Simulation of 3D Laser Radar Systems

Michael E. O'Brien and Daniel G. Fouche

■ **We describe a computer model that was developed to simulate the performance of three-dimensional (3D) laser radars (ladars) that use arrays of Geiger-mode avalanche photodiode (APD) detectors. The model allows considerable flexibility in the specifications of sensor characteristics, 3D scene, background light, and dynamics of the sensor platform. It is used to help design and predict the performance of 3D ladars used for surveillance, city topography, combat identification, and other applications. In particular, the model was used to help design the Laboratory's foliage-penetrating airborne 3D ladar for the DARPA-sponsored Jigsaw program. Results of the model's simulations of Jigsaw Phase 2 experiments agree quantitatively with actual measurements of a tank in the open. In addition, the model's simulations agree well qualitatively with actual measurements of a tank under trees. Both the simulation and the Jigsaw data demonstrate an ability to obtain detailed 3D images of objects under thick foliage.**

Lincoln Laboratory has developed, and continues to develop, laser and detector technologies that make it possible to build compact, highly capable three-dimensional (3D) laser radars, or ladars [1]. The laser technology is based on diode-pumped, solid-state, microchip lasers that are passively *Q*-switched [2, 3]. The detector technology is based on arrays of avalanche photodiode (APD) detectors operating in Geiger mode, with integrated timing circuitry for each detector [4].

Figure 1 shows the basic ladar concept. A short pulse of light from a laser diverges to illuminate the entire scene of interest. The reflected light is imaged onto a two-dimensional array of detectors. Rather than measuring light intensity, as in a traditional camera, these detectors measure the time of arrival of the reflected light. This arrival time is linearly dependent on the range to the part of the scene imaged onto a detector. With each pixel coded with range, the ladar produces a 3D angle-angle-range image from a single laser pulse.

Our initial interest for this type of 3D ladar was discrimination and aim-point selection for advanced interceptor seekers. The ladar is also well suited for surveillance, topography, and navigation of autonomous vehicles and robots. In addition, the Laboratory ladar used for the Defense Advanced Research Program Administration (DARPA) Jigsaw program made 3D images of objects that were obscured by thick foliage [5].

To demonstrate the concept for this type of 3D ladar, the Laboratory originally constructed an experimental system, or brassboard, using a 1-kHz microchip laser, a 4 × 4 array of Geiger-mode APDs, and scanning mirrors to build up images with 128 × 128 or so pixels. Details of the brassboard and the 3D images obtained from it are found elsewhere [4, 6, 7]. More recent 3D ladars use 32 × 32 arrays of integrated detectors and timing circuitry. These ladars and the laser and detector technologies on which they are based are described in other articles [1, 4, 5, 8]. Larger arrays are now being developed.

Before the 32 × 32 detector arrays were built into fieldable systems, we developed a computer model to simulate 3D ladars of this type. We use the model to help design ladars, to predict overall system perfor-

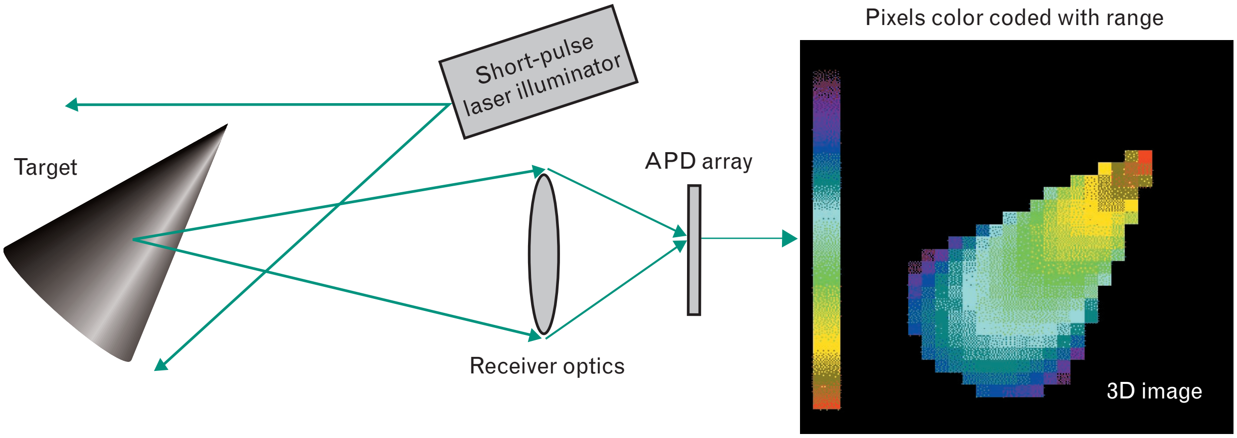

FIGURE 1. Basic concept for three-dimensional (3D) angle-angle-range laser radar (ladar) using an imaging detector array. The entire scene is flood illuminated and imaged on a single laser pulse. Each pixel in the avalanche photodiode (APD) detector array measures the time of arrival of the reflected light. The arrival time depends on the range to the imaged target element. With each pixel coded with range, the ladar produces a 3D angle-angle-range image from a single laser pulse.

mance, and to develop data-processing algorithms. In this article, we describe the model and compare its results to measurement data from the Jigsaw foliage-penetrating ladar.

Analysis of Geiger-Mode Probabilities of Detection and False Alarm

In our simulations, we use the same model for every detector in the focal-plane array. This section describes the single-detector model. We explain the use of Poisson statistics for both signal and noise, describe the detection and timing model, and show single-pulse and multiple-pulse probabilities of detection and false alarm. The multiple-pulse results, which are obtained through a Monte Carlo technique, include those for a target obscured by foliage or camouflage.

Applicability of Poisson Statistics

In Geiger mode, the reverse bias across an APD is so high that the rate of creation of charge carriers by impact ionization exceeds the rate at which the charge carriers can be extracted from the device, and the output current quickly surges to a saturated value set by circuit parameters. A single electron can initiate the current surge, or *firing*. We use the term *primary electron* for an electron that initiates a firing. Primary electrons include both photoelectrons, which are created by absorption of light, and dark-current electrons, which are created by thermal effects within the detector material. Sources of photoelectrons are laser light reflected from the target, background light (e.g., reflected sunlight), and aerosol backscatter of laser light. For our simulations, we assume that detector firings from all light and thermal sources follow Poisson statistics. This assumption is based on the following considerations.

First, Lincoln Laboratory demonstrated experimentally that firings in response to incident incoherent light follow Poisson statistics [7]. We illuminated an array of Geiger-mode APDs with incoherent light from incandescent lamps providing a nearly constant photon rate. We repetitively biased the detectors into Geiger mode and recorded the times (relative to the bias-on time) of the firings. Figure 2 shows the results [6]. The graph shows the measured probability of firing at a particular time after turn-on versus that time. The straight line, a least-squares fit to the data of this log-linear graph, fits the data quite well. It follows the equation $P(t) = 0.0145 \exp(-0.147t)$, the functional form of which agrees well with that of the theoretical equation for a Poisson process; namely, $P(t) = r\exp(-rt)$, where r is the photon rate at the detector. Thus, firing from incoherent light, such as reflected sunlight, is a Poisson process.

Detection of reflected laser light is a Poisson process under two conditions, both of which we have assumed have been met for the simulations to date. The first condition is that the quantity d/A must be much greater

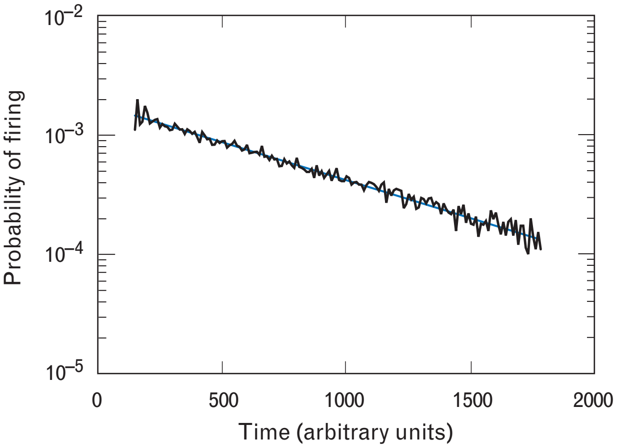

FIGURE 2. Measured probability of firing versus time when an array of Geiger-mode detectors was illuminated with incoherent light. The straight line is a least-squares fit to the data of this log-linear graph. The exponential dependence indicates that firing of Geiger-mode APDs by incoherent light follows Poisson statistics.

than one, where d is the number of degrees of freedom in the sampled intensity (speckle) distribution and A is the mean number of photoelectrons in the measurement interval [9]. Generally, $d/A >> 1$ for Lincoln Laboratory ladars, since we design them so that $d >> 1$ and operate them such that $A < 1$. The value of d is large because the instantaneous field of view of a detector is several times larger than the diffraction-limited angle. Therefore, many speckles in the reflected intensity pattern fall within the receiving telescope. For Jigsaw, d/A is of order 10^4, and the first condition is well satisfied.

The second condition is that atmospheric turbulence must have a negligible effect on the speckle pattern at the receiver. For the Jigsaw simulations presented in this article, the vertical propagation distances are so short (about 150 m) that turbulence effects are indeed expected to be negligible. With both conditions satisfied in the Jigsaw scenario, our use of Poisson statistics for laser detection in the Jigsaw simulation is well justified.

This conclusion applies not only to the target return (i.e., the signal), but also to the backscatter of laser light from atmospheric aerosol in front of the target. Aerosol backscatter will meet the Poisson conditions even better than the signal will, as the value of d can only be larger (if only because the range depth of the backscatter could exceed the coherence length of the laser light), the value of A will likely be smaller (and must be less than one anyway in order to measure the target, as we show later), and the propagation length through turbulence will be shorter.

Finally, we note that Alex McIntosh of the Electro-optical Materials and Devices group at Lincoln Laboratory has experimentally determined that Geiger-mode APD detector firings from dark current follow Poisson statistics [10]. His procedure was the same as that described above for incoherent light (which led to Figure 2), except he prevented any light from reaching the detector.

In our simulations, therefore, we assume Poisson statistics for detector firings initiated by photons and by dark current. Firings from laser light reflected from the target constitute the signal. Firings from background light, aerosol backscatter, and dark current constitute noise. The only other source of noise in Geiger-mode APDs that we must be concerned about is cross talk, and that topic is handled in a later section.

Relevant Properties of Poisson Processes

For a Poisson process, the probability that m events occur between times t_1 and t_2 is

$$P(m;t_1,t_2) = \frac{1}{m!}[M(t_1,t_2)]^m \exp[-M(t_1,t_2)], \quad (1)$$

where

$$M(t_1,t_2) = \int_{t_1}^{t_2} r(t)dt \quad (2)$$

and $r(t)$ is the rate function of the process [9]. In our case, an event is the creation of a primary electron. The mean of this probability distribution is equal to $M(t_1,t_2)$. From Equation 1, the probability that no primary electrons are created between times t_1 and t_2 is $\exp[-M(t_1,t_2)]$, and the probability that one or more are created is $1 - \exp[-M(t_1,t_2)]$.

A Poisson process has the reproductive property; i.e., the sum of any number of statistically independent, Poisson-distributed random variables is itself a Poisson-distributed random variable, with a mean rate given by the sum of the individual mean rates. Another property of a Poisson process is that the number of events generated in any time interval is statistically independent of the number generated in any other non-overlapping interval.

Model for Detection by Typical Geiger-Mode APD and Electronics

Three Poisson processes create primary electrons: absorption of laser photons, absorption of background photons, and generation of dark current. These processes are statistically independent of one another as long as the rate of primary-electron creation is not saturated. We assume that this is the case because our ladars are generally not operated in a saturated regime. Therefore, the three occurring simultaneously constitute a Poisson process with a mean rate of creating primary electrons given by the sum of the three individual mean rates.

The APD is biased into Geiger mode for a particular time interval, or *gate*, on every laser pulse. The detector fires in response to the first primary electron created within the gate. Because the detector takes ten nanoseconds or more to reset after firing, we have made no attempt in our ladars to date to enable the detector to fire more than once per laser pulse. The detector either fires once or not at all. Thus, if the detector fires from a noise source before laser photons reflected from the target arrive, then the detector will not respond to the target return for that laser pulse.

Within the gate, time is divided into discrete bins, each with the same duration. The electronics provide the time (i.e., the bin number) of the detector firing, if any. The bin width is typically 0.5 nsecs, corresponding to a range bin that is 7.5 cm wide.

Single-Pulse Firing Probabilities for Geiger-Mode Detectors

Equations 1 and 2 can be adapted to the case of discrete bins. We identify t_1 and t_2 as the start and stop times of a particular bin, say the ith bin; $M(t_1,t_2)$ as M_i, which is the mean number of primary electrons created in the ith bin from all three Poisson processes; and $P(m;t_1,t_2)$ as $P(m;i)$, which is the probability that m primary electrons are created in the ith bin.

