Conf-930445--10

LA-UR- 93-1199

Title: DETECTING SMALL DEBRIS USING A GROUND-BASED PHOTON COUNTING DETECTOR

Author(s): C. Ho
W. C. Priedhorsky
M. H. Baron

RECEIVED
MAY 06 1993
OSTI

Submitted to: SPIE
Orlando, FL
April 12-16, 1993

MASTER

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

Los Alamos
NATIONAL LABORATORY

Form No. 836 R5
ST 2629 10/91

Detecting small debris using a ground-based photon counting detector

Cheng Ho, Bill Priedhorsky, Miles Baron

Los Alamos National Laboratory,
MS D436, Los Alamos, NM 87545

ABSTRACT

We describe a sensitive technique for detecting small space debris that exploits a fast photon-counting imager. Microchannel plate detectors using crossed delay-line readout can achieve a resolution of 2048×2048 spatial pixels and a maximum count rate of about 10^6 photons per second. A baseline debris-tracking system might couple this detector to a 16-cm aperture telescope. The detector yields x, y, and time information for each detected photon. When visualized in (x, y, t) space, photons from a fast-moving orbital object appear on a straight line. They can be distinguished from diffuse background photons, randomly scattered in the space, and star photons, which fall on a line with sidereal velocity. By searching for this unique signature, we can detect and track small debris objects. At dawn and dusk, a spherical object of 1.3 cm diameter at 400 km will reflect sunlight for an apparent magnitude of $V \approx 16$. The baseline system would detect about 16 photons from this object as it crosses a 1 degree field of view in about 1 second. The line in (x, y, t) space will be significant in a diffuse background of $\sim 10^6$ photons. We discuss the data processing scheme and line detection algorithm. The advantages of this technique are that one can 1) detect cm-size debris objects with a small telescope, and 2) detect debris moving with any direction and velocity.

In this paper, we describe the progress in the development of detector and data acquisition system, the preparation for a field test for such a system, and the development and optimization of the data analysis algorithm. Detection sensitivity would currently be constrained by the capability of the data acquisition and the data processing systems, but further improvements could alleviate these bottlenecks.

1. DETECTION CONCEPT

During dawn and dusk, a telescope located on the night side of the earth can detect sunlight reflected by an object in low-earth orbit. The orbiting object will appear to be moving at a high velocity relative to a fixed background of stars and diffuse light. This signature is unique to fast-moving foreground objects and can be exploited for the detection and tracking of space debris. It is, however, difficult to detect small objects with an imaging detector collecting 2-dimensional (2D) data: the signal from a small debris, which is a faint track with length corresponding to the image integration time, will be overwhelmed by the background. With the advance of fast imaging photon counting detector, the data can be collected in a 3-dimensional (3D) format, i.e. (x, y, t) of individual photons. This additional dimensionality greatly enhances the statistical significance of linear features in the data. Figure 1 shows a schematic comparison between the significance of a line in 2D and 3D data sets.

For photons randomly distributed in a volume V with density ρ, the mean number of photons contained in a line of length L is $\langle n \rangle = \rho V_1$ where $V_1 = LA$ is the "volume" of the line with A being the "cross section" of the line. Thus the number of lines of length L consisting of n photons is given by

$$K = \frac{N_L(V_1)}{n!} \times (\langle n \rangle)^n \exp(-\langle n \rangle), \qquad (1)$$

and $N_L(V_1)$ is the number of lines occupying the volume V_1. The cross section A of the line can be larger than 1 square pixel in digitized data, depending on the "operational" definition of the line.

