Combined range ambiguity resolution and noise reduction in lidar signal processing

Gunnar Arisholm
Torbjørn Skauli
Stig Landrø

Combined range ambiguity resolution and noise reduction in lidar signal processing

Gunnar Arisholm,* Torbjørn Skauli, and Stig Landrø
FFI (Norwegian Defence Research Establishment), Kjeller, Norway

Abstract. When the interval between transmitted pulses is shorter than the time of flight, it is not straightforward for a lidar to determine the distance uniquely. We present a method that uses varying pulse intervals, constructs a set of tentative point positions for each received pulse, and exploits the clustering of such tentative points to determine the correct distance with high probability. The clustering of true points also helps suppress noise pulses, even in a regime where the number of noise pulses is far greater than the number of true return pulses. It is then possible to use a detection threshold close to the noise level. © The Authors. Published by SPIE under a Creative Commons Attribution 3.0 Unported License. Distribution or reproduction of this work in whole or in part requires full attribution of the original publication, including its DOI. [DOI: 10.1117/1.OE.57.7.073103]

1 Introduction

An imaging lidar (light detection and ranging) system based on scanning a single beam needs to operate at high pulse rate to record a point cloud quickly. However, if the lidar transmits a new pulse before the last return pulse from the previously transmitted pulses has arrived, the range measurements will not be unique. The signal processing system has to make some assumption to associate a return pulse with one of the transmitted pulses and compute the correct distance. One solution, which can work well for airborne sensors, is to use a digital elevation model to obtain the approximate distance.[1] Another solution is to label the transmitted pulses. This can be done by a sequence of micropulses within each macropulse,[2] or, in coherent systems, by frequency stepping[3] or phase modulation.[4] A third solution is to transmit pulses with varying intervals between them. This principle has been employed in radar (Ref. 5, Chapter 15 and Ref. 6, Chapter 4.4) and in many photon-counting lidar systems.[7–9] These lidars transmit multiple pulses for each point and apply time correlation to find the correct distance. Varying pulse intervals has also been used in classical lidars, which transmit a single, relatively high-power, pulse for each direction,[10,11] but the method was restricted to a single return pulse for each transmitted pulse. That restriction may be acceptable in the context of airborne laser scanning, but in other cases, one may be interested in detecting multiple return pulses from a single transmitted pulse. Furthermore, in order to maximize the detection range, it is desirable to set the detection threshold to a low level, where there will also be some false detections. In this paper, we describe a processing method that combines range ambiguity resolution and noise reduction and also allows multiple return pulses from each transmitted pulse. The method is not constrained to a specific application or range of spatial resolutions, so it should be applicable for a wide range of single-beam lidars that operate with more than one pulse in the air.
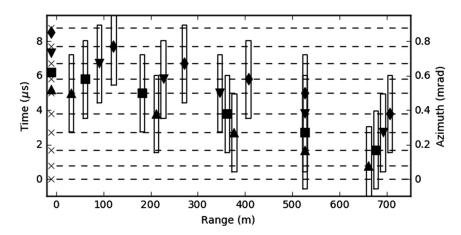
2 Method

The process to extract a point cloud from a detector signal consists of three main stages: filtering, pulse detection, and point detection. Filtering handles saturated pulses and reduces noise. Pulse detection finds all the pulses in the filtered signal that exceed a threshold, and produces a list with time and power for each detected pulse. The filtering and/or pulse detection stages can also include measures to suppress strong backscatter from common optics or from the atmosphere close to the lidar, which can dominate the received signal immediately after each transmitted pulse. The topic of this paper is the point detection stage, which takes as input of the list of detected pulses and the list of times and directions for the transmitted pulses, and produces a point cloud as output.

As in former work,[7–11] our proposed point detector is based on varying the interval between transmitted pulses. For each received pulse, it constructs a set of point candidates by associating the pulse with each of the N most recently transmitted pulses, for some N determined by the maximum likely delay of a returned pulse. In practice, N may be limited by the scan speed and the receiver field of view. A point candidate is represented by its spherical coordinates (r, θ, ϕ), where r is the range and the angles θ (pitch) and ϕ (azimuth) correspond to the direction of the transmitted pulse. (The reason for using the transmitted direction is that the divergence of the transmitted beam is typically smaller than the field of view of the receiver, so the transmitted beam defines the direction more precisely.)