Consider the results of a single laser pulse. Because the detector fires in response to the first primary electron, the only way for firing to occur in the jth bin is to have no primary electrons in the first $j-1$ bins and then at least one primary electron in the jth bin. Therefore, the probability that the detector fires in the jth bin is

$$\begin{aligned} P_j &= \left[\prod_{i=1}^{j-1} P(m=0;i)\right] P(m \geq 1; j) \\ &= \exp\left(-\sum_{i=1}^{j-1} M_i\right)\left[1-\exp(-M_j)\right]. \end{aligned} \tag{3}$$

The first equality in Equation 3 holds because the number of events in any bin is independent of the number of events in any other bin for a Poisson process. Equation 3 shows that, for a Geiger-mode detector, the probability of firing in the jth bin is determined not only by M_j but also by the values of M_i $(i < j)$ in all the earlier bins. In fact, that probability is always reduced by any nonzero values of M_i in earlier bins.

It is important to note that a Geiger-mode detector can provide a linear output. Suppose that the sum of the M_i for all the bins in Equation 3 is much less than one primary electron. Then the first factor on the right-hand side is approximately one, and the second factor is approximately M_j. Therefore, the probability of firing in the jth bin is essentially equal to M_j; i.e., it is linearly proportional to the signal-plus-noise level for that bin. Suppose we repeat the measurement n times under identical conditions for n laser pulses. Then a plot of the number of firings versus bin number (i.e., the time histogram or range histogram) will closely approximate a realization of the single-pulse temporal output from a linear detector, where the linear detector has the same set of M_j values, but with each multiplied by n, and has no noise other than that from the Poisson detection statistics represented by the values of nM_j. By using this technique—reducing signal and noise to much less than one primary electron per pulse and repeating the measurement over many laser pulses—we are able to use Geiger-mode detectors to see targets behind obscurants, as is demonstrated later in this article.

Let us assume that the mean rates of primary-electron creation by absorption of background photons and by dark current are constant during data collection. We define the combined mean rate as N (the noise), with units of primary electrons per gate interval, and the number of bins within the gate as b. Then the noise contribution to all M_i is $w = N/b$. For now, we assume there is no obscurant in front of the target. We define S (the signal) as the mean number of target-reflected laser photoelectrons created within the gate, and we define the bins such that the entire signal falls within a single one called the target bin. Let f be the fraction of the b bins that are in front of the target bin. Then, from Equation 3, the probability of firing in the target bin is

$$P_{\text{target}} = \exp(-fN)\left[1-\exp(-S-w)\right]. \tag{4}$$

Equation 4 shows that the probability of firing in the target bin is reduced by the factor $\exp(-fN)$, where fN is the mean number of noise primary electrons created within the gate before the arrival of laser photons from the target. For noise not to affect P_{target} substantially, the value of fN must be much less than one primary electron.

Figure 3 shows log-linear graphs of P_{target} (solid curves) as functions of the laser signal S, with the noise level N as a parameter. There is no obscurant for these curves. The dotted curves are probabilities of false alarm (P_{FA}), which is the probability that the detector fires in one of the non-target bins. The graphs differ in the value of f. From top to bottom, the target return arrives at the beginning ($f = 0$), in the middle ($f = 0.495$), and at the end ($f = 0.995$) of the gate. The number of bins b is fixed at 200. This choice is not critical, since the results change only slightly as the number of bins varies between ten and infinity.

When the target is at the very front of the gate (Figure 3, top), signal photoelectrons have an opportunity to trigger the detector in the first bin, making the probabilities of detection (i.e., the probabilities of firing in the target bin) nearly independent of the noise level. Increased noise actually increases P_{target} a little, because noise, in addition to the target return, can cause a firing in the target bin.

The situation is reversed when the target is at the very back of the gate (Figure 3, bottom), as noise has full opportunity to trigger the detector in all bins. From Equation 4 with f near unity, P_{target} is limited to approximately $\exp(-N)$ no matter how strong the signal. For example, for P_{target} to be greater than 0.37 requires that noise be no more than one primary electron per gate interval.

When the target is in the middle of the gate (Figure 3, middle), noise can potentially trigger the detector in only half of the bins before the target return arrives. Thus P_{target} is limited to approximately $\exp(-N/2)$.

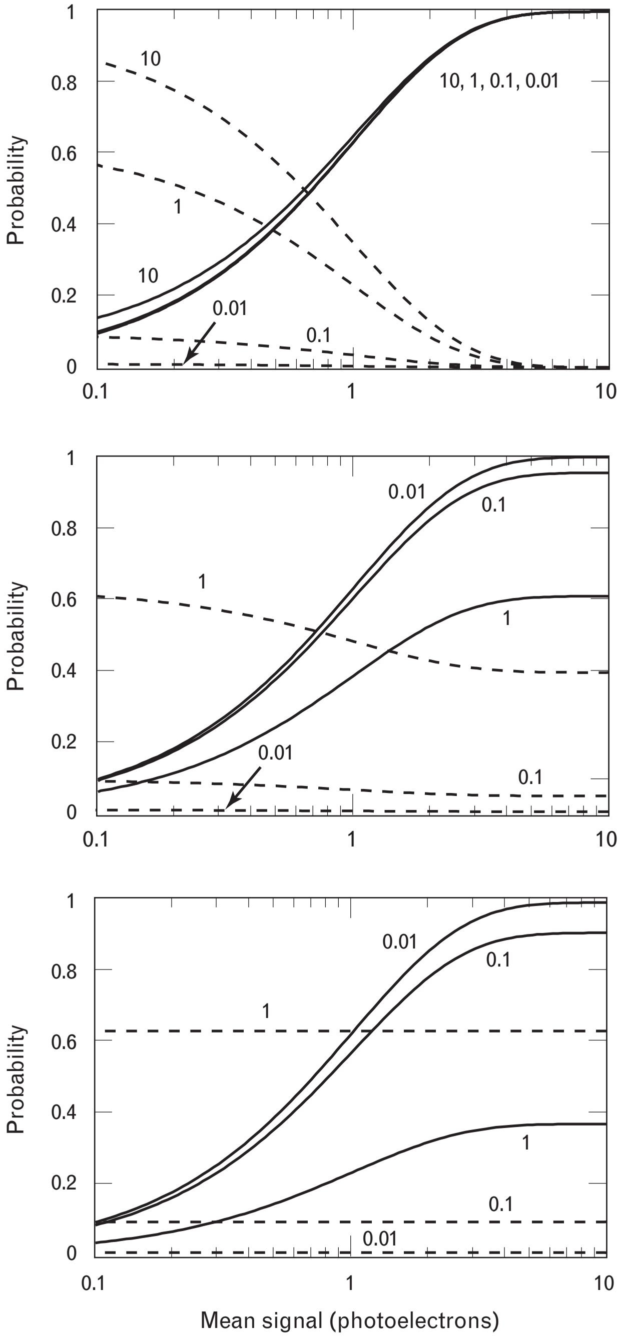

FIGURE 3. Theoretical single-pulse probabilities of target detection (solid curves) and false alarm (dashed curves) versus mean signal level S for a Geiger-mode detector. The curves are differentiated by, and labeled with, the mean noise level N, in units of primary electrons per detector-on time. From top to bottom, the target return arrives at the beginning, middle, and end of the gate. There are 200 time bins within the gate, and the target return falls in one bin. The graphs show that the probability of detection is greatly affected by how much noise occurs before the laser signal arrives at the detector.

Improving Single-Pulse Probabilities

It is clearly advantageous to minimize the value of fN, the mean number of primary electrons created by noise between the beginning of the gate and the target bin. One way to minimize this value is to move the gate so that it opens just before laser photons reflected from the target arrive. The degree of minimization, however, is limited by the range depth of the imaged scene, because in the Lincoln Laboratory ladars all the detectors in the array are biased into Geiger mode simultaneously. Another way to reduce the value of fN is to cut down the noise level, such as by cooling the detector (to reduce dark current) and by using a narrow-bandpass spectral filter in the receiver optical train (to reduce background light).

Despite taking all practical measures, we might find that the background-light level is still high enough to prevent the detector from reaching the desired probability of detection. In that case, we can always further decrease background light at the detector by putting an attenuator into the optical path of the receiver or by reducing the size of the receiver aperture. However, because the target return is attenuated by the same factor as the noise is, we must ensure that enough signal remains to yield the desired probability of detecting the target.

Multiple-Pulse Detection and False-Alarm Probabilities

If we cannot obtain the desired detection and false-alarm probabilities on a single pulse, we can improve the probabilities by processing data from multiple pulses. We assume that any and all target returns from pulse to pulse fall within the same time bin of the gate. The assumption is valid for a static scenario. It would also apply when the delay between transmission of the laser pulse and opening of the gate is adjusted to compensate for any range rate. In either case, signal firings from the target accumulate in one bin, while noise firings from background light and dark current are spread randomly over all bins. In multiple-pulse processing, we simply identify the bin in which accumulation occurs; i.e., we identify coincidences of the target returns.

Data from n laser pulses are processed as a set. On each pulse, we note the bin number of the firing, if any. If we assume that the probability of firing in any bin remains constant for all n laser pulses and that the results of any pulse are independent of the results of all the other pulses, then, for any bin, the number of firings in n pulses follows binomial statistics. We can then calculate analytically the probabilities of interest, at least for a few pulses [11].

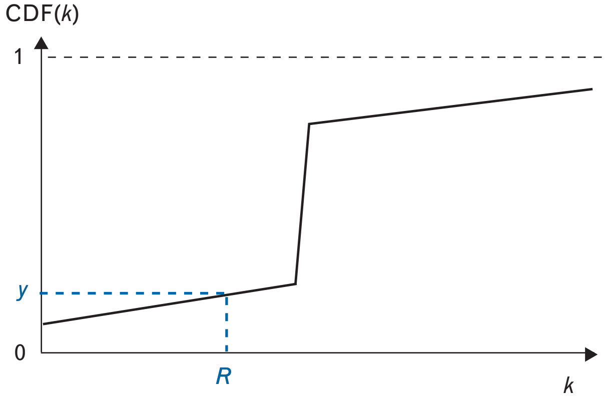

FIGURE 4. Illustration of Monte Carlo technique for choosing the range that a detector measures on a particular laser pulse, given a previously determined cumulative distribution function (CDF). A random number y selected from a uniform distribution between 0 and 1 is mapped into a range R via the CDF.

Monte Carlo Technique

When there is noise and more than a few pulses, we use a Monte Carlo technique to determine the probabilities. We first specify the signal-plus-noise level (in primary electrons) versus time within the range gate. These determine the M_j values for each bin needed for Equation 3. The contributions of background light and dark current to the M_j values are assumed constant for all bins. We specify the range distribution of any obscurant and assume (for now) that the target return is in a single bin. By using Equation 3, we calculate the single-pulse P_j for all range bins j and then calculate its cumulative distribution.

To represent at what range, if any, the detector fires on a given laser pulse, we let the computer choose a random number from a uniform distribution over zero to one, and then we translate the random number to a range via the cumulative distribution function

$$\mathrm{CDF}(k) = \sum_{j=1}^{k} P_j \, .$$

Figure 4 illustrates this translation. We perform this process Q times and store the results. The number Q is large, typically on the order of ten million.

When the scenario of interest involves n laser pulses, we divide the Q results into sets of n pulses. For each of the Q/n sets, we determine according to some detection law if the target is detected, or if there is a false alarm, or if neither happens. An example of a detection law (the most-firings law) is to choose as the target bin that which has the most detector firings. Another law (the threshold law) is to choose as the target bin the only bin for which the number of firings exceeds some threshold, if indeed there is one and only one. For any law, a detection occurs if the chosen bin is actually the target bin, a false alarm occurs if the chosen bin is a non-target bin, and neither occurs if no bin is chosen. The probability of detection is then the number of sets for which a detection occurred divided by Q/n. The probability of false alarm is defined similarly.

In contrast to the single-pulse case, the multiple-pulse probabilities are significantly affected by the number of time bins within a gate of fixed duration. The number of bins is important because dividing a given amount of noise into more bins lowers the probability of accumulating multiple firings in any particular bin. The multiple-pulse calculations shown later in this article are for 200 bins per gate, the same as for the single-pulse results shown above.

Monte Carlo Results (No Obscurants)

Figure 5 shows examples of Monte Carlo results for a single pixel when there is no obscurant, that is, when there is no aerosol and when only the target is within the instantaneous field of view of the detector. For all four graphs, there are 200 range bins and the target is in the middle of the gate ($f = 0.5$). The four graphs differ in noise level B (photoelectrons/gate) and in the detection law, as described in the figure caption. The abscissa for each graph is the number of laser pulses n. The ordinate is the total signal in n pulses, i.e., the signal level per pulse times the number of pulses. Each blue curve is for a specific detection probability and each red curve is for a specific false-alarm probability, as labeled. Presenting the results in this format allows us to quickly determine an acceptable number of pulses over which to divide the total energy available for the measurement. The wiggles in the curves are caused by probabilistic variations in our Monte Carlo calculations and could be smoothed out by using more trials (i.e., a larger Q).