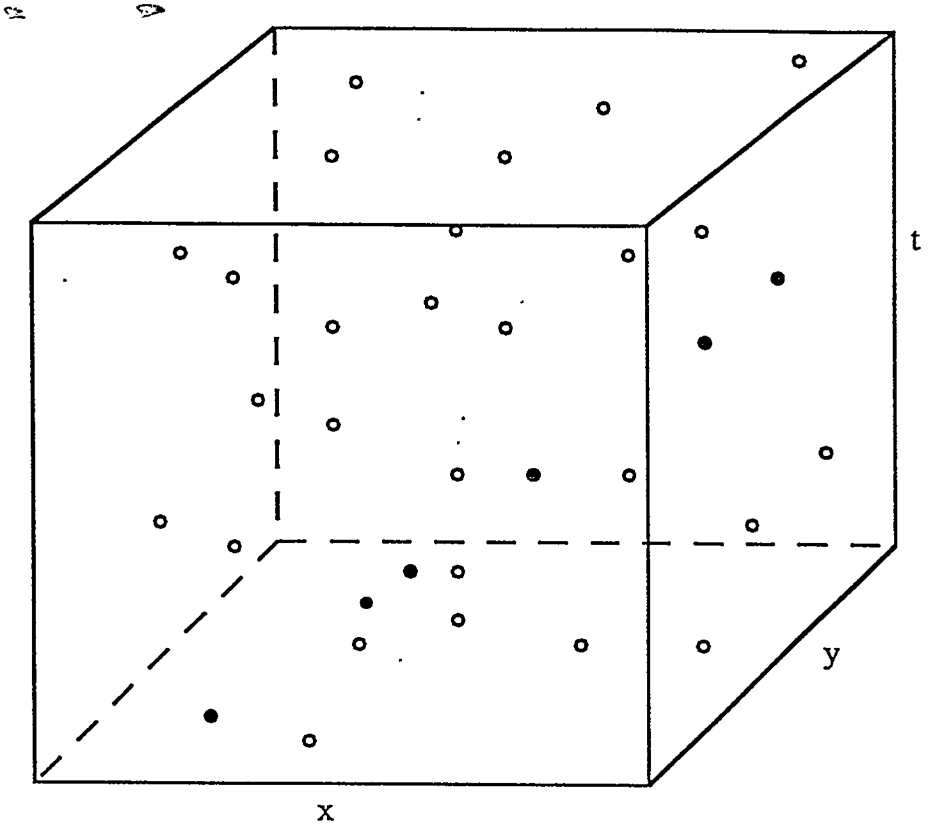

Fig. 1 – Schematics of linear features in a 3D data set. In 3D, the line consisting of filled circles is significant against the background represented by the open circles. The line is not significant in 2D.

For the purpose of discussion, suppose there are B photons randomly distributed in a grid of D pixels in each dimension, then the expected number of lines consisting of n photons due to random chance is roughly

$$\text{Digitized 3D data set :} \qquad K = \frac{D^4}{n!} \times (B/D^2)^n \exp(-B/D^2). \tag{2}$$

For this order-of-magnitude estimate, we have taken $L = D$, $A = 1$, and $N_L = D^4$. Taking $B = 10^6$ and $D = 2048$, we get $K \lesssim 10^{-10}$ for $n = 16$, thus making the detection of a 16-photon line against a background of 10^6 photons highly significant.

In comparison, if the data is in 2D instead of 3D, the expected number of lines would be

$$\text{Digitized 2D data set :} \qquad K = \frac{D^2}{n!} \times (B/D)^n \exp(-B/D). \tag{3}$$

With the same parameters as above, at the same level of significance (10^{-10}), we can only hope to find lines consisting of more than ~ 680 photons. Of these, we expect about 500 photons to come from the background and about 180 from the source.

Summarizing these considerations, we see a great advantage in going into a 3D data format. To accomplish this detection scheme, we need 1) an imaging photon counting system with high count rate and 2) a viable data analysis scheme to search for the line.

2. BASELINE SYSTEM CONFIGURATION

Technology now under development can yield a fast imaging photon counting detector with a 2048 by 2048 format and a 1,000,000 count/s maximum count rate. Such a detector is achievable using microchannel plate intensified crossed delay-line readout.[1,2] A count rate of 10^6/s would be observed by a 16-cm telescope with a 1° circular field of view, assuming a typical quantum efficiency for an unfiltered S-20 detector. This counting rate corresponds to the moonless sky brightness at zenith for a mean galactic latitude, taken to be 22.5 mag/square arcsec. Of the count rate, roughly 25 percent would be from stars brighter than 16th magnitude, and the rest would be air glow, zodiacal light, faint stars, and diffuse galactic light. The roughly 25 percent stellar contribution could be removed in pre-processing, leaving a diffuse background count rate of about 750,000/s to be processed for moving objects. A debris object of visual magnitude ~16 will yield 16 counts/sec, detectable with the algorithm discussed below. This magnitude corresponds to 1.3 cm diameter sphere at 400 km altitude, given a Lambertian scatterer viewed at phase angle 90° with geometric albedo 0.08. An object in a 400 km circular orbit crosses the 1° field of view in about 1 sec.