Point candidates belonging to real objects tend to cluster in space, whereas false point candidates tend to be scattered because of the uneven pulse intervals. Therefore, each point candidate is assigned a figure of merit (FOM) that depends on the other point candidates in its neighborhood. In the simplest form, which is used in most of the examples, the FOM is just the number of point candidates found in a

*Address all correspondence to: Gunnar Arisholm, E-mail: gunnar.arisholm@ffi.no

Fig. 1 Distribution of point candidates in the range-azimuth plane. The crosses along the time axis indicate the transmitted pulses. They have intervals of 0.8, 0.9, 1.0, 1.1, and 1.2 μs, giving a period time of 5 μs and a 750-m range where unambiguous distances can be determined. The other symbols along the time axis indicate the times when four return pulses are received. They are marked by triangle pointing up, square, triangle pointing down, and diamond, in chronological order. The same symbols are used to indicate the range and azimuth of the point candidates when each received pulse is associated with the last transmitted pulse, second last transmitted pulse, and so on up to the fifth. The box around a point candidate shows the neighborhood where other point candidates can contribute to its FOM, 0.45 mrad by 10 m in this example. The neighborhood boxes overlap only for the point candidates at the correct distance of 526 m. The azimuth axis corresponds to the direction of the transmitted pulses. In this example, the relation between time and azimuth is simple because the scan speed is set to be fixed at 100 rad/s.

neighborhood in the (r, θ, ϕ)-space, chosen as described below. A more general FOM measure, which takes the power of the pulses into account, is discussed in Sec. 5.

After computing the FOM values, the algorithm selects the point candidate with the highest FOM. This is taken to represent the correct position of a return pulse, so it is added to the output point cloud. All other point candidates arising from the same pulse are removed and so are their contributions to the FOM of neighbor points. The algorithm then selects the next point candidate with the highest FOM, and so on until no point candidate has FOM above some threshold value. This FOM threshold reflects the assumption that a point candidate with no or few neighbors is probably noise.

Figure 1 shows an example of how the point candidates corresponding to four consecutive return pulses are distributed in the range-azimuth plane. The only place where the neighborhood-boxes of multiple point candidates overlap is at the correct range of 526 m, corresponding to the return pulse delay of 3.51 μs. These points increase the FOM of each other so the algorithm can resolve the ambiguity. In practice, it is of course possible that some of the return pulses fail to be detected. This loss can be compensated using a greater number of pulses, but the figure shows only four for the sake of clarity.

Because the method is based on finding clusters of point candidates, an object too small to give rise to a cluster will not be detected. Furthermore, if the shape and orientation of an object is such that the lidar sees it as separate parts, then only sufficiently large parts will be detected. The number of point candidates required for detection depends on the parameter settings. If the amount of noise is small, two correct point candidates will be sufficient to determine the range correctly.

3 Choice of Parameters

We denote the pulse intervals $T_i = s_i \Delta t$, where s_i are integers, $0 \leq i < n$, and Δt is the chosen minimum difference

between pulse intervals. The sequence of pulse intervals is repeated periodically so that $s_{i+n} = s_i$, and we call the periodic unit of n pulses a pulse group. The pulse intervals should be chosen so false point candidates have minimal probability of forming clusters. For this purpose, it is not enough that each s_i is unique; sums of m adjacent intervals must also be unique for $1 \leq m < n$. The sums can cross pulse groups, so the requirement can be stated as

$$S_{m,j} = \sum_{i=j}^{j+m-1} s_i, \tag{1}$$

being unique for all $1 \leq m < n$ and $0 \leq j < n$.

For small n, it is easy to find sequences satisfying this condition by trial and error. If n is prime, it can be done more rigorously by taking $s_i = i + k$ for $0 \leq i < n$ and fixed k. This ensures that $S_{m,j} \neq S_{m,j'}$ for $j \neq j'$ [because $S_{m,j} - S_{m,j'} \equiv m(j - j') \pmod{n}$]. To make the sums unique for different m, the constant k can simply be taken large enough to make $S_{m+1,j} > S_{m,i}$ for all i, j.

For the method to be able to determine range uniquely, the duration of a pulse group, $\Delta t \sum_{i=0}^{n-1} s_i$, should be greater than the maximum delay of a returned pulse. It is not straightforward to define a maximum range based on the transmitter power and receiver sensitivity, because a strong reflector somewhere in the scene can return a detectable pulse from a distance much greater than the normal operating range of the lidar. However, the maximum delay is eventually limited by the time it takes to scan the receiver field of view past the direction of a returned pulse.

The choice of Δt is a trade-off between performance of the laser and the point detection algorithm. A pulsed laser typically has a limited range of pulse rates where it can operate, and the pulse rate affects the energy and possibly other properties of the pulses. Therefore, the difference between the shortest and longest pulse interval cannot be too large,

and this places an upper limit on $[\max(s_i) - \min(s_i)]\Delta t$. On the other hand, an object will give rise to both correct and incorrect point candidates, and to avoid that the latter fall close enough to reinforce the FOM of each other we must have

$$\min_{m \neq m', j \neq j'} |S_{m,j} - S_{m',j'}| t > 2\Delta z/c, \qquad (2)$$

where Δz is the size of the point neighborhood in the range direction and c is the speed of light. Δz must be greater than the typical range difference between adjacent return pulses from the same object, otherwise they will not be identified as neighbors. The range difference between adjacent return pulses depends on the scan angle increment, the range, and the shape and orientation of the object. It is not possible to account for any shape and orientation of an object, but the important point is that Δz is large enough for a sufficient number of return pulses to be classified as neighbors. This is shown in Fig. 2.

