation coefficient (Miller and Miller 2003) between the two variables $y_{i,j}$ and $y_{i+1,j}$, where $y_{i,j}$ is the backscatter data at the point (r_i, θ_j) and $y_{i+1,j}$ is the backscatter data at the point (r_{i+1}, θ_j). In Fig. 5b the black curve shows the standard deviation calculated over the 31-point running window for the data shown in Fig. 5a (black curve). The red curve is a fuzzy score derived from the 31-point running standard deviation using a simple membership function (a high standard deviation is given a high noise score). Similarly, Fig. 5c shows the R -square statistic calculated over the 31-point running window (black) and a fuzzy score (red; a low

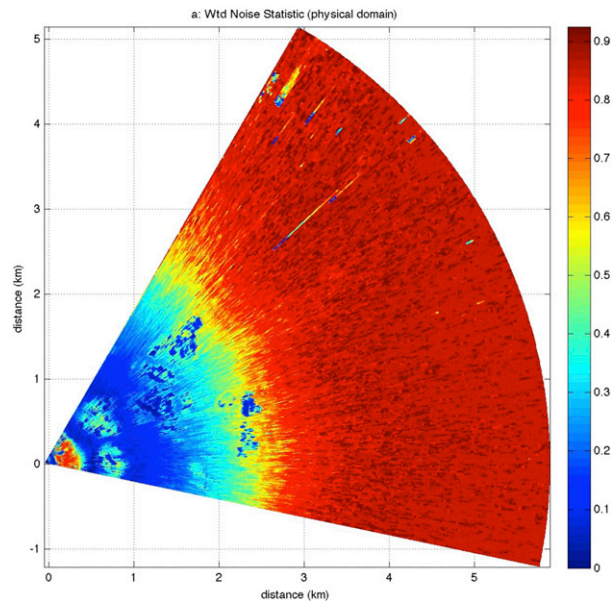


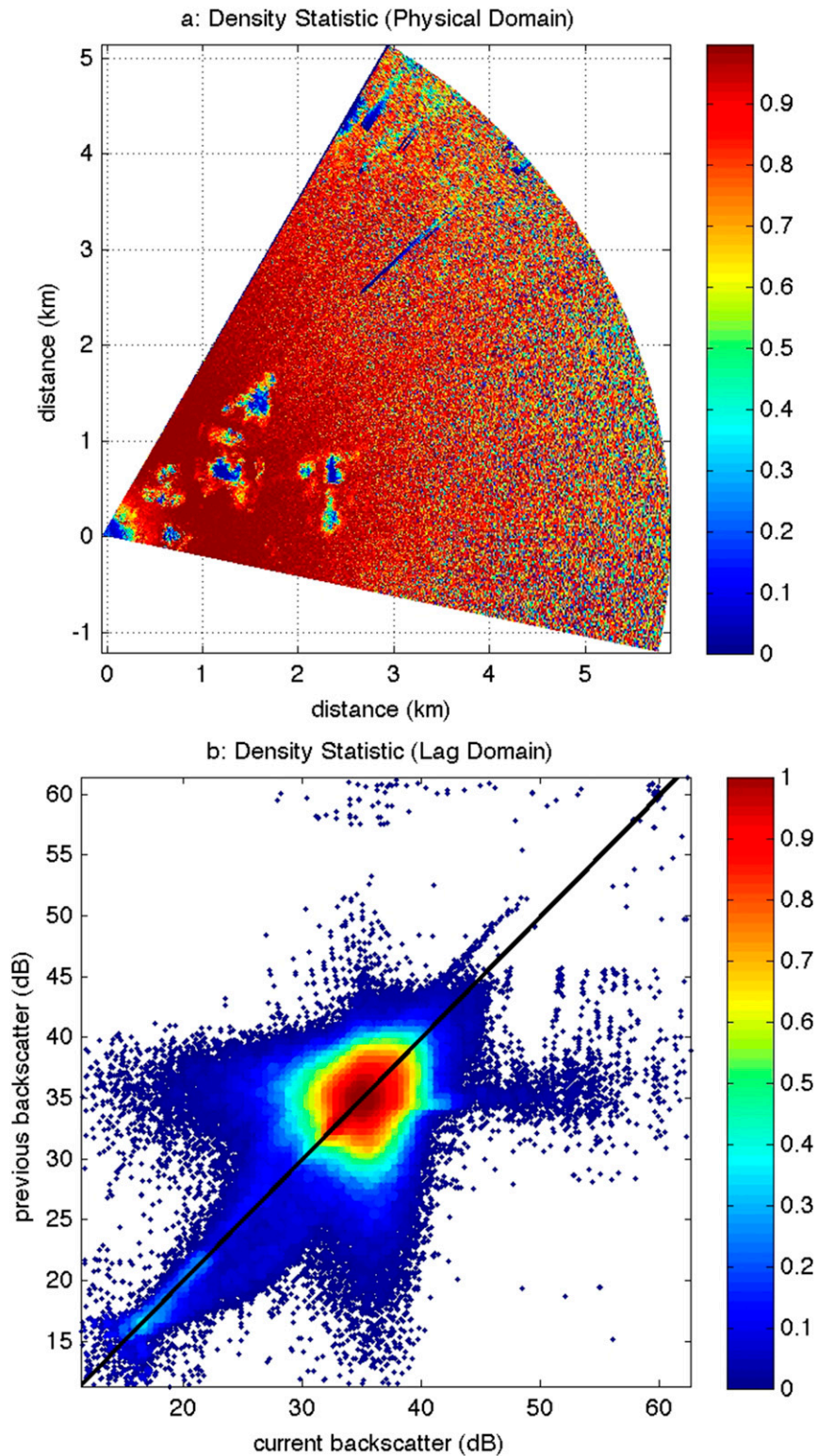
FIG. 6. Weighted noise score for a single lidar scan, where a warm color indicates a large noise statistic and a cool color indicates a low noise statistic.