The detector will detect each photon at its location (x, y). With the detector electronics providing the appropriate time tag t, each photon is represented by a point in a 3D digitized (x, y, t) space. A data set taken in one second consists of 1,000,000 photon records in the 2048 by 2048 by 2048 volume. The detector time resolution is $\sim 1\mu s$ even though we only need a resolution of order 1 ms for this purpose. The reflected sunlight photons from a moving object form a straight line in this volume. In comparison, a stationary object such as a star yield a concentration of photons in certain x, y locations independent of t, and the background photons will be randomly distributed. The data processing task can be reduced to the mathematical problem of finding a statistically significant line at an oblique angle in this volume.

One of the key challenges of this debris detection scheme is the large volume of data and the processing needed to extract the useful information. A crude estimate of the number of possible distinct lines passing through the box is about $(2048)^4 > 10^{13}$. Examination of all possible lines is a formidable task, requiring very long processing time or massive parallelism. In the following section, we discuss the data analysis and the search for line.

3. LINE DETECTION ALGORITHM

3.1 Scope of the problem

The generic problem at hand is to search for possible statistically significant linear features within a data set which is sparsely populated in a high-resolution data space. Following the context of the debris detection and tracking application, we shall refer to the embedded line in the data set as the debris line and the data points forming the line as the debris photons. The key assumption of this problem is that we have no *a priori* knowledge of 1) whether or not there is a line in the data set, and 2) where the debris photons are if there is a debris line.

The scope of the problem can be estimated by considering the following analysis scheme. Characterizing each line in 3D by four parameters [two slopes (velocity of the debris) and two intercepts in a fiducial plane], we can imagine projecting all data points onto the plane whose normal is defined by a specific pair of slopes. If there is a line of significance in the data, then in the 2D projection of the 3D data points with the correct slopes, there will be a significant enhancement at the appropriate location corresponding to the intercepts of the line. This is analogous to applying a Hough transform to the 3D data. In practice, this is the same as having a telescope moving at the same angular velocity as the debris, causing all the debris photons during the debris transit time to fall onto one or a few adjacent pixels in the intercept plane. Since we have no *a priori* knowledge of the velocity and direction of the debris object, there are $\sim D^2$ possible slopes; and since we don't know where the debris is, there are $\sim D^2$ possible intercepts for each projection. A search for the line involves approximately D^2B operations for projection and D^4 operations to search for enhancement in all possible slopes. For $D = 2048$ and $B \sim 10^6$ as described earlier, the number of floating point operations needed for a thorough scan for lines within the data is about 10^{13-14} for a data set collected over 1 second described in Section 1. On a machine with GFLOPS capability ($\gtrsim$ Cray), the analysis of 1 second of data will take up to a day. Such a large ratio, $\sim 10^{4-5}$, between data acquisition rate and analysis rate represents a major impediment to the practicality of this debris detection scheme.

There appear to be several ways to speed up the data analysis. 1) We can restrict the parameter space for the search of the line. For example, we can do a search of line within a narrow range of direction and time. This could apply if we are looking for debris due to a break up event at the pinch point. 2) The generalized Hough transform scheme described above appears to be amenable to massive parallelism. The potential improvement in data processing speed is large. 3) We can develop alternative analysis algorithms. The first approach applies when we do have some *a priori* knowledge of the debris. The second approach requires further examination and detailed implementation. In the rest of this paper, we discuss a method along the third approach: a hierarchical pair and stretch scheme to search for a line embedded in the data set.

3.2 Basic concept of line detection in sparse data

We find debris lines by exploiting the likelihood of close pairs of debris photons. Given a debris line consisting of n photons randomly distributed in a length of L, the probability that none of the debris photons are separated by less than x from its neighboring debris photon is roughly $\sim \exp(-n^2x/L)$. Thus the probability of finding at least one pair of debris photons of spacing x or closer, is $1-\exp(-n^2x/L)$. For example, for $n = 16$ and $L = 2048$, there is an $\sim 86\%$ chance that we will find a pair of debris photons separated by no more than 16 pixels. A small samples of this case is shown in Figure 2. The probability of picking up close debris pairs increases with n and x. On the other hand, the number of pairs formed by the randomly distributed background photons scales like x^3: a small x significantly reduces the number of pairs formed by background photons that we need to examine.