There is no hard limit for the neighborhood size in the angle directions, but some guidelines can be found based on object size and density of noise points. If the probability of a detecting a noise point (after suitable processing of the raw signal) is independent of time, the number of noise points within the neighborhood of a point candidate will be approximately Poisson-distributed with mean proportional to the neighborhood volume. Let N_n be a random variable representing the number of such noise points. The mean $\langle N_n \rangle = \Omega \Delta z A$, where Ω is the solid angle of the neighborhood, Δz is its size in the range dimension, and A is a constant characterizing the noise. Let N_b be another random variable representing the number of detected object points, in the same solid angle Ω, from an object large enough to fill this solid angle. We do not assume a specific distribution for N_b, but we assume that the mean value $\langle N_b \rangle \propto \Omega$. $\langle N_b \rangle$ also depends on the distance and on the reflection properties of the object. The quantity that the point detection algorithm actually works with is the total number of point candidates in the neighborhood, $N_t = N_b + N_n$.

For mathematical simplicity, we now take the FOM to be the number of point candidates in the neighborhood. For the FOM threshold T to distinguish between noise points and true points, it should be set so that $P(N_n >= T)$ is small and $P(N_t >= T)$ is large when a real object is present. We define the signal-to-noise ratio (SNR) of the FOM as

$$\frac{\langle N_b \rangle}{\sigma(N_n)} \propto \frac{\Omega}{\sqrt{\Omega \Delta z}} \propto \Omega^{1/2}, \qquad (3)$$
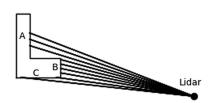

Fig. 2 Lidar imaging of an L-shaped object oriented so it hides parts of itself. Adjacent return pulses from the faces A or B will have small range difference, but if A and B are separated by more than Δz they will have to be detected independently, without contributing to each other FOM. On face C, the range difference between adjacent pulses is large because of the large angle of incidence. Faces with such slant orientation are difficult to detect, but in practice, most objects also present some faces with more favorable orientations.

where $\sigma(N_n)$ denotes the standard deviation of N_n. The equation shows that Ω should be large to maximize the SNR, but this only holds as long as the object fills this solid angle. Even a small object can fill a neighborhood at close range, but since close objects are likely to be detected anyway, the neighborhood size should be chosen to optimize detection of typical objects of interest at the maximum operating range of the lidar.

To set the FOM threshold automatically, we estimate the density of noise points $\langle N_n \rangle$ and choose the threshold to make the probability of accepting noise points suitably small. Even if the field of view is completely filled by objects, they will give rise to return pulses from only a small fraction of the scanned volume, because the objects are localized in the range direction. Thus, if we divide the volume into small cells, we can assume that most of them will be empty or contain only noise points, i.e., $N_t = N_n$ for these cells. We assume, conservatively, that the 80% of the cells with fewest point candidates do not contain real objects and estimate $\langle N_n \rangle$ by fitting a Poisson distribution to this tail of the distribution of N_t. The approximation here is that a few of these cells may in fact contain real return pulses, and end up in the lower tail because they happen to contain few noise pulses. If the noise level is low, it is possible that most of the point candidates arise from real objects, and that the lower tail of the N_t distribution is all zero. In this case, we cannot estimate $\langle N_n \rangle$ by fitting, so instead we find an upper bound for it, based on the observed probability of finding zero points in a cell. Once $\langle N_n \rangle$ has been estimated, the cumulative distribution function can be used to set the FOM threshold, T_F, so the probability of accepting a noise point has a desired value e

$$P(N_n > T_F) \approx e. \qquad (4)$$

In other words, e is the probability that a noise point has enough noise neighbors to bring its FOM above T_F. Because of the approximations in the distribution and in the estimate of $\langle N_n \rangle$, this error probability cannot be taken literally, but it is a useful way to parameterize the FOM threshold. In the examples in the next section, we set $e = 10^{-5}$. Some experiments with a more general FOM measure are described in Sec. 5.

Although the threshold for pulse detection is not a parameter to the point detector, it must be taken into account because it effectively determines the number of point candidates. If the objects of interest are known to have high SNR (defined as the ratio of return pulse amplitude to RMS noise), the detection threshold can be set so high that noise points are eliminated, and the task of the point detector is reduced to solving the range ambiguity. In the more interesting case, where it is desirable to maximize the range for a given transmitted power, the detection threshold must be set to a low value, where many noise pulses will appear. A method for automatic setting should not make any assumption about the number of true return pulses in the scene. However, as explained above in the context of setting the FOM threshold, true return pulses can be expected to appear in only a small part of the scanned volume. Thus, if the detection threshold is set to a value such that most of the scanned volume is filled by a relatively uniform density of point candidates, then these can be assumed to represent noise. A large number of noise points increase the run time of the point

detector, so the actual setting of the detection threshold is a trade-off between sensitivity and run time. A useful parameter is the number of detected pulses divided by the number of transmitted pulses. Various detection thresholds, giving up to five detected pulses per transmitted pulse, are tested in the examples.