For Figure 5(a), there is no noise, and the threshold law with a threshold of two is used. We see that obtaining a detection probability of 99% requires a total of seven photoelectrons spread over ten or more pulses. Even more total signal is required at fewer than ten pulses because the detector is entering saturation. As a point of comparison, it takes only 4.6 photoelectrons to obtain a detection probability of 99% for either a single pulse, the most-firings law, or the threshold law with a threshold of one.

For Figure 5(b), noise at a level of 0.1 photoelectrons per gate is included. We see that there is a penalty for dividing the fixed amount of total signal into too many pulses. The penalty arises because the 0.1 photoelectrons of noise is encountered on every pulse. With more pulses, noise has a better chance of triggering the detector at least twice in the same bin. Once a noise bin crosses the threshold, a detection is no longer possible under the threshold law, no matter how many times the detector fires in the target bin. On the other hand, dividing the signal into too few pulses causes detector saturation and reduces the chances that the signal will trigger the detector at least twice. Thus there is an optimum number of pulses for a given detection probability. For example, ten to fifteen pulses is optimum for 99% detection probability, and the minimum total signal required is eight photoelectrons—a little higher than the seven photoelectrons required when there is no noise.

For Figure 5(c), the threshold is three firings in a bin. Raising the threshold has reduced the probability that noise firings reach it, so the curves are flatter at the higher number of pulses. However, with a higher threshold, more total signal is needed to reach a given detection probability. Here it takes nine to ten photoelectrons to achieve a detection probability of 99%.

For Figure 5(d), the most-firings law is used. Compare this result to Figure 5(b). The detection-probability curves for the most-firings law are flatter at high values of n because the target bin is more likely to have more firings than any noise bin, even though one or more noise bins are likely to have at least two firings. The false-alarm probabilities at low values of n are higher because the number of target firings has

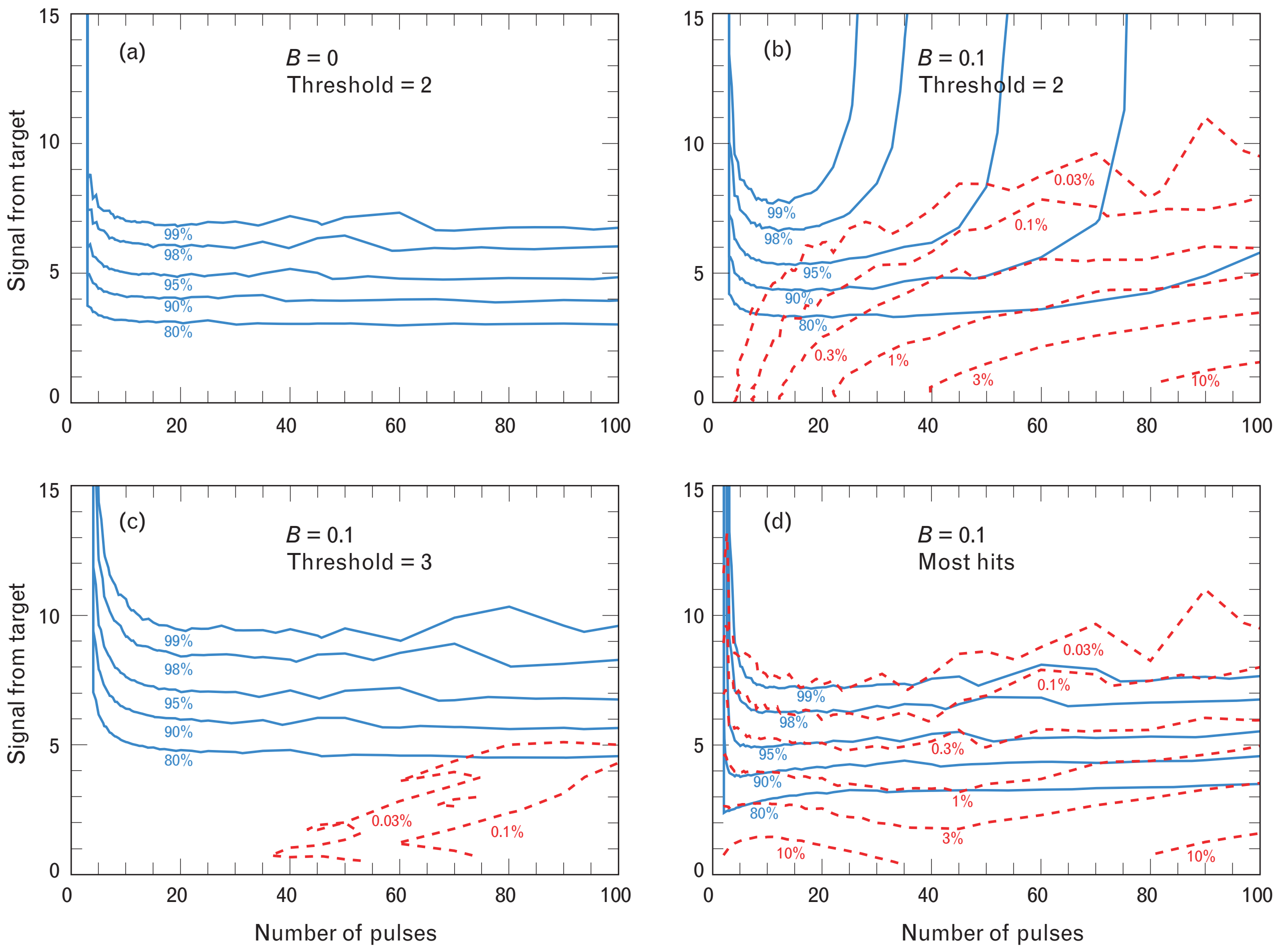

FIGURE 5. Results from multiple-pulse Monte Carlo simulations of Geiger-mode detection when there are no obscurants. The abscissa is the number of laser pulses n processed as a set. The ordinate is the total signal (i.e., the single-pulse signal level S times the number of pulses n) in photoelectrons. The blue and red curves are lines of constant probability of detection and false alarm, respectively, as labeled. Shown on each graph are the noise level B, in primary electrons per range gate, and the detection law (threshold law with a particular threshold t or most-hits law). For this and the following figure, the target is in the middle of the range gate, which is divided into 200 bins. Such graphs can be used to determine the optimum number of pulses over which to spread the total signal.

decreased more than the number of noise firings, owing to detector saturation for the target signal. At high values of n, the false-alarm probabilities are the same because the probability of a particular experimental result dominates the overall false-alarm probability for both the threshold law and the most-firings law; that experimental result is two firings in a single non-target bin with zero or one target firings.

Monte Carlo Results for an Obscured Target

For a Geiger-mode detector to see a target that is partially obscured by foliage or a camouflage net, the signal from the obscurant must be small. With this condition, the signal from the target will likely be small, too; therefore, the target will have to be illuminated many times.

In this section, we present results of Monte Carlo calculations for multiple-pulse illumination of a target behind an obscurant. We address only a single detector and assume that its field of view encompasses both obscurant and target. The obscurant can be an opaque surface such as foliage or camouflage that partially fills the field of view, or it can be a partially transmissive substance such as smoke, fog, or aerosol that fully fills the field of view.

The detection law that we use to determine target range from the range histogram is to choose the last bin for which the number of firings exceeds a threshold. With this law, the detection probability depends on the total signal from the obscurant but is independent of its range distribution, as can be deduced from Equation 3. The false-alarm probability, however, in general does depend on the distribution of the obscurant.

Figure 6 shows some results of Monte Carlo calculations. The abscissa is the number of laser pulses *n*, and the ordinate is *n* times the mean number of photoelectrons that would be created by the single-pulse target return if there were no obscurant. The curves are those of constant detection probability, as labeled. The probabilities of false alarm are not shown because, for the conditions of these graphs, they are simply one minus the corresponding detection probabilities. For all three graphs, the range-integrated return of the obscurant is nine times that from the target (90% obscuration), there are two hundred range bins, and the target is in the center of the range gate. The straight dotted lines (included in the bottom graph only) represent the mean number of photoelectrons per pulse from both target (10% of the total) and obscurant (90%). The three graphs differ in their values for the average number of photoelectrons from background light and dark current per range gate (*B*) and in the threshold used in the range-coincidence processing.

All the detection-probability curves in Figure 6 have an upper and a lower portion. The upper portion of each curve is set by the per-pulse signal from the obscurant. When the obscurant return gets as high as a few photoelectrons per pulse, the single-pulse probability of firing from the obscurant approaches one, leaving little chance for a target firing. The lower portion of each curve is set by the noise level *B* and the threshold and behaves similarly to the curves in parts *a*, *b*, and *c* in Figure 5. For *B* = 0 (top graph), the lower portion approaches the no-obscurant case as the number of pulses increases and therefore as the obscurant return per pulse decreases. As *B* increases (middle graph), the lower portions trend increasingly up with higher values of *n*, because post-target bins have more opportunities to build up to threshold, and those opportunities must be decreased by increasing the single-pulse laser return. These trends are counteracted by increasing the value of the threshold (bottom graph).

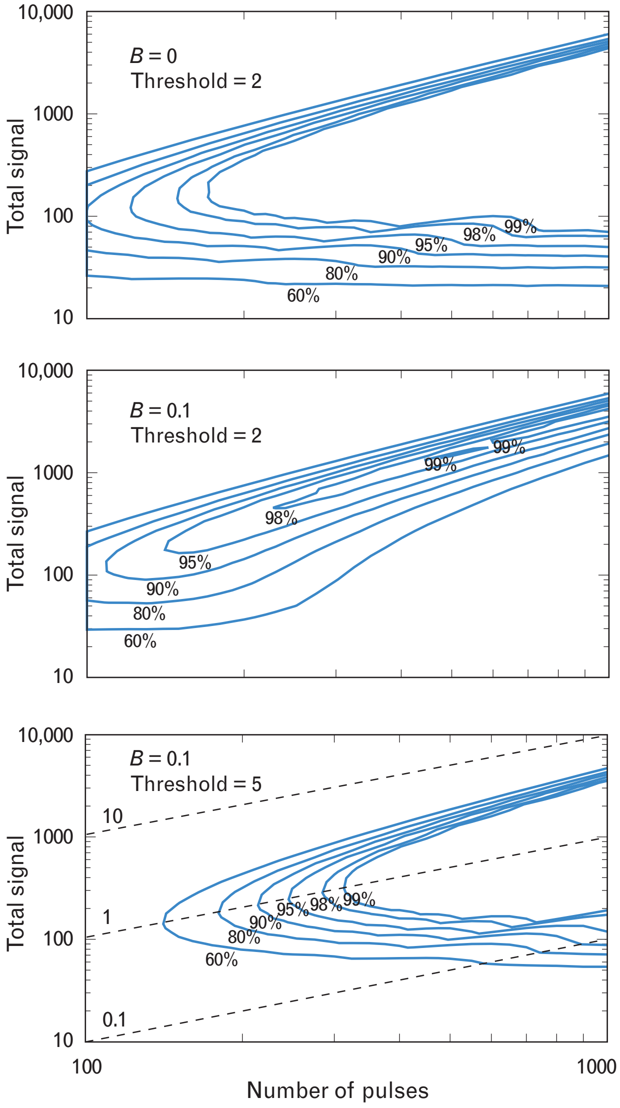

FIGURE 6. Results from multiple-pulse Monte Carlo simulations when the target is 90% obscured. The total signal combines target and obscurant. The blue curves are those for constant probability of detection, as labeled (false-alarm curves are not included). The straight dotted lines on the bottom graph represent the number of photoelectrons per pulse from both target (10% of total) and obscurant (90% of total). The detection law we use selects the last range bin for which the number of firings exceeds a threshold. The threshold and the noise level *B* are shown on each graph. The bottom graph shows that a 99% probability of detection can be obtained with under twenty photoelectrons from the target, despite the noise and 90% obscuration.

The graphs in Figure 6 indicate that seeing targets through obscurants is quite possible with a Geiger-mode detector, even when the obscurant return is nine times stronger than the target return. For example, the bottom graph for B = 0.1 and a threshold of five indicates that a 99% probability of detecting the target can be obtained with fewer than two hundred photoelectrons from obscurant and target (and fewer than twenty from the target alone) by dividing this signal over about five hundred to a thousand pulses. For Lincoln Laboratory ladars, which have typical pulse rates of order 10 kHz, a thousand pulses are obtained in about 0.1 sec.

We have found that the general characteristics of the graphs in Figure 6 are present for a wide range of values for noise level B and obscurant percentage. In general, the upper and lower portions of the constant-detection-probability curves get squeezed together as B or the obscurant percentage, or both, increase. Concomitant with the squeezing is a movement of the curves up and to the right on the graphs. Raising the threshold reduces the squeezing, but results generally in more movement up and to the right. A movement up corresponds to needing more total signal, and a movement to the right corresponds to dividing the total signal into more pulses, in order to obtain a given probability of detection. For an obscurant-penetrating ladar, we would like to maintain enough space between the upper and lower portions of the curves to accommodate reflectivity (i.e., signal level) differences from different pixels in a scene. Also, we must keep in mind that the obscurant percentage for foliage will vary from pixel to pixel.