R -square statistic is given a high score). In Fig. 5d, the red curve is the geometric mean of the two fuzzy scores from Figs. 5b and 5c, and the black curve is the median of this score over multiple azimuths. The geometric mean is used to mimic a “logical and” between the two scores. Noise tends not to be consistent over multiple azimuths, so the median value is taken to provide a more stable spatial quantity. The geometric mean was chosen because for the fuzzy noise score to be large, both the R -square fuzzy score and the standard deviation fuzzy score must be large; that is, at a fixed range in the shot, one assigns a fuzzy score to be the median fuzzy score (Fig. 5d, red) at all other points at that fixed range. Notice that the median fuzzy score over all azimuths has the desired properties; that is, the fuzzy noise score gradually grows as a function of range. The final fuzzy noise statistic ($n_{i,j}$), the green curve in Fig. 5a, is the geometric mean of the two functions shown in Fig. 5d. Observe the large noise statistic at the end of the dead zone region. Notice that the green curve in Fig. 5a tends to be smaller for the plumes and for the hard target. Although these features have large variances, they also have a smaller noise score because they tend to be more correlated than noise. Figure 6 shows the noise statistic plotted for an entire scan. Notice that the plumes get a low noise score, as do the near edges of the hard targets. This indicates that the noise statistic can also be used to separate noiselike points from plumes. Finally, notice the high noise values at the end of the dead zone region

near the origin and the transition to higher noise statistics at about 2.5 km.

6. The lag density statistic

Recall that the lag plots in Fig. 2 are created from successive lidar shots and the time difference between each shot is roughly 20 s. Over such a short time interval, one would expect the power statistic of background points not to change by a large amount. Also, from Fig. 1b it can be seen that the background is clustered around a power of 35 dB with a fairly low spatial variance, and one expects a low temporal variance as well because of the previous discussion. This results in a region of densely packed data around (35,35) in the lag space. This motivates the use of density calculated in the lag domain to find the background data in the physical domain. Density in the lag domain is estimated using a set of overlapping tiles with a fixed size and calculating the percentage of the data in each tile—that is, for a fixed tile count, the number of points ($r_i, \theta_j, y_{ij}, y'_{ij}$) in the lag space (also known as delay space) where (y_{ij}, y'_{ij}) is in this fixed tile, including possible repeats. To convert from counts to density, the total number of points in the lag space divides the count for a tile. This number is assigned to the center point of the given tile, and in this way, the density forms a gridded field. Next, linear interpolation is used to compute a lag density for every point (y_{ij}, y'_{ij}) in the lag domain. Since the densities can be small, we normalize each density by the largest density. This gives the density statistic $d_{i,j}$ scaled to the closed interval [0,1]. The choice of tile size depends on the quantization of the underlying backscatter data. In this case the tile size is 0.5 units [recall the backscatter is in arbitrary units (dB) after the preprocessing]. The idea is to set the bin size to be small enough to resolve detail but not so small that it contains too few data points. The lag density statistic (score) is shown in Fig. 7b. Notice the large density around (35,35). The statistic for the point (r_i, θ_j) in Fig. 7a is $d_{i,j}$ and is called the lag density statistic, where the statistic $d_{i,j}$ is transferred from the lag domain to the physical domain. Here a cool color represents a low density and a warm color represents a high density. Notice that the plumes, noise regions, and hard targets (Fig. 7a) have a lower density score than the background points. This is because the relative backscatter intensity of these point types changes more between scans than for the background points. In other words, the power in these regions is more spread out. Thus, these point types tend to be farther from the dense region in the lag domain and form less dense regions. The background points have the highest density. Since some noise points have high lag density, it is important