For comparison, we note that lidars that transmit multiple pulses for each point[7–9] often use a pseudorandom pulse sequence and correlation detection. Our reason for not adopting that approach is that our method is designed for lidars that transmit a single pulse for each point, and when the beam is scanned the range difference between adjacent pulses will often be too large for direct correlation with the return signal to work well. The pulse sequence described above is designed for minimum ambiguity in realistic scenes, whereas a random or pseudorandom sequence might be suboptimal.

4 Examples

Although the method has been used successfully in tests on real data, we prefer to use simulated data in the examples because they allow the results to be compared to the exactly known scene under fully controlled conditions. The simulator works in two stages. First, it propagates the beam from the lidar to the target, assumes diffuse reflection, and propagates back to the plane of the lidar receiver. This is repeated for various distances and for multiple random realizations of the diffuse reflecting surface, which give rise to different speckle. The result is a catalog of realistic return beams, represented as matrices of complex amplitudes, for various ranges. The second stage traces each transmitted laser pulse through the scene, and when a pulse hits an object, it randomly draws one of the return beam realizations for the corresponding range. The received power is calculated based on the reflectance of the target and the integral of the intensity of the selected return beam over the receiver aperture, and a pulse with this power is added to the simulated received signal. More complicated propagation effects, such as beam distortion by atmospheric turbulence and pulse distortion by reflection from surfaces not perpendicular to the beam, are omitted because they are not important for the short ranges considered in our examples. Thus, the shape of the received pulse is identical to the shape of the transmitted pulse.

In a real lidar, there are multiple noise sources such as backscatter from the optics and the atmosphere, background light, detector noise, and electronic noise. For the purpose of this paper, it is not important to model these in detail, so the simulator simply adds generic Gaussian noise to the electronic signal. As mentioned in Sec. 2, scattering from common optics (if the transmitter and receiver share the same aperture), and from the close part of the atmosphere, can dominate immediately after each transmitted pulse. Instead of modeling this scattering, we make the worst case assumption that the lidar has to discard the signal in an interval after each transmitted pulse. In the examples, we set this interval to 50 ns, which corresponds to 4.2% of the mean interval between transmitted pulses.

In order to facilitate the analysis, the first scene simply consists of nonoverlapping, rectangular planes. A more complex scene, approximating two buildings partially obscured by forest, is treated in Sec. 4.2.
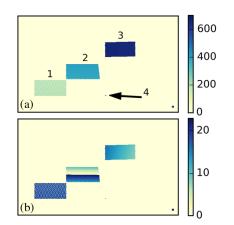
4.1 Scene 1

This scene contains four rectangular planes: size 10×5 m at 200-m distance, 20×10 m at 380 m, 30×15 m at 650 m, and 0.8×0.8 m also at 650 m. The three large objects have a reflectance of 0.1 and the small one 0.8. The laser pulse intervals are 1.0, 1.1, 1.2, 1.3, and 1.4 μs. The field of view, which is 250 mrad in azimuth by 150 mrad in pitch, is raster scanned with an azimuth scan speed of 300 rad/s and a pitch interval between adjacent scan lines of 0.5 mrad. Scanning the whole field of view takes 253 ms, so the number of transmitted pulses is about 2.1×10^5. The duration of the transmitted pulses is 4 ns (FWHM), and the receiver sample rate is 1 GHz. The signal from the detector is processed by a matched filter for the laser pulse shape. The actual value of the simulated transmitted power is not meaningful in itself, so to compare examples with different power we simply define the highest power to be 0 dB and give the power in the other examples as relative values. Similarly, we give dimensionless, normalized values for the detector signal and detection threshold.

The point clouds obtained under different conditions are compared to the known scene to count the points on each object, noise points close to objects, and noise points unrelated to objects. The point counts are compared to the corresponding numbers for a reference point cloud from a simulation without noise.

4.1.1 Simulation without noise

The pulse detection threshold in this case is set sufficiently low to detect all the return pulses. Because the objects in the scene do not overlap, it is convenient to display the point cloud as a color-coded range image, as shown in Fig. 3. With the chosen pulse intervals and scene, some of the return pulses from object 1 happen to fall in the masked zones after

Fig. 3 Range image without noise. The distances to the three large objects are, from left to right, 200, 380, and 650 m. For later reference, we label these objects 1, 2, and 3, respectively. The small object, which we label 4, is also at 650 m distance, and it is marked by an arrow because of the small size. All the distances are correctly identified by the proposed procedure. The field of view is 250 mrad × 150 mrad. The square in the lower right corner, which is not an object, shows the 3 × 3 mrad total angular size of the neighborhood used by the point detector. (a) Color-scale spans the whole depth of the scene. The bar to the right shows the range in meters. This color-scale is also used in the subsequent figures unless noted otherwise. (b) Color-scale repeats at 23 m intervals to show details better. The middle and right objects are tilted, which explains why they do not appear exactly rectangular.