Graphs such as those in Figure 6 helped Lincoln Laboratory design the Jigsaw 3D ladar described elsewhere in this issue of the *Journal*. Jigsaw data presented later in this article demonstrate obscurant penetration.

Multipixel System Simulation

The single-detector analysis presented in the previous section is the basis for the simulation of a ladar with a detector array viewing a complex scene. The flow chart in Figure 7 illustrates the relationships among the various elements of the simulation. The input scene at the top of the chart is rendered as a grid of pixels, each with a range and reflectivity. This rendered input scene is more highly resolved than the image provided by the ladar; therefore, we call the pixels of the input scene *micropixels*. For example, the input scene may have 4 × 4 micropixels falling within the field of view of a single detector. With this approach, we are able to allow for range and reflectivity variations within a single detector pixel, as may be required to simulate foliage or a camouflage net over an extended object on the ground.

The green boxes show the sources of light associated with each micropixel. The four sources are laser light and sunlight reflected both from the surface represented by the micropixel and from aerosol located between the ladar and micropixel. We account for the time profiles and optics-induced blur of each of these sources.

In the upper two blue blocks, we sum the time profiles of all the micropixels (say 4 × 4) associated with a particular detector, apply the proper scaling factor to convert from reflectivity to photoelectrons, and add a constant rate of primary electrons from dark current. In the lower two blue blocks, we compute the probability that each detector will fire versus time. This computation includes the effect of APD timing jitter, as represented by a temporal response function.

Next we use a Monte Carlo technique to yield the time that each detector fires on a laser pulse, including those firings from cross talk within the detector array. We repeat this step multiple times if the scenario involves multiple laser pulses in an essentially static situation.

At this point, coincidence processing can be performed on the multiple-pulse results for each detector to yield an image for display. This step may involve coordinate transformations. For example, the Jigsaw system converts from an angle-angle-range coordinate system to an earth-fixed Cartesian coordinate system before coincidence processing.

If the scenario is dynamic, because either the scene changes with time or the sensor moves relative to the scene with time, or both, then the process illustrated in Figure 7 is repeated multiple times. Each time, the input scene is changed slightly from the previous one according to the selected time increment between repeats. Coincidence processing and image display can be done with the combined results of multiple time steps.

The remainder of this section, along with the following four sections, describe the elements of the simulation in more detail. Examples are drawn mostly from simulations of the foliage-penetrating Jigsaw system discussed in more detail in a later section entitled "Jigsaw Case Study" and elsewhere is this issue of the *Journal* [5]. The primary objective of the Jigsaw system is to identify a single target under trees.

Input Scene Generation

The first step of the simulation process is to generate a scene that we want to view with a 3D ladar system. A scene typically consists of many different types of objects such as buildings, vehicles, and trees. Some objects (the vehicles) may be obscured by other objects (the trees). Since the ladar systems are usually mounted on a moving platform (such as an aircraft) when collecting data, we need to view the scene of interest from many different view points to simulate the motion of the platform. Any scanning subsystem is also simulated.

The scene-simulator program, written in C, uses the OpenGL graphics library to generate a scene. The program then renders computer-aided design (CAD) models of objects within OpenGL's depth buffer. This process utilizes the power of existing graphics cards for the 3D computations. The OpenGL depth buffer provides range from the sensor to the scene at a selectable angular resolution. The OpenGL library creates the proper perspective view with hidden surface removal. The flight path of the platform (at selectable temporal sampling) is an input to the code. The input scene remains the same for a fixed amount of time (defined by the temporal sampling), which results in a slightly inaccurate view of the scene for each laser pulse. We minimize this problem by sampling (in time) the scene more finely upon generation, with a cost of storing a larger file.

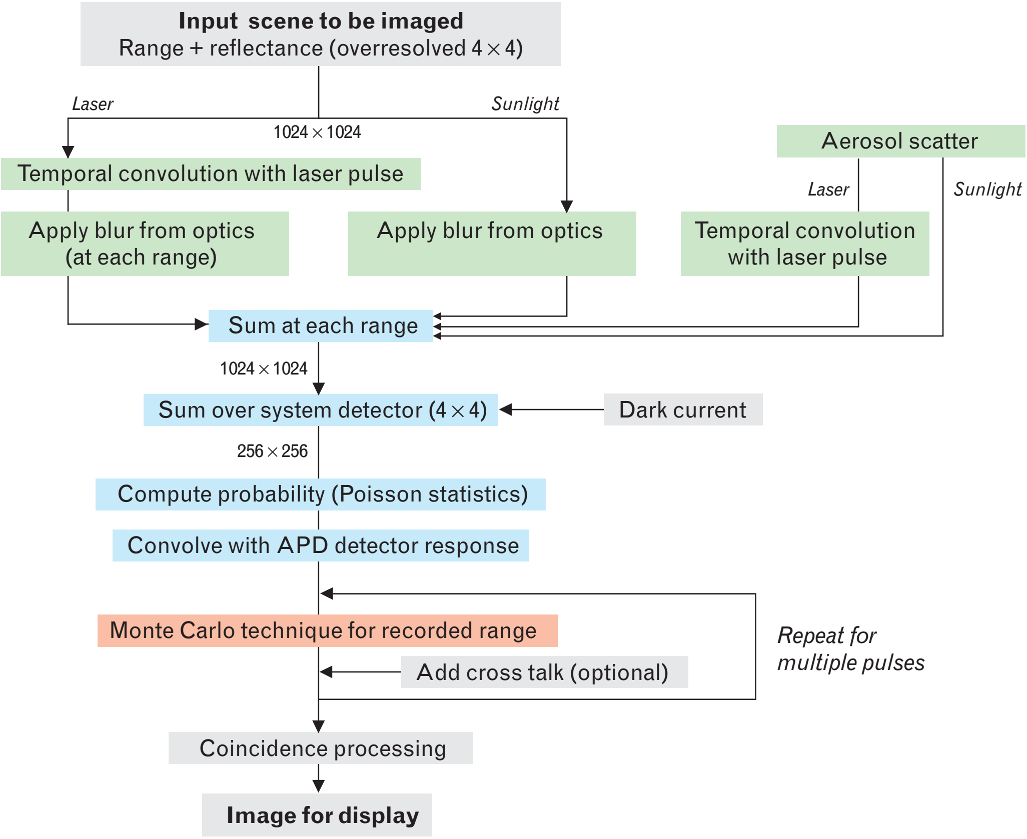
Input scene to be imaged
Range + reflectance (overresolved 4 × 4)
Laser
Sunlight
1024 × 1024
Temporal convolution with laser pulse
Apply blur from optics (at each range)
Apply blur from optics
Aerosol scatter
Laser
Sunlight
Temporal convolution with laser pulse
Sum at each range
1024 × 1024
Sum over system detector (4 × 4)
Dark current
256 × 256
Compute probability (Poisson statistics)
Convolve with APD detector response
Monte Carlo technique for recorded range
Add cross talk (optional)
Repeat for multiple pulses
Coincidence processing
Image for display

FIGURE 7. Flow chart of simulation steps for a static scenario, illustrating how the various elements fit together. The numbers given in the flow chart are typical image sizes used in the simulation (1024 × 1024 for the input scene and 256 × 256 for the APD array). The green blocks address the sources of signal and noise included in the simulation. These are combined for each range value and for each APD pixel, as shown in the first two blue blocks. The last two blue blocks are the steps used to calculate a CDF for each APD pixel. This CDF is used in combination with the Monte Carlo technique, as illustrated in the red block. The Monte Carlo and cross-talk steps are repeated for each laser pulse simulated. All these steps, including the generation of the input scene, are repeated for each time step.

The angular resolution of the scene is greater than the angular resolution of one APD pixel in order to provide range and reflectivity variations within each APD pixel. This additional resolution is especially important when we are simulating foliage, because we need to represent a variety of ranges (from the depth of the tree canopy and whatever is under it) and reflectivities (from the near-random angle of incidence of the foliage) within a single APD pixel. Typically, we divide each APD pixel into 4 × 4 or 8 × 8 micropixels.

For each surface element in the scene, we specify a reflectivity at normal incidence. We assume that the object surfaces are Lambertian scatterers so that the reflectivity of the surface is proportional to the cosine of the angle of incidence of the laser. Within the C code, this calculation is implemented by color coding the reflectivity in one of the OpenGL color buffers. Since each polygon in the CAD model is color coded, we can also have different reflectivities within an object (leaves versus bark on trees, for example), provided each polygon in the model has a material type assigned to it. In the graphics card there is a one-to-one correspondence between pixels in the depth buffer and the color buffer. Figure 8 shows an example scene from the scene-generation process.

Signal and Noise Sources Included in the Simulation

There are many sources of primary electrons that can trigger an avalanche (causing a firing) within the APD. These sources are divided into signal (the laser return reflecting from the target) and noise (everything else). This section discusses the various sources we include in the simulation.

Laser Return from Hard Target

Signal photoelectrons result from the reflection of the laser from the surface of an object. The previous section described how we model the relative reflectivity within our scene-generation step. This relative reflectivity is scaled by the product of the average reflectivity of the scene of interest and the average number of photoelectrons received from the target by our system.

The average number of photoelectrons derived from the target is itself a product of many system parameters, including average power of the laser, quantum efficiency of the detector, atmospheric transmission, receiving optics aperture size, and spectral-filter bandpass. The number of photoelectrons depends on the inverse square of range. The simulation code was written to take aver-

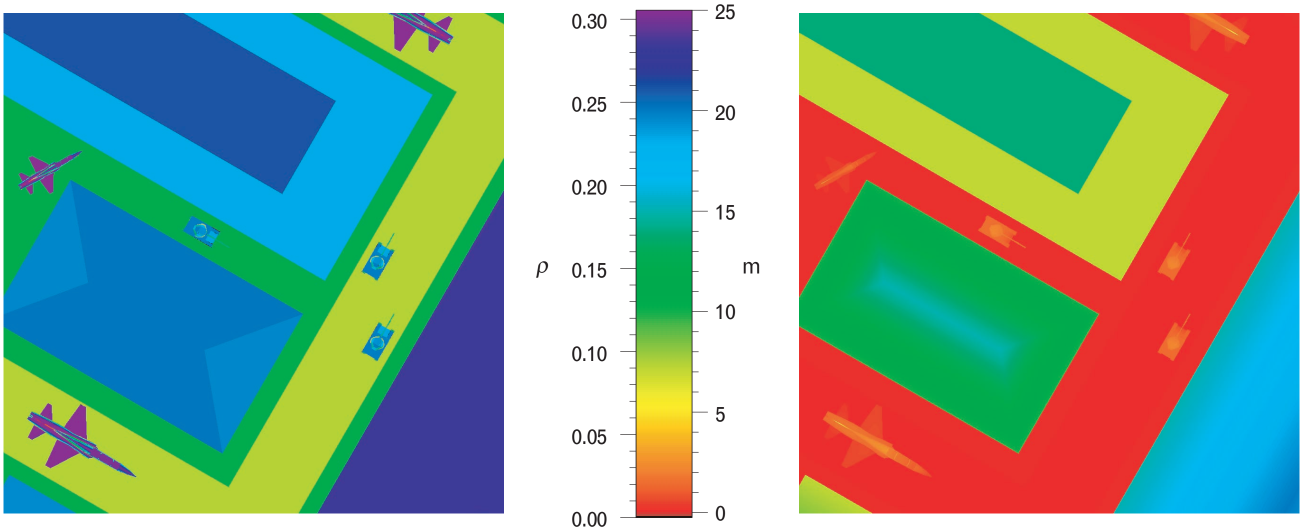

FIGURE 8. Images of color-coded relative reflectivity (left) and color-coded height (right) generated by our scene-generation code. The color bar shows the correspondence between color, reflectivity ρ, and height in meters. The relative reflectivity image shows the cosine dependence on the angle of incidence of the surface of each surface element. These images are example inputs into the APD simulation.

age photoelectron rate (at a particular range) as input, so the calculation using the system parameters is done externally to the code.

The laser return also has a temporal distribution. We currently assume a truncated Gaussian temporal distribution for the transmitted laser pulse, but other distributions could easily be modeled. With the Gaussian distribution, we specify the width of the pulse.

Transverse intensity variations across the laser beam are also included in the simulation. Our simulated laser source has a selectable spatial distribution at the scene. We currently have implemented three different distributions: uniform, Gaussian, and an array of Gaussian spots (used in the Jigsaw system). Any other spatial distribution could easily be modeled in the simulation.