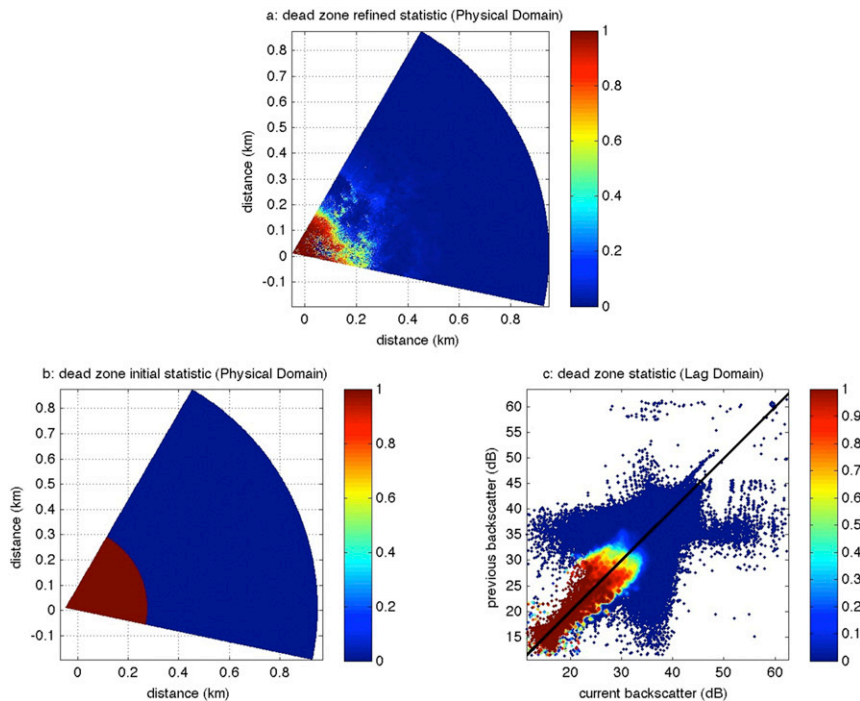


FIG. 8. (a) Dead zone statistic in the physical domain, where a warm color indicates a large value and a cool color indicates a low value. (b) Initial guess of the dead zone region. (c) Initial dead zone region in the lag domain.

to use the noise statistic to separate noise points from background points. The lag density statistic is adaptive since it depends only on the lag density, which is calculated from the present and previous scans.

7. The dead zone statistic

Recall the dead zone is a region of the backscatter waveform that begins at the origin and tapers off gradually due to geometrical overlap of the laser beam and the receiver field of view. In the REAL, full overlap should be achieved by the 500-m range. Notice that in Fig. 1a, the backscatter increases from zero until it reaches a value near 30 dB and then flattens out. In Fig. 3a, the dead zone appears as a small dark blue triangular wedge near the origin. Figure 8a shows the final dead zone statistic in the physical domain. The calculation of the dead zone score is performed in several steps. First, an initial statistic $I_{i,j}$ is computed by setting $I_{i,j} = 1$ if the point (r_i, θ_j) is within 0.325 km of the lidar; otherwise, it is set to zero. This is shown in Fig. 8b. The statistic $I_{i,j}$ is then assigned to the point $(r_i, \theta_j, y_{i,j}, y'_{i,j})$ in the lag space. Next, a density statistic is computed in the lag domain similar to the lag density statistic. The count is over all points in the lag space, where $(r_i, \theta_j, y_{i,j}, y'_{i,j})$ has an initial score of one and $(y_{i,j}, y'_{i,j})$ is in the tile. The

count of points in a tile includes repeats since the count is over all points $(r_i, \theta_j, y_{i,j}, y'_{i,j})$. This density is the dead zone statistic $dz_{i,j}$ for each point $(y_{i,j}, y'_{i,j})$ in the lag domain and is shown in Fig. 8c. Finally, the dead zone statistic is mapped back to the physical domain and the result of this process is shown in Fig. 8a. The interpretation of this statistic is that a point farther away from the lidar than 0.325 km may have a high dead zone statistic if the corresponding lag point has nearby dead zone points; that is, the receiver may not be able to process points near the dead zone, so points near dead zone points (in the lag domain) should also have a higher dead zone statistic. This is the only case where an initial statistic in the physical domain is transferred to the lag domain and then used to define a statistic in the physical domain.

8. Classification

A single classification is assigned to a data point (r_i, θ_j) once the individual scores are calculated for the point—specifically, power ($p_{i,j}$), dead zone ($dz_{i,j}$), lag density ($d_{i,j}$), and noise ($n_{i,j}$). To assign a classification to a data point, fuzzy maps are applied to the statistics to convert them into membership values. For instance, a plume membership value is calculated by applying a fuzzy map

to the power statistic. This map takes the shape of a logistics curve, which approaches one as the power statistic increases to one. The fuzzy score for the plume statistic is called the plume score, denoted p_{ij}^l , and is motivated by Fig. 4, where the plumes have large power values. Similar scores l_{ij}^l are computed for each of the classifications of noise, lag density, and dead zone. The score d_{ij}^l is called the background score and is motivated by reference to Fig. 7a, where it was shown that background points have a large lag density statistic. Each of these fuzzy maps has similar shapes, but they have different parameters. These parameters allow for a certain amount of tuning. Once these four scores have been computed, individual classifications may be calculated. The point (r_i, θ_j) is given the classification score $S_{ij} = \max(p_{ij}^l, n_{ij}^l, dz_{ij}^l, d_{ij}^l)$ and is given the classification C_{ij} , where the classification is that of the category where the maximum was attained. For example, if $S_{ij} = p_{ij}^l$, then the point (r_i, θ_j) is classified as a plume point. Similarly, if $S_{ij} = n_{ij}^l$, dz_{ij}^l , then the point (r_i, θ_j) is called a noise, or dead zone, point. In the case of a tie, pick the first index (as determined by the order of the scores in the max) where the maximum occurs. This strategy slightly favors selecting a plume classification. Every point in the image is classified as either a plume point, a noise point, a dead zone point, or a background point. The fuzzy values are thus transformed, or defuzzified, into classifications by categories; that is, a point is most like the classification that has the highest score.

Hard target points have yet to be classified. These points have a high power score, as do bright plumes. To distinguish between bright cluster points and hard target points, we look for clusters of points that make up hard targets. Specifically, clustering is used to assign a classification to the data inside the cluster. Hard targets, such as buildings and the shadow regions behind the buildings, persist in time. For example, consider the bright target in Fig. 3a and the elevated (in backscatter) streak behind this target in the upper-left-hand corner of the figure called a shadow region. Notice how the shadow feature is roughly aligned with the radial. The average relative backscatter over some previous scans is used to find these persistent features. In this paper 100 previous scans were used. Figure 9a (persistent statistic) shows the average relative backscatter for the 100 scans. The hard targets and shadow regions are clearly visible as warm colors. The bright region on the extreme upper-left edge of Fig. 4 does not appear in Fig. 9a, suggesting that this feature is not persistent in time and is likely a bright plume. Also, this possible bright plume does not contain a shadow region.