Table 1 Summary of the number of correctly identified object points and the number of noise points in point clouds. P is the transmitted peak power in dB, T_D is the detection threshold (arbitrary units). Bin is the angular bin size in mrad, and T_F is the FOM threshold. Values calculated from Eq. (4) are marked by *. Total pulses are the number of detected pulses. N_i is 100 times the number of correct points on object i, divided by the corresponding number in the reference cloud. The objects are numbered as in Fig. 3. The tolerance for a point to be accepted as correct is 0.4 m. $N_{e,i}$ is 100 times the number of noise points between 0.4 and 8 m from object i, divided by the number of points on the corresponding object in the reference cloud. The number of such noise points can be relatively high because their FOM is increased by the real points on the close object. N_e is the number of other noise points, that is, noise points that are not associated with one of the true objects. The first row of numbers corresponds to the noiseless reference point cloud. The actual point counts for the reference cloud are given in the bottom row.

Figure	Parameters				Total		% Correct points				% Noise points				
	P	T_D	Bin	T_F	Pulses	Points	N_1	N_2	N_3	N_4	N_{1e}	N_{2e}	N_{3e}	N_{4e}	N_e
3			1.5	4*	17,944	17,944	100	100	100	100	0	0	0	0	0
4(a)	0	1	1.5	8*	74,451	15,302	100	100	53	0	1	1	1	0	0
4(b)	0	1	1.5	6	74,451	15,372	100	100	53	67	1	1	1	0	53
4(c)	0	0.8	1.5	20*	470,233	17,178	100	100	71	0	4	5	7	0	28
7(a)	−3	0.8	1.5	20*	468,852	12,884	100	96	7	0	3	6	3	0	29
7(b)	−3	0.7	1.5	34*	1,075,483	13,012	93	96	5	0	7	11	4	0	71
7(c)	−3	0.7	1.5	30	1,075,483	15,826	97	98	13	0	8	12	8	0	1730
7(d)	−3	0.8	3	50*	468,852	14,693	100	97	23	0	6	11	8	0	14
8(a)	−4.8	0.8	3	50*	465,966	11,789	100	77	2	0	7	11	2	0	24
8(b)	−4.8	0.8	3	44	465,966	12,915	100	77	5	0	7	12	4	0	725
8(c)	−4.8	0.7	3	93*	1,074,982	13,500	100	85	3	0	12	22	3	0	72
10(a)	0	1	1.5	9	74,451	15,328	100	100	53	100	1	1	1	17	0
10(b)	−3	0.7	1.5	42	1,075,483	14,138	100	98	7	0	10	15	5	0	13
3		Actual point counts in ref. cloud:			5383	6597	5958	6	0	0	0	0	0		

some of the output pulses, and this explains why object 1 does not look dense. The neighborhood size (in either direction from a point candidate) is set to 1.5 mrad in each of the angles and 5 m in range, and the FOM threshold automatically determined from Eq. (4) is 4. FOM thresholds in the range from 2 to 30 work well, although the small object is lost when the FOM threshold exceeds 5. Table 1 summarizes the number of points on each object, as well as the number of noise points, for this and the following examples. We have verified that the point cloud is not very sensitive to neighborhood size, provided that the FOM threshold is adjusted correspondingly.
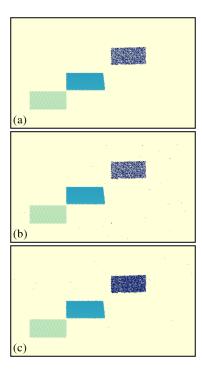
4.1.2 Transmitted power 0 dB

In this and the following simulations, noise is included. We define the SNR for an object as the mean return pulse amplitude in the absence of noise divided by the RMS noise after the matched filter. For objects 1 to 4, as defined in Fig. 3, the SNR is 37, 11, 3.5, and 28, respectively.

The detection threshold is first set to 1, which equals the mean return pulse amplitude from the far object. This yields about 75,000 detected pulses, ∼0.35 per transmitted pulse. Figure 4(a) shows the range image for FOM threshold 8, which is the value from Eq. (4). The small object is not visible in this case. Its angular extent is only 1.2 mrad, much smaller than the total neighborhood size of 2×1.5 mrad, so

it does not give rise to enough point candidates to overcome the FOM threshold. It can be recovered by reducing the FOM threshold to 6, as shown in Fig. 4(b), but this also introduces more noise points, as seen in Table 1. The detected parts of the far object depend on speckle patterns and noise, so the details will vary with different realizations of the simulated signal.

To find more points on the far object, the detection threshold is reduced to 0.8. The number of detected pulses rises to 4.7×10^5, or 2.2 per transmitted pulse, and Fig. 4(c) shows the corresponding range image. As expected when the number of point candidates increases and the parameter for error probability is fixed, the number of noise points increases, as shown in Table 1.