Sunlight Reflected from Hard Target

The main source of background photoelectrons in the simulation comes from sunlight reflecting from the scene. As a result, we again use the reflectivity of the model, just as we do for laser reflections. Here the relative reflectivity modeled in the scene generation step is scaled by the expected number of photoelectrons from solar reflections. This value is again the product of many of the systems parameters listed above (excluding laser power), in addition to atmospheric parameters such as time of day and relative angle between sensor and sun. Unlike the laser return, however, this source of background photoelectrons has a constant mean rate because the sun is a continuous source of photons.

Scatter by Atmospheric Aerosols

Aerosols scatter both sunlight and laser light. Aerosol-scattered sunlight photons are uniformly distributed in both time and space. Backscattered laser photons are modeled in the same way, except the photons come only from the region in front of the target (between the laser and the target). The simulation accounts for attenuation of the two laser signals (reflections from a hard target and backscatter from aerosols) by aerosol absorption.

Dark Current

Dark current, created by thermal effects within the APD detector material, is assumed to be uniformly distributed in time. All APD pixels are assumed to have the same dark current.

APD Simulation

Now that we have generated an input scene and characterized the sources of photoelectrons and dark current, the next step in the simulation is to define the APD parameters. These include the dimensions of the APD array in pixels, the width of the range gate (i.e., the length of time the APD is "armed"), the size of the range bin (i.e., the temporal resolution of the digital timing circuitry integrated with the APD), and the size and shape of the APD area that is sensitive to photons. We also simulate the blurring, if any, caused by the optics of the system.

For each range bin and for each APD pixel, we add all of the sources of photoelectrons (as described above) and dark current for each micropixel, and then add the contributions from each micropixel in the APD pixel. We next calculate the cumulative probability distribution function, as described earlier, and then "throw the dice" to determine when, if at all, each detector fires on a given laser pulse. This process is repeated a number of times (set by the laser pulse rate) for each rendered image corresponding to a discrete time step (the scene may vary from step to step, owing to motion of the sensor). Typically, the time step is a fraction of a second. Figure 9 shows an example return from a 256 × 256 element APD for a single laser pulse for the scene shown in Figure 8 (from the scene generator). Black pixels represent APD array elements that did not fire, and white pixels represent those that fired from noise rather than from the laser return from the target.

As discussed earlier, we can do range-coincidence processing with either multiple pulses or multiple detectors or both. Figure 10 shows the results of range-coincidence processing with fifteen laser pulses. The image on the left maintains 256 × 256 elements in the processing. The image on the right combines each 2 × 2 set of elements to create a 128 × 128 image with much better fill.

Additional Features in the System Simulation

We can supplement the simulation with other system processes such as pointing jitter, scanning, range tracking, APD cross talk, and APD response function (which characterizes timing jitter). These additional effects are discussed in more detail in this section.

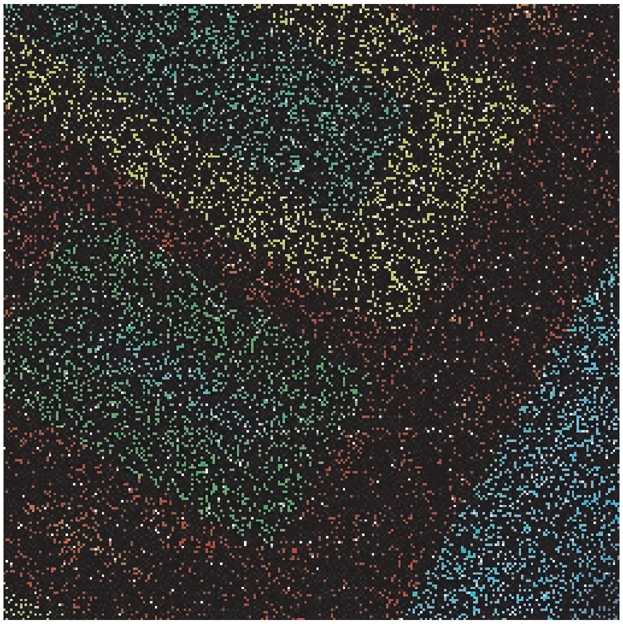

FIGURE 9. Sample single-pulse output of the simulation for a 256 × 256 element APD. Black pixels represent APD array elements that did not fire, and white pixels represent those which fired from noise rather than laser return from the target. The input scene is that shown in Figure 8, and the image is color coded by height, as in Figure 8.

APD Response Function

Inherent in the physics of the APD is a random delay between photon absorption and full development of the avalanche. The time distribution of this delay is called the APD response function and is included in the simulation. The system response function (laser pulse width plus APD response function plus range discretization) is measured by collecting data while imaging a flat plate. Ideally, a flat plate would give the same range measurement with each laser pulse. The spread in the ranges defines the system response function. For the Jigsaw system, the system response function is dominated by the APD response function.

For the simulations presented in the next section we use a piece-wise analytic function to model the APD response function. Cross-talk effects are incorporated into this function, but are negligible. Figure 11 compares simulated (shown in red) and measured (shown in black) range histograms for a flat plate (the system response functions). The figures show four different real pixel histograms. Although slight variations occur

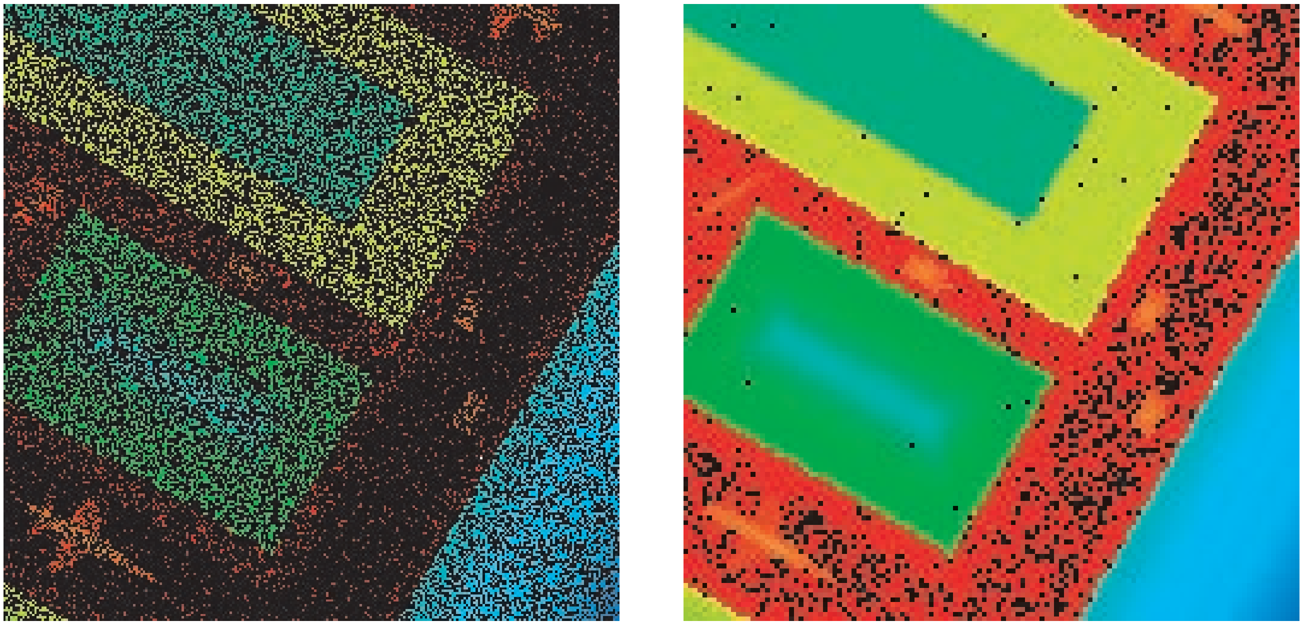

FIGURE 10. Images showing results of range-coincidence processing with fifteen laser pulses. The fifteen frames (laser pulses) are independent realizations, one of which is shown in Figure 9, and are color coded by height, as in Figure 8. The left image maintains 256 × 256 APD elements in the processing. The right image combines each 2 × 2 set of APD elements to create a 128 × 128 image with much better fill rate. These images illustrate that during post-processing a trade-off can be made between resolution (with lower fill rates) and fill rate (with lower resolution), and both can be done with the same data.

from pixel to pixel in the measured data, we assume in the simulation that all pixels have the same system response function. Thus the four red curves are identical.

Cross Talk

Cross talk is quantified as the probability versus time that a particular neighboring APD pixel will fire after a given APD pixel fires. We made measurements of cross talk of actual APD arrays and implemented the results in the simulation in the following way. After we generate a simulated frame of APD data (for a single laser pulse), we add additional detections with the desired probabilities and temporal distributions.

Figure 12 shows a comparison of simulated data to measured data. The graph is a range histogram of data from a flat plate. From the measured data we estimated the number of photoelectrons from the target and the noise, assuming no cross talk was present, which inflated the estimates. Then we ran a simulation without cross talk included (blue curve). Although the simulated and actual data agree for ranges up to and including the plate, the simulation shows more detections after the plate than were actually measured. These extra detections are due to the inflated noise estimate. We then incorporated cross talk, adjusted the estimated signal and noise levels appropriately, and ran a new simulation (red curve). We can see that the new simulation and the measured data generally agree within the shot-noise scatter of the data points. Note that this APD had a particularly large amount of cross talk that is not typical of most of our APDs.

Scanning

The 3D ladar systems we are currently using incorporate 32 × 32 APD arrays. In order to build an image with more pixels we must scan the APD array over the scene. Different scanning schemes can be evaluated with the simulation. In order to accommodate scanning, the input scene we generate must be much larger than the field of view of the APD array. Then, on each laser pulse, we look at a particular subscene as defined by the scanning pattern.

Pointing Jitter

On any moving platform that is tracking a target, pointing jitter is an important factor in system performance. In our simulation we have implemented several jitter models, including Gaussian jitter, linear drift plus

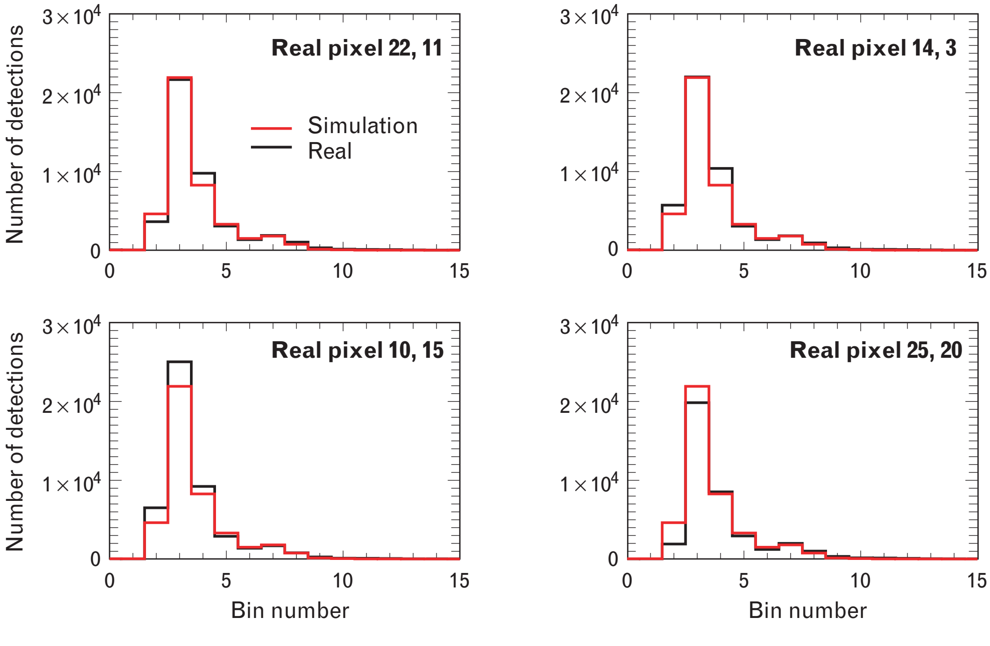

FIGURE 11. Comparison of simulated versus measured range histograms. The black curves show measured system response functions of four APDs in a 32 × 32 array. The red curve, which is identical for all four pixels, is the system-response function used in simulations involving this system. The pixel-to-pixel variations in the measured functions are slight enough to be ignored in the simulation.

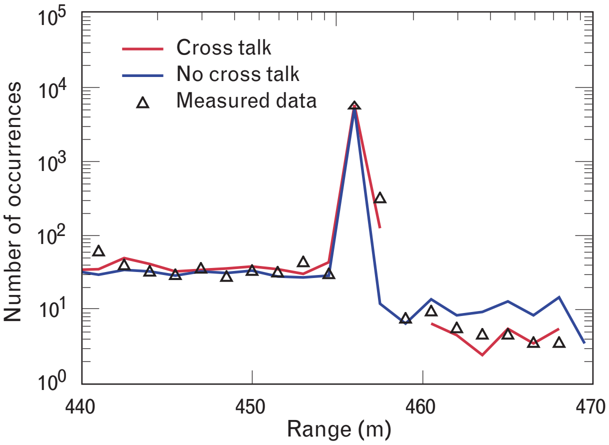

FIGURE 12. Range histograms of APD data for an image of a flat plate. The measured data points (triangles) are taken from an actual APD with an unusually high amount of cross talk. The simulated data without cross talk included (blue curve) match the measured data only before and at the flat plate, but not after the flat plate. Adding a model of cross talk to the simulation (red curve) yields good agreement with the measured data at all ranges.