A logistic fuzzy membership function is applied to the persistence statistic to create the persistence score. An

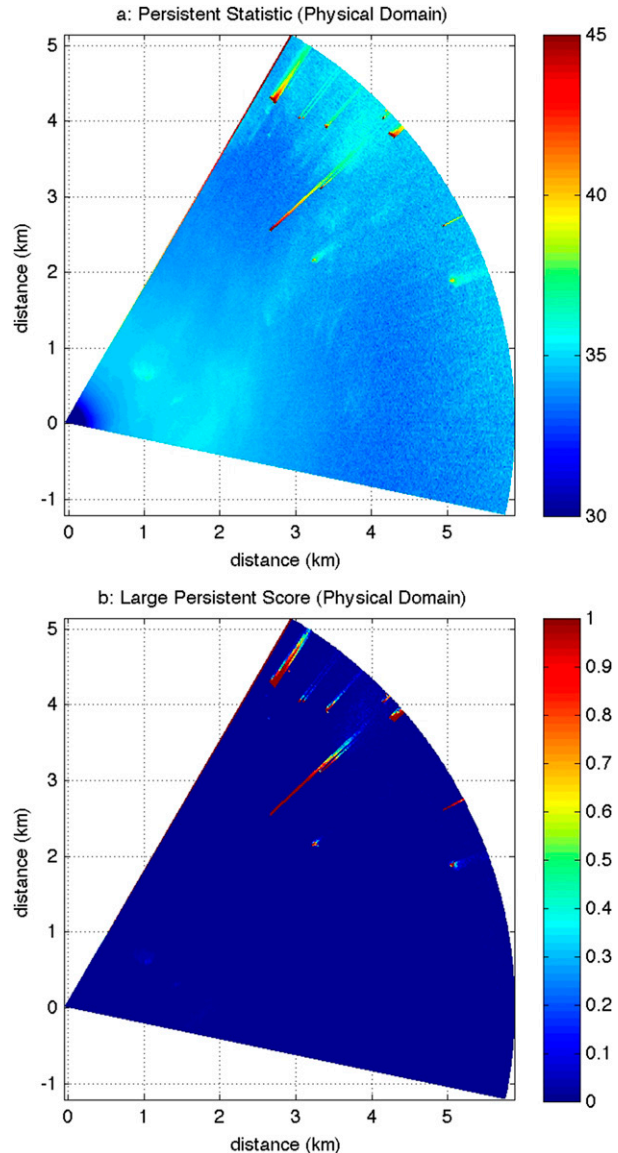


FIG. 9. (a) Persistence statistic, which is the average backscatter over 100 lidar scans. (b) Large persistence score after a sigmoid is applied to the persistence statistic.

example of this is shown in Fig. 9b. The persistence score helps to classify the hard targets and shadow regions. However, it could be that there are bright, persistent plumes from sources such as incinerators or heating plants that could be confused with hard targets. To perform this separation, mathematical morphology techniques are applied. A threshold is applied to the points in the persistence image (Fig. 9b) and clusters of points above this threshold (see Shih 2009) are formed.

It is the large persistence clusters that are aligned with radials that form the shadow regions. To detect these radial clusters, a kernel or structuring element is used

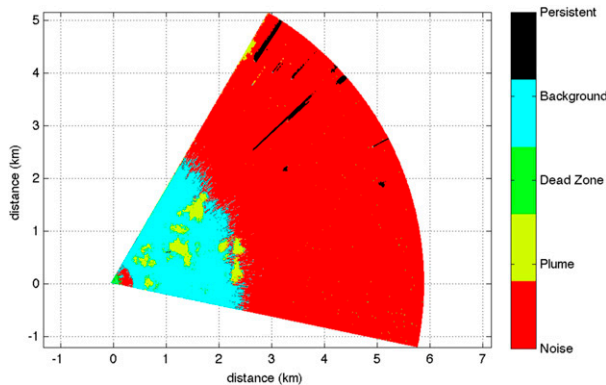


FIG. 10. Classifications for the entire lidar scan in the physical domain.

(Shih 2009). This kernel has the shape of six azimuths by 90 range gates. A thick persistence score is calculated by taking a morphological opening followed by a morphological closing with the structuring element and the large persistence score. Clusters are found in the large persistence score by finding the connected components above a threshold value (0.2). A thick score for each cluster is calculated as the average of the morphological thick score found previously. If enough data inside the cluster are *thick*, then the cluster is labeled as such; otherwise, it is identified as *thin*. All of the points in the thick and thin radial clusters are classified as persistent. The persistent classification given to these points in the cluster takes precedence over any previous classifications assigned to the data.