For illustration of the noise suppression, Fig. 5(a) shows the point cloud of a volume around the far object with the same parameters as in Fig. 4(c). This can be compared to Fig. 5(b), where the FOM threshold is 1 so all detected pulses are identified as points. It is clear that the algorithm removes most of the noise points. However, noise points close to the object remain because they acquire a high FOM from the neighboring, correctly detected, points on the object. The number of such noise points can be reduced using a smaller neighborhood size in the range direction, but as explained in Sec. 3, this would also reduce the detection performance for objects with a large variation in range.

Fig. 4 Range images with 0-dB transmitted peak power and 1.5-mrad angular bin size. The color-scale and the field of view are the same as in Fig. 3(a). (a) Detection threshold 1 (3.5 times the RMS noise) and FOM threshold 8 (automatic). (b) FOM threshold 6. (c) Detection threshold reduced 0.8 and FOM threshold 20 (automatic).
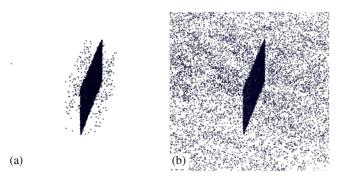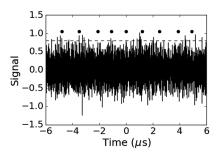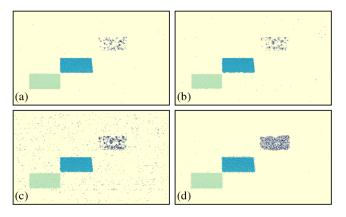

Fig. 5 Point cloud of the far object with 0-dB transmitted peak power and detection threshold 0.8. The size of the object is 30 m × 15 m. (a) FOM threshold 20. (b) FOM threshold 1.

4.1.3 *Transmitted power −3 dB*

At this power level, the SNR for the return pulses from object 3 is 1.75, and they are almost indistinguishable from the noise spikes, as shown in Fig. 6. Nevertheless, the proposed method yields a point cloud where parts of the far object can be seen, as shown in Fig. 7. The pulse detection threshold is 0.8 or 0.7, and the corresponding mean number of detected pulses per transmitted pulse is ∼2.2 or 5, respectively. In Fig. 7(d), the angular neighborhood size is increased to 3 mrad in azimuth and pitch. As expected from Eq. (3), using a larger neighborhood improves the SNR, but compared to Fig. 7(a), some more noise points appear along the edges of objects. Furthermore, the small object is difficult to recover in this case because it is much smaller than the neighborhood size.

Fig 6 Typical return signal from the far object in the case with −3 dB transmitted peak power. The dots above the graph mark the positions of expected return pulses, and the dashed line shows the detection threshold 0.8 used in Fig. 7(a).
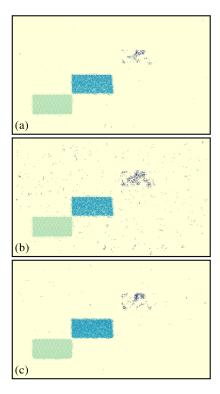

Fig. 7 Range image with −3-dB transmitted peak power. (a) Detection threshold 0.8, 1.5 mrad angular neighborhood size and FOM threshold 20 (automatic). (b) Detection threshold 0.7 and FOM threshold 34 (automatic). (c) As (b), but with FOM threshold 30. (d) Detection threshold 0.8, 3 mrad angular neighborhood size and FOM threshold 50 (automatic).

4.1.4 *Transmitted power −4.8 dB*

This power corresponds to an SNR of 1.2 for object 3. Figure 8 shows corresponding range images with different FOM thresholds. In Fig. 8(a), with detection threshold 0.8, only small patches of object 3 are detected. Reducing the FOM threshold below the automatic value, in Fig. 8(b), introduces a lot of noise while detection of object 3 is still poor. In Fig. 8(c), the detection threshold is reduced to 0.7. The result is not very much better than in Fig. 8(a), but nevertheless, the fact that a significant part of the far object is correctly detected even under these conditions illustrates the power of the algorithm.

4.2 *Scene 2*

In this section, we show a somewhat more realistic scene consisting of two identical box-like "buildings," in some cases partially obscured by "forest." Figure 9(a) shows the point clouds of the buildings when the forest is absent. The two long sides of the buildings are 20 m, and they are 7 m high. The left and right buildings lie in the range intervals 680 to 700 m and 650 to 670 m, respectively, and the lidar looks down from an altitude of 100 m. The reflectance of the buildings and the ground is 0.1. The lidar field of view is 250 mrad in azimuth by 30 mrad in pitch, and other parameters are the same as for scene 1.

Fig. 8 Range images with −4.8-dB transmitted peak power and 3-mrad angular neighborhood size. (a) Detection threshold 0.8 and FOM threshold 50 (automatic). (b) As above, but FOM threshold 44. (c) Detection threshold 0.7 and FOM threshold 93 (automatic).

The transmitted power is 0 dB. The bin size is 1.5 mrad in pitch and azimuth and 5 m in range.