Gaussian jitter, and colored jitter. With colored jitter we use the actual or predicted spectrum of a real system. To implement jitter, on each laser pulse we shift the scene by a particular number of micropixels in each dimension.

Range Tracking

Range tracking entails real-time processing of either APD data or GPS data to keep the targets of interest within the range gate of the system, preferably near the front in order to minimize noise effects. On a moving platform this restriction may require that the system track the range to the target. Various range-tracking algorithms can be explored with the simulation.

Jigsaw Case Study

The objective of the Jigsaw program, sponsored by the Department of Defense Advanced Research Projects Agency (DARPA), is to locate and identify obscured targets within a small area (about twenty meters square) from an air vehicle at low altitude (about a hundred meters). Figure 13 illustrates the basic concept of Jigsaw. Identification is aided by using angle diversity to maximize penetration of foliage. Data from various views are combined to generate a high-resolution (7.5 cm) 3D image of the target. Identification is done by comparing the 3D data against models of known targets of interest. Details of the Jigsaw program can be found elsewhere in this issue of the *Journal* [5].

We first show qualitative and quantitative comparisons of Jigsaw data to simulated data of a target in the clear. Later we show qualitative comparisons for a target under trees. For the target-under-trees case, it is impractical to attempt quantitative comparisons, because the trees for the Jigsaw experiment were not well characterized and were undoubtedly very complex objects.

The simulation includes the basic system characteristics of Jigsaw: signal level, noise level, APD response function, flight path, scanning, and data-processing algorithm. The signal level was derived from a calibration performed during the same flight as the data being simulated. The noise level was estimated from the actual data being simulated, since it depends on many parameters such as time of day. The measured cross talk for Jigsaw is small and is neglected in the simulation. Our model of the actual vehicle is not exact, but it is

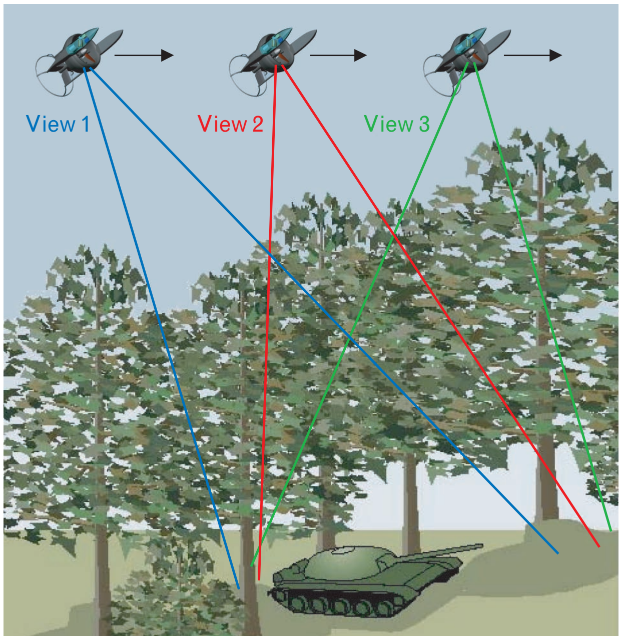

FIGURE 13. Concept of the Jigsaw program. The objective is to locate and identify obscured targets within a small area (about twenty meters square) from a small organic air vehicle at low altitude (about a hundred meters). Three-dimensional data from various views are combined in post-processing to generate a more complete 3D image of the target than that from any one view.

close (a minor feature at the rear of the turret is missing from our model). A model for the local ground was generated from the Jigsaw data. Jitter is not included in the simulation.

The Jigsaw scanning system uses two counter-rotating wedged (Risley) prisms. They create a scan pattern resembling a rotating rosette. This pattern is evident in the images included in this article. During the time that it takes to complete a rosette (¼ sec), the pattern rotates only a few degrees. The data collected over ¼ sec are processed together to yield an image frame. While we simulated the actual scan pattern, we did not try to match its phase with that of the actual data, and there is a slight phase difference.

The coincidence-processing algorithm used for the Jigsaw program is called voxel coincidence processing. It is based on the coincidence processing discussed earlier, except that it is performed in three dimensions. Basically, we compute a 3D histogram after we convert all the data points from an angle-angle-range coordinate system to a fixed earth-based Cartesian (*xyz*) coordinate system. Histogram bins, or *voxels*, can be ignored if too few data points (i.e., fewer than some threshold) land within it. This processing has been performed with a specified threshold on all of the results that follow.

Before comparing Jigsaw data to simulated data, we note that the actual Jigsaw images are slightly blurred and distorted. These image defects arise from uncertainties in sensor line-of-sight angles, aircraft position, and aircraft attitude during data collection. Errors in these metric quantities cause errors in placing data points in the earth-fixed Cartesian coordinate system. These errors are unknown and time dependent, and they cannot be entirely removed by translating and rotating (registering) one frame with respect to another. They are not included in the simulations. We believe that these unmodeled errors explain many of the differences between the simulated and actual images shown in this article.

Figure 14 shows the real Jigsaw image data (top row)

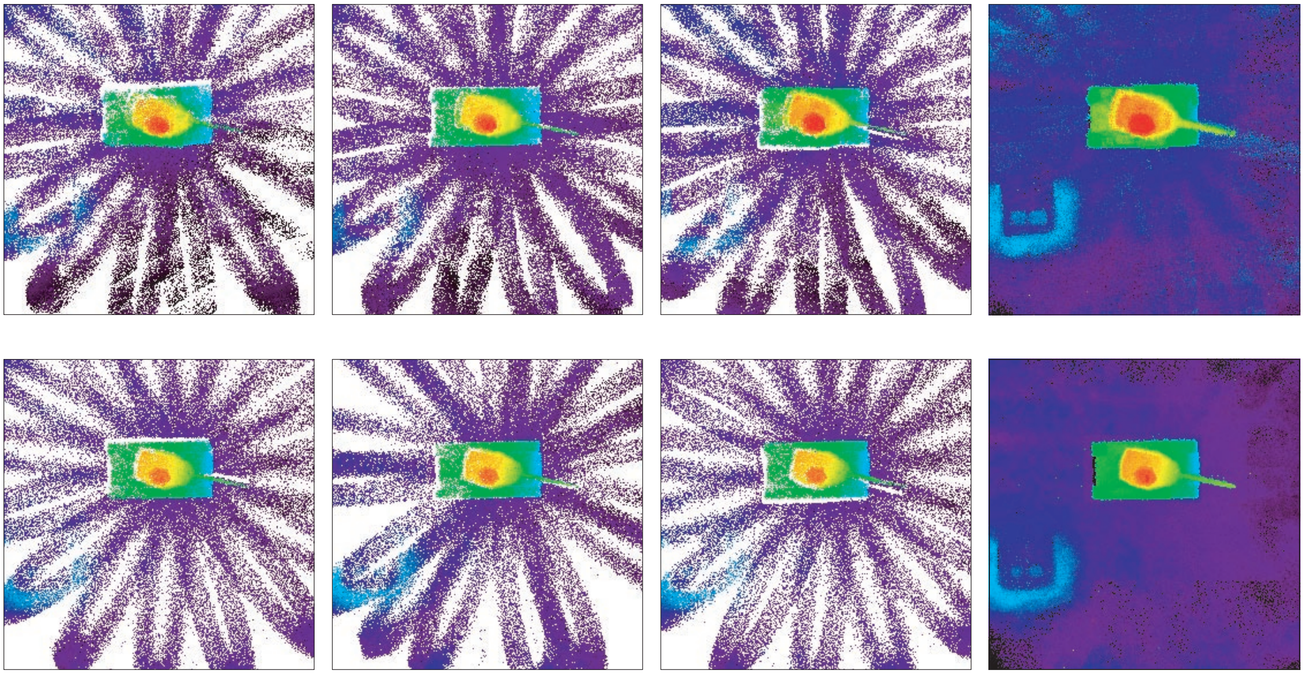

FIGURE 14. Color-coded height images of real Jigsaw data of an M-60 tank in the clear (top row) and corresponding images of simulated Jigsaw data (bottom row). The first three images in each row are individual ¼-sec frames. The fourth image in each row is the composite of eighteen ¼-sec frames. The data are viewed from directly above, although the individual frames were collected from different views. The flight path starts south of the tank (north is at the top of the page), resulting in the shadow above the tank in the first image. The flight path proceeds due north, passing over the tank, and ends to the north of the tank in the third image, resulting in the shadow below the tank in that image. Images from the simulation are clearly a good qualitative match to the images from the real data.

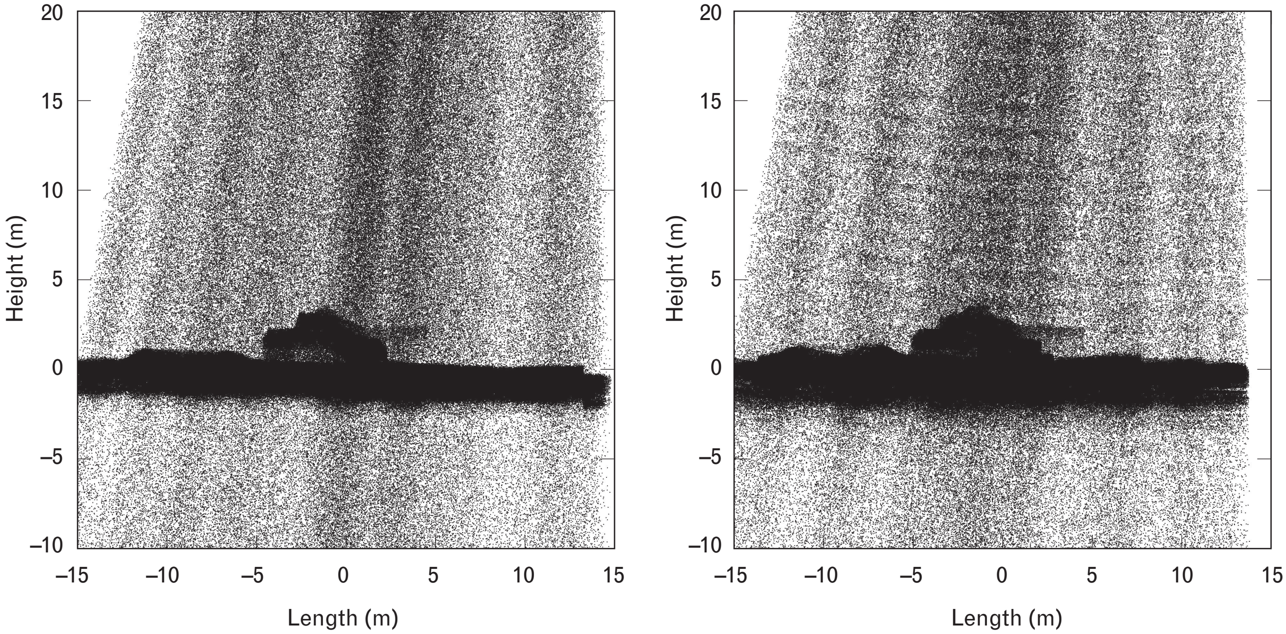

FIGURE 15. Scatter plots of simulated (left) and real (right) Jigsaw data, using voxel coincidence processing with a threshold of one. These are the same data as shown in the second frame in each row of Figure 14. The differences arise mostly from two effects: blurring of the real data due to uncertainties in pointing and aircraft position and attitude, and the difference in phase of scanning between the real and simulated data. This phase difference causes different ground regions to be illuminated, resulting in different profiles. Overall, however, this figure shows good qualitative agreement between simulated and real data.

and the simulated image data (bottom row) of an M60 tank in the clear. The color coding represents height above ground. These images have been processed with a threshold of two (i.e., at least two detections per voxel). The four frames in each row represent three different measurement times during the flight and the composite image of all eighteen frames. The real data (top row) have errors arising from the metric uncertainties mentioned in the previous paragraph. For example, height errors are indicated by the barely perceptible differences in the ground (indicated by slight differences in color) in the individual frames in the top row and by the appearance of the rosette pattern in the composite frame. Comparing now the real (top row) and simulated (bottom row) images, we see that the phases of the rosette scan patterns differ. Other differences between real and simulated data are small and are most apparent at the back end of the turret where our CAD model did not match the real tank.

Figure 15 shows the data from the second frame in each row of Figure 14 (the view closest to nadir) in another representation. Figure 15 is a scatter plot of all the data (processed by using voxel coincidence processing with a threshold of one) as a function of height and *x*-axis position. The left frame is simulated data and the right frame is real data. The cloud of points outside the black areas of the tank and ground is noise from dark current and background light. The left-to-right variation of the noise arises from the irregular scan pattern of the Risley prisms. The horizontal stripes in the real data are caused by unmodeled minor imperfections in the real focal-plane array.