Figure 10 shows the classifications assigned to the data shown in Fig. 3a. Comparing Figs. 3a and 10, notice that the desired separation between hard targets (buildings and shadow regions) and plumes has now occurred. A database of hard targets and shadow regions could be constructed over time using the persistent classification information. The classification of the plumes roughly agrees with the classification a human might give by looking at Fig. 3a. The dead zone region is seen at the apex of the scan, and there is a noise region at the end of dead zone (see Figs. 1a and 1b). The background region is for the most part a solid region around the plumes, and there are some background points among the noise points at distances greater than 3 km. This is also seen in Fig. 1b. The impression given in Fig. 3a is also that some background points persist beyond 3 km but are swamped by noise because of weak backscatter return at larger ranges. Clusters of points that are plumes can be identified and not every point has to have a perfect classification in order to find these plumes. In Fig. 10, notice the cluster of points in the upper-left extreme edge has been classified as a plume and illustrates how bright (and

large) plumes may sometimes be detected even though the surrounding points are mostly noise points.

Figures 11a–d show the point classifications in the lag domain. Given the point $(r_i, \theta_j, y_{ij}, y'_{ij})$, if the point (r_i, θ_j) has classification C_{ij} , then the point (y_{ij}, y'_{ij}) will also have classification C_{ij} . A point (y_{ij}, y'_{ij}) could have more than one classification. Besides a point having more than one classification, there is also the problem that if two classification types are intermixed in the lag domain, the colors tend to run together when plotted. Figure 11 shows only one or two of the classifications at a time. Two classifications are plotted together only when they are separated in backscatter, and again that illustrates that backscatter alone does not separate these point types. In Fig. 11a both background and plume are plotted. Notice the background points remain close to the $y = x$ line because background points, in the cases studied, do not change much in relative backscatter from scan to scan. Some of the plume points are away from the line $y = x$ because these plumes are moving in time. Edge points of the plume may change quite a bit in time since the plume is moving, so background points could become a plume point or vice versa. It is also possible that some of the plume points away from the $y = x$ line are misclassified and might be noise and may have happened because the plume points were defined in terms of power only. Figure 11b shows the dead zone data; it can be seen that most of these points are near the line $y = x$. Figure 11c shows the noise points. Notice that since noise is not consistent over time, there are a large number of points far from the $y = x$ line. Notice that part of the noise region overlaps with the dead zone region. In this case both the dead zone data and the noise data have similar backscatter, which may arise from noise at the end of the dead zone region. Notice also, that noise points can have a wide backscatter range. This means that scores depending only on power could misclassify some points. Misclassification of noise is not a severe problem for finding plumes since noise points tend not to form coherent clusters with similar backscatter or power values. Figure 11d shows the persistent points in the lag domain. As expected, the persistence points have elevated power values. The points far from the $y = x$ line could be misclassified noise points. Clusters of data classified as persistent tell us the locations of suspect data such as hard targets and shadow regions, and are regions where the backscatter is not from the true atmosphere. These figures suggest a change in classification may be needed if a point is far from the $y = x$ line and has low density in the lag domain. For instance, one could consider changing the classification to noise. However, the reclassification of the persistent