Figure 3 shows the schematics of the hierarchical pair and stretch algorithm. This algorithm include steps of finding a close pair, stretching the pair to form triplets, stretching the triplets to form quadruplets, and finding photons possibly from the line and determining the presence of line and its significance. In the following, we discuss the motivation and concept of individual steps in the line search hierarchy.

Without *a priori* knowledge of whether a close pair is from the debris line or the random background, we need to examine every close pair. This is shown as the first step (PAIR) in the hierarchical diagram shown in Figure 3.

Once we find a close pair, as a candidate of the possible debris line, we need to collect all photons that are co-linear with the pair. Because the pair is closely spaced, they poorly define the candidate line shown as the cone in Figure 3. Since the number of background photons within the cone scales like the third power of the length of the cone, whereas the probability of finding a debris photon scales linearly, the search to find the third point along the line is optimized if we search for photons out to only a finite distance. This second step is shown as the formation of TRIPLETS in Figure 3. The distance that we stretch the cone defined by the pair is optimally chosen to 1) minimize the background photons and 2) maximize the probability of picking up a third photon from the debris line.

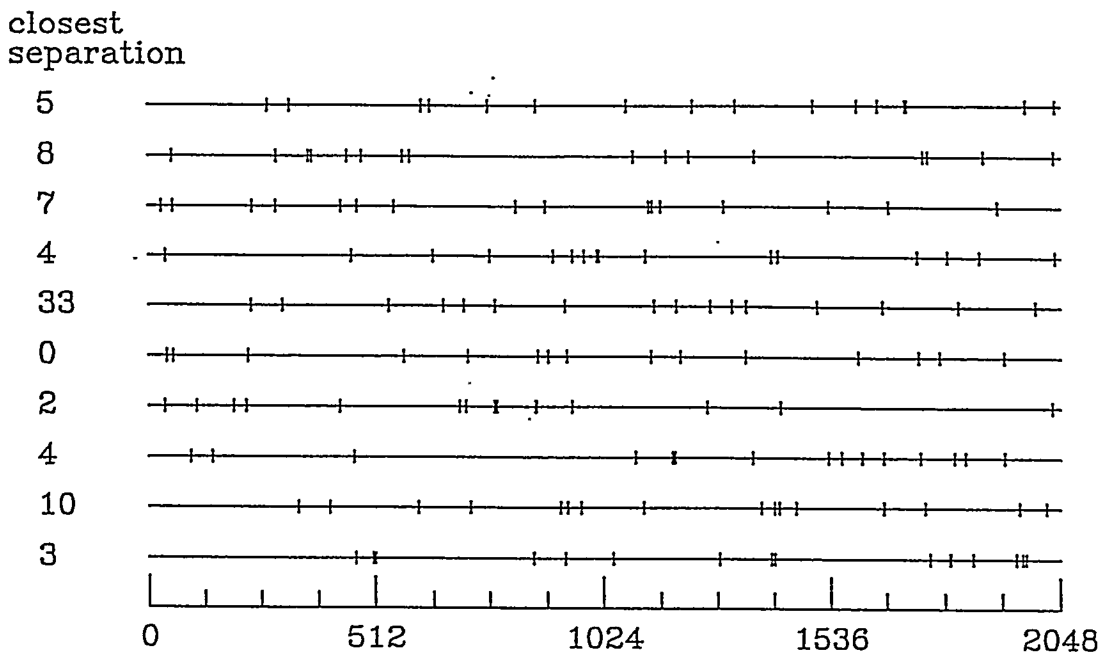

Fig. 2 – Randomly distributed events tend to include close pairs. This figure gives ten examples of 16 randomly distributed points along a line of length 2048. The closest separation between adjacent points is listed on the left. The probability that the closest separation between adjacent points is no more than 16 pixels is 86%. In this small sample of 10, only one shows a closest separation greater than 16.

The pair made up of the two outer photons in a triplet defines the candidate line more precisely than the original close pair. We can use them to further stretch out the line and collect more co-linear photons. This is done in the step shown as QUADRUPLET in Figure 3.