The left building is deliberately placed at a distance where some of the return pulses fall in the discarded intervals after the transmitted pulses. Specifically, the range 690 to 697.5 m corresponds to delays of 4.6 to 4.65 μs. Return pulses originating from the first pulse in a pulse group (the pulse followed by the 1 μs interval) will fall in the interval discarded after the fourth pulse (the pulse before the 1.4 μs interval). Thus, one in five return pulses from this range interval will be discarded. Comparison of the two buildings in Fig. 9(a) does indeed show that there are fewer points on the left part of the left building. Because of the large angle of incidence to the roof, this part of the object has relatively few points even under the best conditions, so it is sensitive to a few missing return pulses.

In Fig. 9(b), a partially transmitting "forest" is placed between the lidar and the buildings. The forest is modeled

as a screen with holes such that the incident laser beam has equal probability of being transmitted or reflected (with reflectance 0.1). Furthermore, the reflected pulses are given a random delay corresponding to a range variation uniformly distributed in a 10-m interval. With only half of the pulses reaching the buildings, the two walls facing the lidar are still detected clearly, but the point clouds are sparse on the other parts of the buildings. When the FOM threshold is reduced below the automatic setting, as in Fig. 9(c), denser point clouds are obtained.

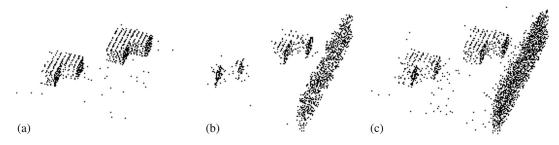## 5 Generalized Figure of Merit

The FOM measure described in Sec. 2 and used in Sec. 4, which is simply equal to the number of point candidates in the neighborhood, has the advantage of being mathematically simple so a threshold value can be calculated from the statistics, Eq. (4). However, from a practical point of view, it would make sense to give more weight to pulses well above threshold than to pulses just above, which are more likely to be noise. To implement this idea, each point candidate is assigned a quality value $Q = V_p/T_D$, where V_p is the peak voltage of the pulse and T_D is the detection threshold. To avoid that a very strong pulse overwhelms a lot of weaker pulses, Q is clipped at an upper limit Q_{max}. The new FOM is the sum of the Q-values from the point candidate itself and its neighbors. It is much more complicated to derive a distribution for the FOM values in this case, so as a practical solution we calculate the threshold as before and multiply it by a correction factor

$$T_F' = T_F \times \langle Q \rangle, \tag{5}$$

where the mean quality value $\langle Q \rangle$ is calculated over the 80% of the cells with fewest point candidates, that is, the same cells that are used to estimate the density of noise points. If there are only a few point candidates in these cells, $\langle Q \rangle$ is instead calculated from the set of cells which contain a single point candidate each.

Consider again Fig. 4(b). The small object is visible, but it is difficult to distinguish from the noise that also appears when the FOM threshold was reduced. However, the small object has higher reflectance than the other objects, so when the pulse power is included in the FOM, it can be recovered, as shown in Fig. 10(a). $Q_{max} = 3$ and the FOM threshold was increased to 9, by Eq. (5).

Similarly, Fig. 10(b) can be compared to Fig. 7(b). The small object is not recovered, but for the other objects the numbers of correct points and close noise points increase while the number of other noise points decreases. To

Fig 9 Point clouds of two buildings on a flat ground. To make the buildings clearly visible, only points above the ground level are shown. (a) Obscuring forest absent, detection threshold 1 and FOM threshold 11 (automatic). (b) Forest present, same parameters as in (a). (c) Forest present, FOM threshold 4.
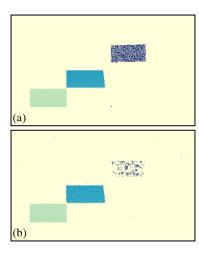
Fig. 10 Range images with modified FOM measure and $Q_{max} = 3$. (a) 0-dB transmitted power, detection threshold 1, and FOM threshold 9. This can be compared to Figs. 4(a) and 4(b). (b) −3 dB transmitted power, detection threshold 0.7, and manually tuned FOM threshold 42. The result can be compared to Fig. 7(b).

suppress noise, the FOM threshold in Fig. 10(b) was manually tuned to 42 instead of 39 from Eq. (5). Therefore, this example illustrates the limit of the method rather than practical performance.

6 Discussion

The angular neighborhood size is a trade-off between sensitivity for large objects and risk of suppressing smaller objects. The improved FOM measure can help detecting small objects, but only if they have relatively high reflectance and not if they are too small compared to the neighborhood size. Detection of both small and large objects could be improved by running the algorithm multiple times with different parameters. Some of the processing could be common, so the run time would not scale linearly with the number of passes. It must be noted that the size of the neighborhood boxes does not limit the resolution or accuracy of the point cloud. The boxes are used to solve ambiguity, whereas the actual point positions are entirely determined by the input data.