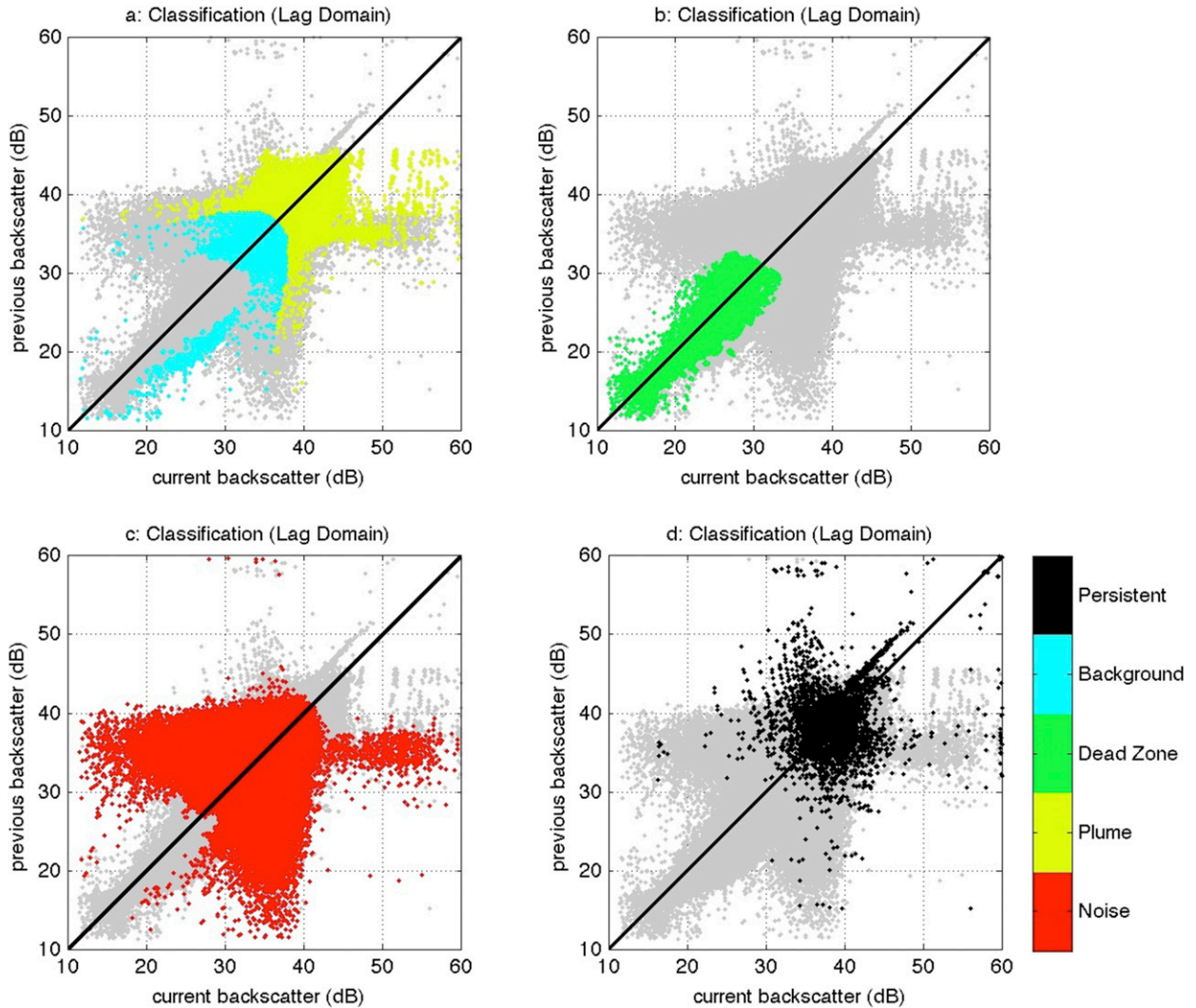


FIG. 11. Classifications for the entire lidar scan in the lag domain; gray data points indicate the scatterplot of all the data in the lag domain, and the classified data are shown in color.

data has not been done since these problematic points appear to have low impact on finding plume clusters.

9. Tuning and validation

To validate the algorithm described above, plumes were identified in multiple PPI scans through a human truth effort. A bounding box was drawn around a plume of interest and a threshold was set; points above the threshold and inside the box were marked as plumes. A similar mask was created for the results of the algorithm; specifically, points that were identified as plume were assigned a value of one and all other classifications were given a value of zero.

The entire dataset consisted of 142 scans, 100 scans of which were used to build the persistent target score,

leaving 42 scans for truth. These remaining 42 scans were split roughly evenly before and after the 100 scans used for the persistent score. The early scans, numbered 1–21, were mostly clear with only a few small plumes in near-range gates—specifically, range gates less than 2.5 km (the noise region started roughly at 2.5 km). The second group of scans used in the human truth effort numbered 122–142 and were much larger, and extended much further in range. In all the scans truthed, plumes were not identified in regions where noise persisted to avoid contaminating the statistics with noisy data.

A single scan of human-truthed data was used to further tune the algorithm by finding the best set of parameters for the plume and background fuzzy membership functions. The best set of parameters was found by creating a coarse mesh for the membership function

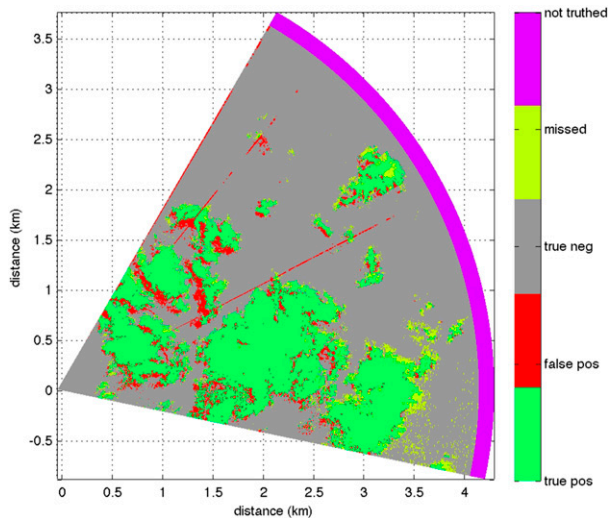


FIG. 12. Plot of skill metrics: true positive (green), false positive (red), true negative (gray), missed detections (yellow), and data outside of the truth dataset (purple). The leading and trailing edges of the plume correspond to the false positive and missed detection values (along with regions that were misidentified as plume data). Notice that edge values and a few radial features have been mislabeled as plume data.

parameters, running the algorithm for each parameter setting, calculating a plume mask for the algorithm, and finding the absolute difference between the algorithm mask and the human truth mask. To test that the algorithm was not overtrained, an early scan, which was mostly clear with just a few small plumes, was evaluated. It was found that the algorithm did not overclassify plume regions in this case. Figure 12 shows the skill metrics for a single scan of data—specifically, the true positive data (green), false alarms (red), true negative (gray), and missed detections (yellow). Notice that the leading and trailing edges of the plumes are classified as either missed detections or false positive values and is caused by the plume moving between scans. Additionally, there are a few radials that are misidentified as plumes by the algorithm. These misidentified plumes were caused by what is called spoking.² The REAL data in particular exhibit this characteristic. It is due to the difficulty of making high-precision measurements of laser pulse energy. Scanning micropulse lidar systems appear to be less prone to this idiosyncrasy (Mayor et al. 2015). In general, the algorithm performs well in matching how a human characterized the data. Notice there is some disperse data that are missed by the

² *Spoking* refers to an artifact in the lidar data that is caused by shot-to-shot fluctuations in the laser pulse energy that result in what appears as radial streaks in the image data.

algorithm (yellow) near 4 km, which is a result of the method used to truth the data. These were in fact incorrectly identified as plumes by the human truth effort. A total of 1 355 285 pixels were identified as true positives in the human truth scans, 320 921 pixels were identified as false positives, 8 522 371 pixels were identified as true negatives, and 267 638 pixels were identified as missed detections. Using these values yields a true skill score of 0.798.