The differences between the real and simulated data arise mostly from the two effects mentioned earlier: the uncertainties in pointing and aircraft position and attitude, and the scan-pattern phase difference between the real and simulated data. The position and attitude uncertainties cause a blurring of the ground layer, making it appear thicker. The phase difference in the scanning results in different parts of ground being illuminated for any given ¼-sec frame. This difference in illumination, when projected horizontally (as in Figure 15) results in different profiles. The difference in the two scatter plots is most evident at the feature between lengths –13 m and –5 m. This feature is the projection of a mound on the ground (seen in the lower left part of the images

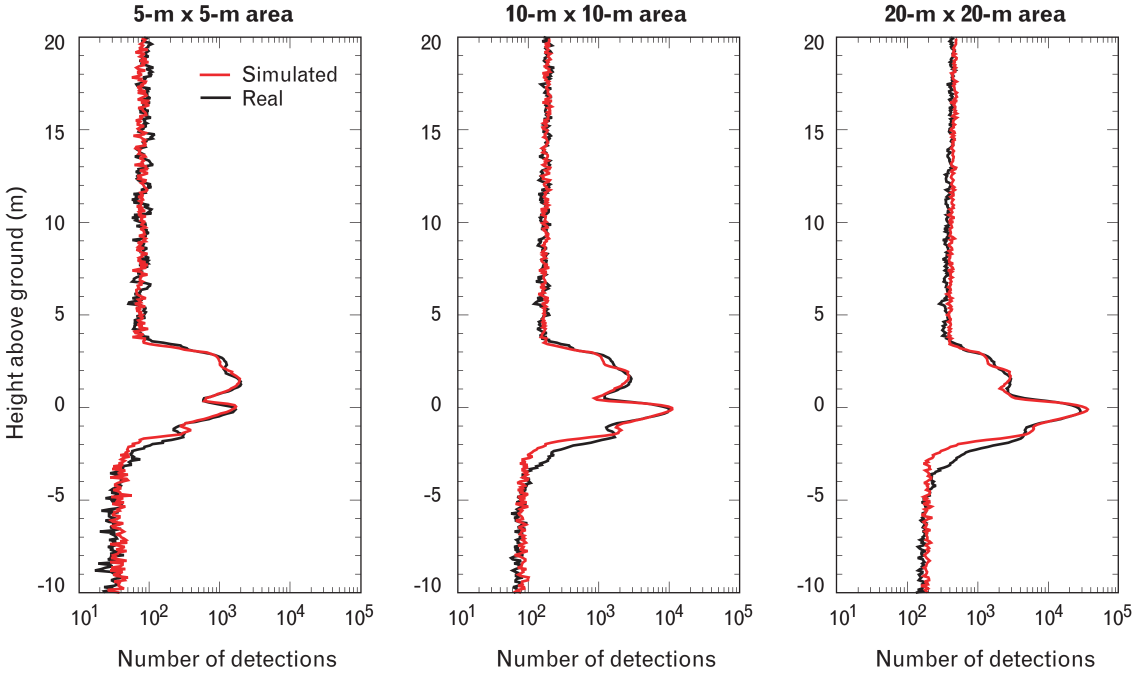

FIGURE 16. Height histograms of the Jigsaw data shown in Figure 15. Three different regions centered about the tank are shown: 5 × 5, 10 × 10, and 20 × 20 meters. There is generally good agreement between the simulated data (shown in red) and the real data (shown in black).

in Figure 14). For the simulated data (in the left plot of Figure 15), the entire length of the mound is illuminated, making a flat profile, whereas for the real data (in the right plot), only the two sides are illuminated, resulting in the two humps.

While Figures 14 and 15 show qualitative agreement between simulated and real data, Figure 16 provides a quantitative comparison by showing the histograms (in the height dimension) of the data presented in Figure 15. Three different *xy* regions are shown in order to sample non-uniformity in the scan pattern. We can see that there is generally good agreement between the real Jigsaw data (shown in black) and the simulated data (shown in red). The discrepancy at heights near –2 m is probably caused by the unmodeled metric uncertainties mentioned earlier, and by differences between the actual and modeled ground. The extra fluctuations in the real data above four meters, which correspond to the horizontal stripes in Figure 15, are caused by unmodeled imperfections in the real focal-plane array.

One measure of the performance of the simulation is based on how well simulated data and real data can be registered to a CAD model. The iterative closest

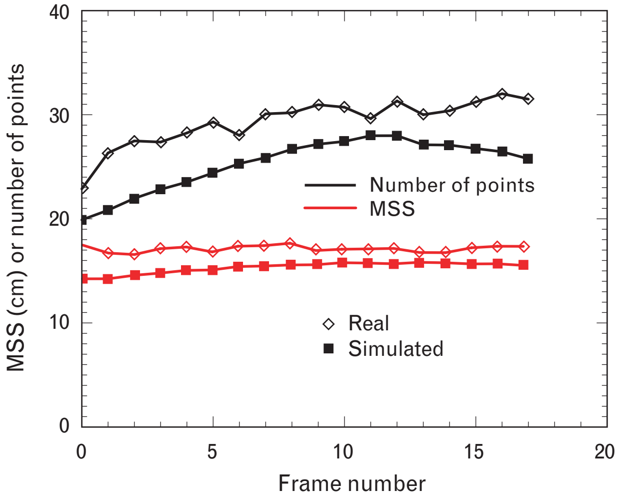

FIGURE 17. Goodness-of-fit measures of iterative closest point (ICP) registration for both simulated and real data for all eighteen frames simulated. The mean-squared separation (MSS) is shown in red, and the number of matched points (divided by 1000) is shown in black. These two measures for the simulated data are within about 20% of the measures for the real data. The discrepancies arise largely from not modeling the blurring of the real data caused by uncertainties in pointing and the relative position and attitude of the aircraft. This result shows that simulated data can probably be reliably used to study registration-algorithm performance.

point (ICP) algorithm [12–14] is a method commonly used for registration. Briefly, given two sets of points, the ICP algorithm pairs each point in one set with the closest point in the other set, as long as their separation is below some selectable value. The algorithm then calculates the amount of translation and rotation to apply to one set of points to minimize the mean separation of the pairs of points. The algorithm iterates these steps until some error threshold is achieved. There are two primary goodness-of-fit measures that come out of an ICP registration of data points to points on a CAD model: the final number of data points that are paired with CAD-model points, and the final mean-squared separation between the pairs.

We ran an ICP algorithm to register the real data to a CAD model and then to register the simulated data to the same CAD model. Figure 17 shows the resulting goodness-of-fit measures. The figure shows mean-squared separation in red and the number of matched points (divided by 1000) in black for all eighteen ¼-sec frames. The agreement is generally very good.

Both measures are about 20% larger for the real data than for the simulated data. The explanation for this discrepancy is that the real-data image is blurred. In our Jigsaw data processing, we have seen that uncertainties in pointing and the relative position and relative attitude of the aircraft do indeed cause image blurring within a single ¼-sec frame. These effects are absent in the simulation, for which we assumed that pointing and the aircraft position and attitude were known exactly throughout the data collection. We verified that adding range blur to the simulation increases both the mean-squared separation and the number of matched points.

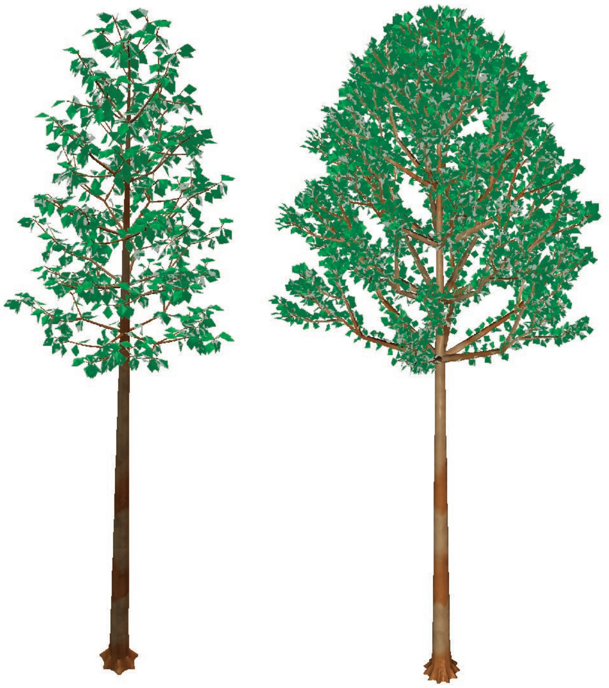

FIGURE 18. CAD models of trees from Arete Associates. A sugar maple is on the left and a yellow birch is on the right.

Table 1. Distribution of Trees over 0.1 Acres for 90% Obscuration at Nadir

Height of tree (ft)	*Sugar maple*	*Yellow birch*
20	5	1
30	5	1
40	10	2
50	16	3
60	5	1
70	12	3
80	5	1

Adding Obscurants

Arete Associates, a contributor to the JIGSAW program, was tasked to develop relevant tree models for the program. Arete gave Lincoln Laboratory CAD models of trees as well as a distribution of those trees which would provide 90% obscuration at nadir. The tree models consisted of two different species—yellow birch and sugar maple—at seven different heights. Figure 18 shows a sample of these two tree models. Table 1 shows the distributions of trees needed for 90% obscuration at nadir over one tenth of an acre (20 m × 20 m). The trees were distributed randomly in space with random rotations. This distribution was scaled to yield other obscuration ratios.

With the tree models provided by Arete Associates, we generated scenes that represented 71% and 98% obscuration at nadir. The 98% obscuration scene consisted of over six hundred trees. These scenes were com-

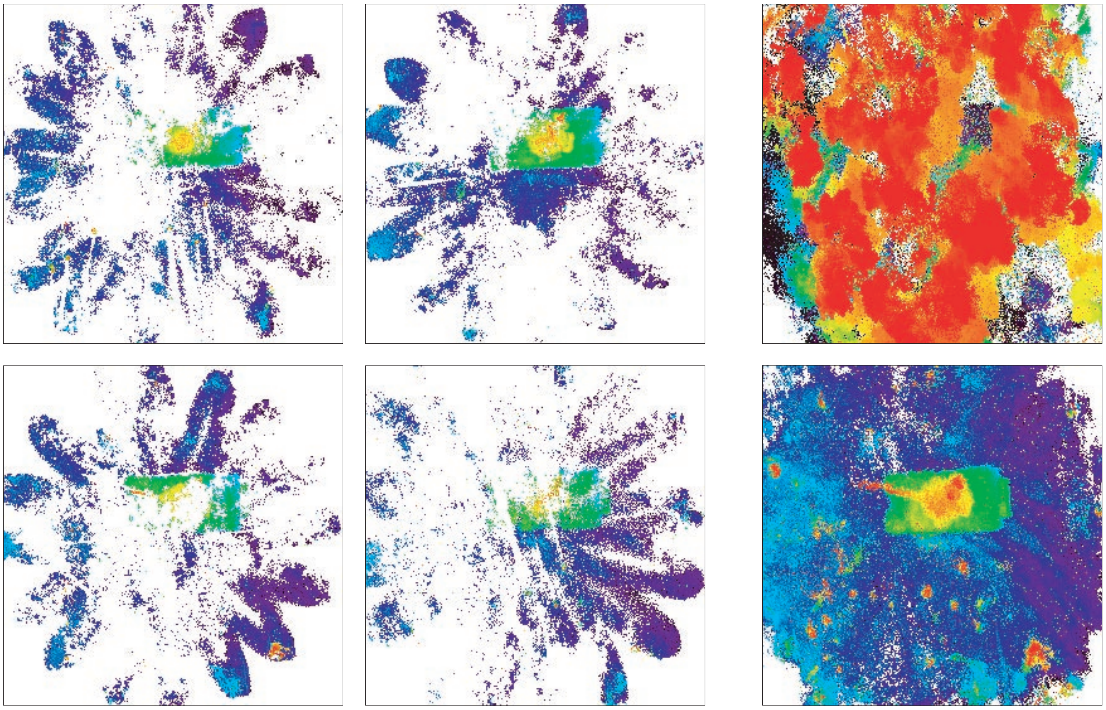

FIGURE 19. Color-coded height images of real Jigsaw data of a T-72 tank under 50% to 60% obscuration by foliage. The data are viewed from nadir even though they were collected from other angles. The four images on the left are individual ¼-sec frames of data, and the two images on the right are the composite of eighteen ¼-sec frames. The four images on the left and the one in the lower right have the canopy excised to show the underlying tank. The image in the upper right does not have the canopy excised and the tank cannot be seen.