As shown in Sec. 4.2, the automatic setting of the FOM threshold can be too high in scenes where objects are partially obscured. The setting in such cases should include application-specific information such as whether objects of interest are likely to be obscured. It would also be possible to detect obscuring objects such as trees and adjust the FOM setting in their shadow adaptively, but this would be a topic for future investigation.

The observed number of noise points outside the neighborhood of any objects, N_e in Table 1, can be compared to the expected number $\hat{N}_e = eN_d n$, where e is the error probability in Eq. (4), N_d is the number of detected pulses, and $n = 5$ is the number of different pulse intervals, i.e., the number of point candidates created for each detected pulse. For detection threshold 0.7 or 0.8 and T_F values calculated with Eq. (4) (marked by * in Table 1), the observed N_e values are indeed consistent with $\hat{N}_e$. For higher detection thresholds, the number of noise points is too small to estimate $\langle N_n \rangle$ accurately.

Operation with a very low SNR, as in some of our examples, may seem contrived. If the pulse energy from the laser could be adjusted freely, only constrained by a maximum average power, it would be better to transmit stronger pulses at a lower rate. However, real lasers have a maximum pulse energy and a limited range of pulse rates, so operation close to the noise limit may be of interest in practical lidar applications.

The method can of course also be used for noise suppression in a system with fixed pulse intervals. If the pulse interval is long enough compared to the maximum delay of a return pulse, there will be only one point candidate for each pulse and no risk of placing an object at the wrong distance.

It should be mentioned that an additional reason to use varying pulse intervals is to ameliorate the problem with scattering by common optics or by the atmosphere close to the lidar. Such scattering can dwarf weak return pulses arriving immediately after an output pulse. If the pulse interval is fixed, all the lost return pulses will come from distances corresponding to multiples of the pulse interval, so there will be more or less blind zones around these distances. On the other hand, with varying pulse intervals, the lost return pulses will be spread over different distances and not give rise to blind zones.

The current implementation runs on a PC and can take tens of seconds to process images with millions of initial point candidates, such as Fig. 8, but this program has not been optimized. In addition to improving the code, there is potential to save time by simplification and parallelization. One possible simplification is to group point candidates into cells in a 3-D grid and work with the grid cells instead of individual points. A disadvantage of this method is that the ability to detect an object will depend on how it is placed and aligned with respect to the grid. For parallelization, the 3-D grid could be partitioned into blocks, which could be processed almost independently.

7 Conclusion

We have demonstrated that a method based on clustering of point candidates can resolve range ambiguity and suppress noise pulses. The efficient noise suppression makes it possible to push the detection threshold close to the noise level and still obtain relatively clean point clouds.

References

1. H. Lu et al., "An automatic range ambiguity solution in high-repetition-rate airborne laser scanner using priori terrain prediction," *IEEE Geosci. Remote Sens. Lett.* **12**, 2232–2236 (2015).
2. G. Kim and Y. Park, "Lidar pulse coding for high resolution range imaging at improved refresh rate," *Opt. Express* **24**, 23810–23828 (2016).
3. P. Lindelöw and J. J. Mohr, "Coherent lidar modulated with frequency stepped pulse trains for unambiguous high duty cycle range and velocity sensing in the atmosphere," in *IEEE Int. Geoscience and Remote Sensing Symp.*, IEEE (2007).
4. M. U. Piracha et al., "Range resolved lidar for long distance ranging with sub-millimeter resolution," *Opt. Express* **18**, 7184–7189 (2010).
5. G. W. Stimson et al., *Introduction to Airborne Radar*, 3rd ed., SciTech Publishing, Mendham, New Jersey (2014).
6. M. Skolnik, *Radar Handbook*, 3rd ed., McGraw-Hill, New York (2008).
7. R. E. Warburton et al., "Subcentimeter depth resolution using a single-photon counting time-of-flight laser ranging system at 1550 nm wavelength," *Opt. Lett.* **32**, 2266–2268 (2007).

8. P. A. Hiskett et al., "A photon-counting time-of-flight ranging technique developed for the avoidance of range ambiguity at gigahertz clock rates," *Opt. Express* **16**, 13685–13698 (2008).
9. N. J. Krichel, A. McCarthy, and G. S. Buller, "Resolving range ambiguity in a photon counting depth imager operating at kilometer distances," *Opt. Express* **18**, 9192–9206 (2010).
10. P. Rieger and A. Ullrich, "Resolving range ambiguities in high repetition rate airborne lidar applications," *Proc. SPIE* **8186**, 81860A (2011).
11. P. Rieger, "Range ambiguity resolution technique applying pulse-position modulation in time-of-flight scanning lidar applications," *Opt. Eng.* **53**, 061614 (2014).

Gunnar Arisholm received his BSc degree in computer science from University of Strathclyde in 1987 and joined FFI the same year. At FFI he changed subjects to lasers and nonlinear optics, and in 2000 he received his PhD in physics from the University of Oslo. His current interests include optical frequency conversion, lidar and atmospheric beam propagation.

Biographies for the other authors are not available.