10. Future work

The current dataset used in both the development and evaluation of the algorithm was limited. Future work should include expanding the cases evaluated to more thoroughly test the assumptions and performance of the algorithm and to identify examples cases to improve the algorithm. Such cases should include transient hard targets, such as insects and wires. Tuning the algorithm should use more sophisticated optimization techniques and more data to improve the estimation of the parameters in the membership maps. To this point, synthetic data might be used to train the algorithm and further develop and refine the detection capabilities of the algorithm; for instance, Hamada et al. (2016) made use of synthetic aerosol data in testing their algorithm. Future work might also include the addition of a tracking algorithm (Dérian et al. 2015; Hamada et al. 2016). A tracking algorithm would help detect plumes that are moving with the wind at a high rate of speed and improve the detection of plume edges and might help in estimating the ambient wind. To address issues with leading and trailing edges of plumes, it would be useful to include geometrical information such as lag spaces created by shifting the given present image in azimuth and radial directions. The background classification should be used in the preprocessing of the data (refer to the appendix) rather than making assumptions about the data—specifically, what data belongs to the background. Also, lag space techniques should be used to identify radials with elevated power relative to the immediate neighbors in azimuth (previously termed spoking).

All of these techniques should be applicable to a wide range of devices such as radars, wind profilers, and sodars. These methods could also be applicable to medical imaging devices, such as computerized axial tomography (CAT) scans, magnetic resonance imaging (MRI), and ultrasound images. The authors have done some studies applying these techniques to analyze satellite, radar, and CAT images. It may be that lag and other fuzzy scores may prove to be useful feature vectors in machine learning algorithms for image segmentation, and object classification.

11. Conclusions

The goal of classifying lidar backscatter data using scores in the physical and lag spaces has been achieved. Scores for multiple classes of data found were developed as part of the classification scheme. Specifically, scores for background, persistent features, plumes, noise, and the dead zone were developed. An important use of the lag domain was to compute the lag density of points, which was used in the classification of background points. The definition of shadow regions used techniques from morphology and was done in the physical domain. The plume score was based only on backscatter in the physical domain. The noise score was defined in the physical space by statistical means. Both the lag domain and physical domain was used to score the points in the dead zone; both the lag and the physical domains are useful in classifying points in the physical domain. These classifications are generated for each scan and are used to identify the plumes. The algorithm not only classifies plumes, but it locates hard targets and shadow regions behind hard targets. In addition, the algorithm may be used to identify persistent plumes coming from sources such as incinerators and power plants.

Acknowledgments. The authors acknowledge the funding provided by the National Center for Atmospheric Research and STAR LLC to conduct and publish this research. We acknowledge the support provided by the National Renewable Energy Laboratory to publish this work. We thank the reviewers for the many useful editorial suggestions. We also thank one of the reviewers for sharing technical information about the REAL.

APPENDIX

Calculation of Relative Backscatter Power

The steps to calculate the relative backscatter power are as follows: 1) power normalization. The data in each shot are divided by a normalization factor provided by the lidar for each shot. This factor accounts for the fact that the power in each shot is not constant. 2) Range correction. The data value is multiplied by the square of the range because the backscatter drops off by the inverse square law. This keeps the backscatter in a comparable range throughout the image. 3) Reduce dynamic range. This is to make the data look more like radar data and reduce the dynamic range of the data. Similar to radar data, the backscatter Z is replaced by $Y = 10\log_{10}(Z)$ and is not allowed to be below a set

minimum value. 4) To further mitigate variations in transmitted laser pulse energy, the average of the data of a set of predetermined range gates is computed, subtracted from the data, and then a predetermined constant is added to the data. The predetermined range gates are set outside of the dead zone region but well away from the noise region. It is assumed that these predetermined range gates contain only background points. Steps 1, 2, and 4 mitigate the problem of spoking, where spoking is a radial with large power relative to its immediate neighbors in azimuth.

REFERENCES

- Airouche, M., L. Bentabet, and M. Zelmat, 2009: Image segmentation using active contour model and level set method applied to detect oil spills. *Proceedings of the World Congress on Engineering*, S. I. Ao et al., Eds., Lecture Notes in Engineering and Computer Science, Vol. 2176, Newswood Limited, 846–850.
- Ben-David, A., C. E. Davidson, and R. G. Vanderbeek, 2007: Lidar detection algorithm for time and range anomalies. *Appl. Opt.*, **46**, 7275–7288, doi:10.1364/AO.46.007275.
- Blackledge, J. M., and M. J. Turner, Eds., 2001: *Image Processing III: Mathematical Methods, Algorithms and Applications*. Horwood Publishing, 298 pp.
- Chi, Z., H. Yan, and T. Pham, 1996: *Fuzzy Algorithms: With Applications to Image Processing and Pattern Recognition*. Advances in Fuzzy Systems—Applications and Theory, Vol. 10, World Scientific, 225 pp.
- Cornman, L. B., R. K. Goodrich, C. S. Morse, and W. L. Ecklund, 1998: A fuzzy logic method for improved moment estimation from Doppler spectra. *J. Atmos. Oceanic Technol.*, **15**, 1287–1305, doi:10.1175/1520-0426(1998)015<1287:AFLMFI>2.0.CO;2.
- Delanoy, R. L., and S. W. Troxel, 1993: Automated gust front detection using knowledge-based signal processing. *Record of the 1993 IEEE National Radar Conference*, IEEE, 150–155, doi:10.1109/NRC.1993.270475.
- Dérian, P., C. F. Mauzey, and S. D. Mayor, 2015: Wavelet-based optical flow for two-component wind field estimation from single aerosol lidar data. *J. Atmos. Oceanic Technol.*, **32**, 1759–1778, doi:10.1175/JTECH-D-15-0010.1.
- Hamada, M., P. Dérian, C. F. Mauzey, and S. D. Mayor, 2016: Optimization of the cross-correlation algorithm for two-component wind field estimation from single aerosol lidar data and comparison with Doppler lidar. *J. Atmos. Oceanic Technol.*, **33**, 81–101, doi:10.1175/JTECH-D-15-0009.1.
- Kettaf, F. Z., D. Bi, and J. P. Asselin de Beauville, 1996: A comparison study of image segmentation by clustering techniques. *1996 Third International Conference on Signal Processing: Proceedings*, B. Yuan and X. Tang, Eds., Vol. 2, IEEE, 1280–1283, doi:10.1109/ICSPGP.1996.566528.
- Lenschow, D. H., V. Wulfmeyer, and C. Senff, 2000: Measuring second- through fourth-order moments in noisy data. *J. Atmos. Oceanic Technol.*, **17**, 1330–1347, doi:10.1175/1520-0426(2000)017<1330:MSTFOM>2.0.CO;2.
- Ma, J., and S. Perkins, 2003: Time-series novelty detection using one-class support vector machines. *Proceedings of the International Joint Conference on Neural Networks, 2003*, Vol. 3, IEEE, 1741–1745, doi:10.1109/IJCNN.2003.1223670.