With the increase of the pair forming hierarchy, the line is better defined and the number of candidate lines due to random background is reduced. In principle, this hierarchy can continue indefinitely. With the system configuration described earlier, we find that a quadruplet defines a candidate sufficiently well that, based on the outer pair of photons in a quadruplet, we can scan the entire data set to collect all possible photons from the debris line. This is shown as the LINE TRACING step in Fig. 3.

After we have collected all photons that could be on the line prescribed by the quadruplet, it is straightforward to examine the much reduced data set (typically consisting of 60 photons) for the presence of a line. Algorithms such as least square fit or robust fit can be employed to extract the debris photons and the debris line.

3.3 Implementation

The hierarchical pair and stretch scheme has been implemented in C. We use linked list technique to manage the memory and expedite access to the large amount of sparsely populated data in very fine resolution. The linked list memory management scheme is an integral part of the algorithm. The current implementation of the algorithm allows us to specify a wide range of analysis parameters, such as the range of velocity and the direction of the line, and the length of the triplet stretch cone.

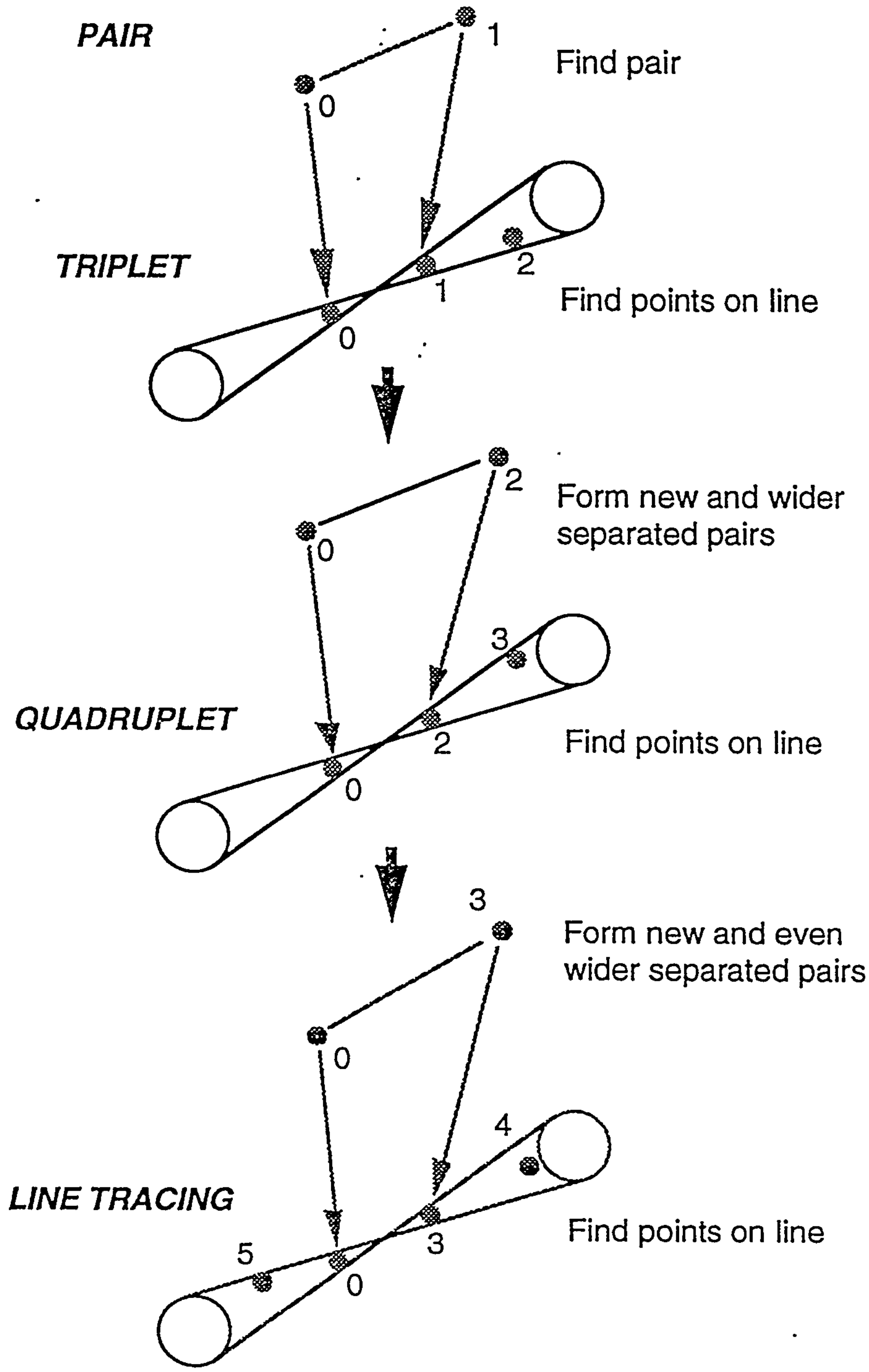

Fig. 3 – Schematics of the hierarchical line searching algorithm.