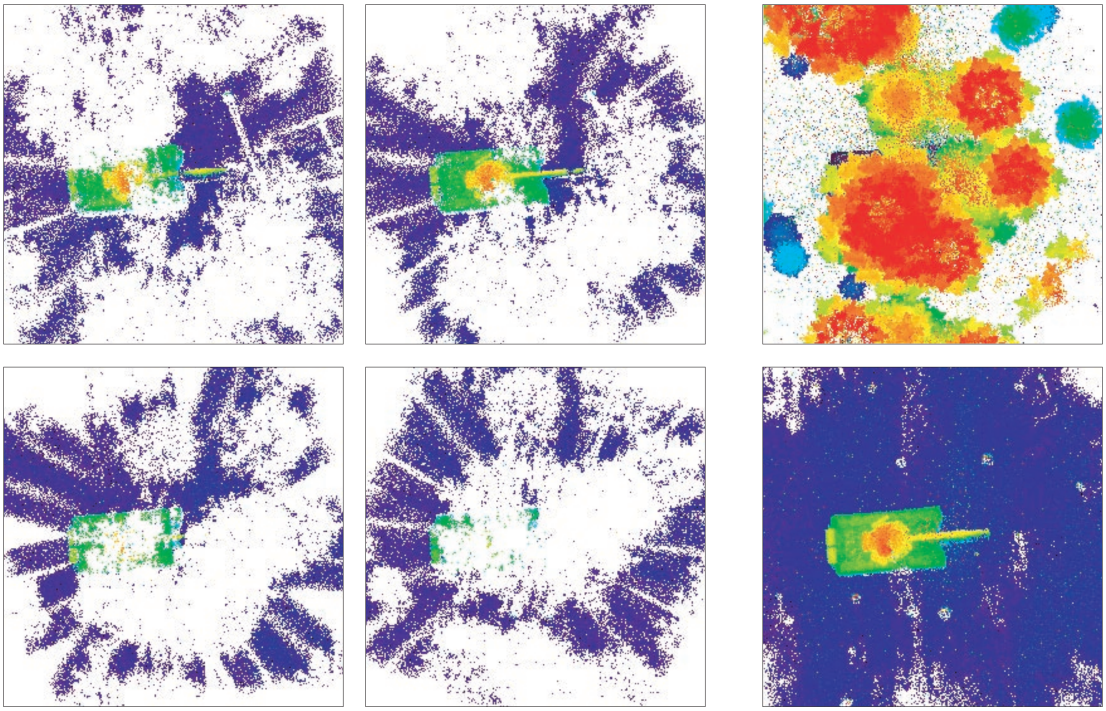

FIGURE 20. Color-coded height images of simulated data of a T-72 tank under 71% obscuration by foliage. The qualitative agreement between these images and the corresponding images in Figure 19 is very good.

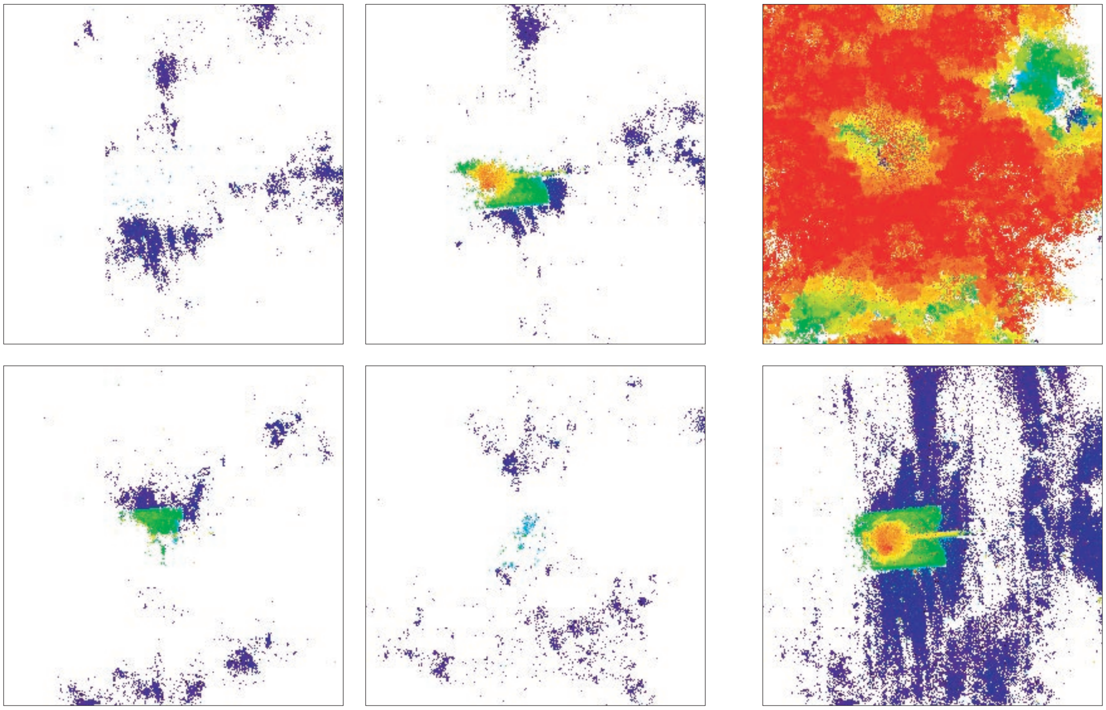

FIGURE 21. Color-coded height images of simulation data of a T-72 tank under 98% obscuration by foliage. The layout of the images is the same as that in Figures 19 and 20.

bined with a model of a T-72 tank. The flight path was the same as for the simulations used above.

Figure 19 shows real Jigsaw data of a T-72 tank under 50% to 60% obscuration by foliage. The images are color coded by height above ground. The data are viewed from nadir even though they were collected from other angles. The four images on the left side of the figure are individual ¼-sec frames of data with the tree canopy excised in order to show the underlying tank. The two images on the right are a composite of eighteen ¼-sec frames. The image in the lower right also has the canopy excised. The image in the upper right does not have the canopy excised and the tank cannot be seen. This figure clearly illustrates the benefits of looking at the tank from different viewpoints. Parts of the tank hidden by the foliage from one viewpoint may be revealed from another. Once the data points are converted from angle-angle-range sensor coordinates to earth-fixed Cartesian coordinates, the data from all viewpoints are combined to form a more complete image of the tank than that from any single viewpoint. It is the 3D nature of the data that allows us to aggregrate the data in this way.

Figure 20 shows simulation data of a T-72 tank under 71% obscuration by foliage. The layout of the figure is the same as the layout in Figure 19. The simulation parameters were not set to match exactly the real Jigsaw data shown in Figure 19; only the underlying object (the T-72 tank) is the same. The foliage in the Jigsaw experiments was not well characterized and was undoubtedly complex. The qualitative agreement of the real imagery (Figure 19) and simulated imagery (Figure 20) is very good. Notice that in the composite image, the real data is slightly blurred because of the effects discussed earlier.

Figure 21 shows the results for the 98% obscuration simulation. Again, the layout of the figure is the same as that in Figures 19 and 20. This simulation shows that penetration through even dense foliage is possible. While parts of the object may never be visible, combining data from different views may still allow target identification or target classification (for example, tank versus jeep).

Summary

We used Poisson statistics and a Monte Carlo technique to calculate the probabilities of detection and false alarm for a ladar that has a single Geiger-mode detector. Sets of parametric curves relate these probabilities to signal and noise levels, number of laser pulses, and obscuration ratio. Such curves have helped Lincoln Laboratory design 3D ladars that have been fielded.

The single-detector technique is the basis for a computer model that simulates the 3D images of a complex scene obtained by a ladar with an array of Geiger-mode detectors. We have used the computer model to help predict ladar performance and to develop data-processing techniques. All of the critical features of the Geiger-mode APDs are included in the simulation.

We performed a detailed validation of the simulation and found that simulation output agreed qualitatively with measured data from the DARPA-funded Jigsaw program. In addition to qualitative measures, we selected three quantitative measures for comparison. The first was a histogram of detections in the height direction. The other two came from performing an iterative closest point (ICP) registration with a CAD model. The ICP registration reports a mean-squared separation (MSS) and the number of points that were matched between the data and the CAD model. By these two measures, the agreement between real and simulated data is within 20%.

Acknowledgments

The authors wish to thank Glenn Boudreaux for the initial development of the scene generator, Ross Anderson for greatly expanding the flexibility and capabilities of the scene generator, and Alex Vasile for providing the ICP registration code. We also acknowledge the support of DARPA and the U.S. Air Force.

REFERENCES

1. R.M. Heinrichs, B.F. Aull, R.M. Marino, D.G. Fouche, A.K. McIntosh, J.J. Zayhowski, T. Stephens, M.E. O'Brien, and M.A. Albota, "Three-Dimensional Laser Radar with APD Arrays," *SPIE* **4377**, 2001, pp. 106–117.
2. J.J. Zayhowski, "Passively Q-Switched Microchip Lasers and Applications," *Rev. Laser Eng.* **29** (12), 1988, pp. 841–846.
3. J.J. Zayhowski, "Microchip Lasers," *Linc. Lab. J.* **3** (3), 1990, pp. 427–446.
4. B.F. Aull, A.H. Loomis, D.J. Young, R.M. Heinrichs, B.J. Felton, P.J. Daniels, and D.J. Landers, "Geiger-Mode Avalanche Photodiodes for Three-Dimensional Imaging," *Linc. Lab. J.* **13** (2), 2002, pp. 335–350.
5. R.M. Marino and W.R. Davis, "Jigsaw: A Foliage-Penetrating 3D Imaging Laser Radar System," in this issue.
6. M.A. Albota, B.F. Aull, D.G. Fouche, R.M. Heinrichs, D.G. Kocher, R.M. Marino, J.G. Mooney, N.R. Newbury, M.E. O'Brien, B.E. Player, B.C. Willard, and J.J. Zayhowski, "Three-Dimensional Imaging Laser Radars with Geiger-Mode Avalanche Photodiode Arrays," *Linc. Lab. J.* **13** (2), 2002, pp. 351–370.
7. M.A. Albota, R.M. Heinrichs, D.G. Kocher, D.G. Fouche, B.E. Player, M.E. O'Brien, B.F. Aull, J.J. Zayhowski, J. Mooney, B.C. Willard, and R.R. Carlson, "Three-Dimensional Imaging Laser Radar with a Photon-Counting Avalanche Photodiode Array and Microchip Laser," *Appl. Opt.* **41** (36), pp. 7671–7678.
8. R.M. Marino, T. Stephens, R.E. Hatch, J.L. McLaughlin, J.G. Mooney, M.E. O'Brien, G.S. Rowe, J.S. Adams, L. Skelly, R.C. Knowlton, S.E. Forman, and W.R. Davis, "A Compact 3D Imaging Laser Radar System Using Geiger-Mode APD Arrays: System and Measurements," *SPIE* **5086,** 2003, pp. 1–15.
9. J.W. Goodman, *Statistical Optics* (Wiley, New York, 1985).
10. K.A. McIntosh, J.P. Donnelly, D.C. Oakley, A. Napoleone, S.D. Calawa, L.J. Mahoney, K.M. Molvar, E.K. Duerr, S.H. Groves, and D.C. Shaver, "InGaAsP/InP Avalanche Photodiodes for Photon Counting at 1.06 μm," *Appl. Phys. Lett.* **81,** 2505–2507 (2002).
11. D.G. Fouche, "Detection and False-Alarm Probabilities for Laser Radars That Use Geiger-Mode Detectors," *Appl. Opt* **42** (27), pp. 5388–5398.
12. P.J. Besl and N.D. McKay, "A Method for Registration of 3-D Shapes," *IEEE Trans. Pattern Anal. Machine Intell.* **14** (2), 1992, pp. 239–256.
13. Z. Zhang, "Iterative Point Matching for Registration of Free-Form Curves and Surfaces," *Int. J. Comput. Vis.* **13** (2), 1994, pp. 119–152.
14. A.N. Vasile and R.M. Marino, "Pose-Independent Automatic Target Detection and Recognition Using 3D Laser Radar Imaging," in this issue.

MICHAEL E. O'BRIEN
is an associate staff member in the Active Optical Systems group. He joined Lincoln Laboratory in 1989 after receiving a B.S. degree in mathematics from Rensselaer Polytechnic Institute. He received an M.S. degree in applied mathematics from Worcester Polytechnic Institute in 1998. His work has focused on simulations and image and signal processing. The simulations have been of high-energy laser propagation and 3D laser radars. The image and signal processing has focused on passive optical and electro-optical systems, including 3D laser radar data.

DANIEL G. FOUCHE
is a senior staff member in the Active Optical Systems group. He joined Lincoln Laboratory in 1972 after receiving an A.B. degree in engineering and applied physics from Harvard University and a Ph.D. degree in applied quantum physics and quantum electronics from Yale University. His doctorate concerned inelastic light scattering from molecules. At the Laboratory, he has contributed to systems studies, hardware development, and experimentation in several electro-optical areas, including high-energy-laser propagation, precision tracking, adaptive optics, passive IR imaging, nonlinear optics, and laser ranging. Recently, he has concentrated on imaging laser radars.