- Mayor, S. D., and S. M. Spuler, 2004: Raman-shifted eye-safe aerosol lidar. *Appl. Opt.*, **43**, 3915–3924, doi:[10.1364/AO.43.003915](https://doi.org/10.1364/AO.43.003915).
- , —, B. M. Morley, and E. Loew, 2007: Polarization lidar at 1.54 μm and observations of plumes from aerosol generators. *Opt. Eng.*, **46**, 096201, doi:[10.1117/1.2786406](https://doi.org/10.1117/1.2786406).
- , P. Dérian, C. F. Mauzey, S. M. Spuler, P. Ponsardin, J. Pruit, D. Ramsey, and N. S. Higdon, 2015: Comparison of aerosol backscatter and wind field estimates from the REAL and the SAMPLE. *Lidar Remote Sensing for Environmental Monitoring XV*, U. N. Singh, Ed., International Society for Optical Engineering (SPIE Proceedings, Vol. 9612), 96120G, doi:[10.1117/12.2187538](https://doi.org/10.1117/12.2187538).
- Miller, I., and M. Miller, 2003: *John E. Freund's Mathematical Statistics with Application*. 7th ed. Prentice Hall, 624 pp.
- Morse, C. S., R. K. Goodrich, and L. B. Cornman, 2002: The NIMA method for improved moment estimation from Doppler spectra. *J. Atmos. Oceanic Technol.*, **19**, 274–295, doi:[10.1175/1520-0426-19.3.274](https://doi.org/10.1175/1520-0426-19.3.274).
- Nachtegaal, M., D. van der Weken, E. E. Kerre, and W. Philips, Eds., 2007: *Soft Computing in Image Processing: Recent Advances*. Studies in Fuzziness and Soft Computing, Vol. 210, Springer, 487 pp.
- Rosenstein, M. T., and P. R. Cohen, 1998: Concepts from time series. *Proceedings of the 15th National Conference on Artificial Intelligence*, AAI Press, 739–745.
- Sasano, Y., H. Hirohara, T. Yamasaki, H. Shimizu, N. Takeuchi, and T. Kawamura, 1982: Horizontal wind vector determination from the displacement of aerosol distribution patterns observed by a scanning lidar. *J. Appl. Meteor.*, **21**, 1516–1523, doi:[10.1175/1520-0450\(1982\)021<1516:HWVDFT>2.0.CO;2](https://doi.org/10.1175/1520-0450(1982)021<1516:HWVDFT>2.0.CO;2).
- Shih, F. Y., 2009: *Image Processing and Mathematical Morphology: Fundamentals and Applications*. CRC Press, 439 pp.
- Weekley, R. A., R. K. Goodrich, and L. B. Cornman, 2003: Fuzzy image processing applied to time series analysis. *Third Conf. on Artificial Intelligence Applications to the Environmental Science*, Long Beach, CA, Amer. Meteor. Soc., 4.3. [Available online at https://ams.confex.com/ams/annual2003/techprogram/paper_55981.htm.]
- , —, and —, 2004: Feature classification for time series data. U.S. Patent 6,735,550, filed 15 January 2002, and issued 11 May 2004.
- , —, and —, 2010: An algorithm for classification and outlier detection of time-series data. *J. Atmos. Oceanic Technol.*, **27**, 94–107, doi:[10.1175/2009JTECHA1299.1](https://doi.org/10.1175/2009JTECHA1299.1).
- Wishart, D., 1969: Mode analysis: A generalisation of nearest neighbour which reduces chaining effects. *Numerical Taxonomy: Proceedings of the Colloquium in Numerical Taxonomy Held in the University of St. Andrews, September 1968*, A. J. Cole, Ed., Academic Press, 282–311.
- Zadeh, L. A., 1965: Fuzzy sets. *Inf. Control*, **8**, 338–353, doi:[10.1016/S0019-9958\(65\)90241-X](https://doi.org/10.1016/S0019-9958(65)90241-X).