We have developed the necessary simulation routines to generate data with reasonable fidelity. We typically simulate a one second of observation. To test the performance of line detection, a debris line of prescribed velocity and direction containing any number of debris photons can be incorporated into the simulated data.

The development, simulation, and processing have been done in the Sun SPARC environments. The code itself is not optimized to any specific machine and has been tested on two other platforms. In the next section, we discuss the performance of the processing algorithm evaluated in two critical aspects: the processing time and the probability that an embedded line is picked up. All of the results reported below are for the search of debris lines with angular velocities between 1°/s to 0.5°/s (corresponding to an altitude of about 400 to 800 km) and moving in all directions.

4. ALGORITHM PERFORMANCE AND BACKGROUND

In this section, we discuss the "performance" of the algorithm, i.e. how fast the data are processed and how frequently we can pick up a debris line. We also discuss the distribution of lines caused by the random background, based on which we estimate the significance of any detected line.

4.1 Processing time

We have carried out benchmarks of processing time for various situations and on different platforms. The results reported here is done on SPARC 2 with 32 MBytes of Random Access Memory. The overall processing time is dominated by the number of background photons in the data set. Table 1 gives the processing time for different values of the total photon counts B. A 1 second data set consisting of ~750,000 random background counts will be processed by the SPARC 2 in about 1,000 seconds. Since data acquisition for orbiting object detection will occur for about 2 hours every day during dawn and dusk, a factor of 100 improvement in the processing speed will allow the data processing to proceed at one tenth real time, and thus keep up with the daily data in-take. We have use three sets of analysis parameters, yielding high, medium and low sensitivity for line detection. See discussion below on sensitivity. As expected, the lower the sensitivity the faster the processing time. For the high sensitivity line search, the processing time roughly scales like B^3. For line searches of lower sensitivity, the scaling becomes less steep.

Table 1. Summary of Processing Time (@SPARC2)

Total Counts B	Processing Time (sec)		
(10^6 photons)	high-sens.	med.-sens.	low-sens.
0.25	32	14	—
0.50	230	66	—
0.75	940	220	12
1.00	2,900	680	—

4.2 Pick-Up Probability

The second aspect of the performance is the frequency that a real debris line would be detected by the algorithm. For each prescribed line of n debris photons, we carry out 500 Monte Carlo simulations with 750,000 background photons. We have also changed the analysis parameters such as the length of the triplet stretch cone to yield high, medium and low pick-up probability for faint

Table 2. Summary of Pick-Up Probability

Debris Photons	Debris Size (cm)	Pick-Up Probability high-sens.	med.-sens.	low-sens.
16	1.3	53%	—	—
18	1.4	69%	—	—
20	1.5	82%	45%	—
24	1.6	92%	65%	—
32	1.8	99%	91%	29%
48	2.3	—	—	68%
64	2.6	—	—	96%

lines. We then examine the fraction of times that the embedded debris line is picked up by the processing algorithm. The pick-up probability for different lines is listed in Table 2.

4.3 Background and Statistical Significance

The significance of a debris line is roughly estimated in Eqs. (1)–(3). It is critically dependent upon the length of the line and the operational definition of a line. For example, we have used a cross section of $A = 1$ for Eqs. (2) and (3). In practice, the cross section needs to be enlarged in the analysis to allow for variation caused by digitization. The number of possible background lines is also a sensitive function of length. To better understand the background, we have carried out 150 Monte Carlo simulations with no debris lines and apply the analysis algorithm to the simulated data. Figure 4 shows the distribution of lines formed by random chance and reported by the analysis algorithm, averaged over 150 runs. The x-axis gives the length L of the reported line and the y-axis gives the number of reported lines of length less than L, as a function of L and number of photons on the reported line. Here, the length of the line is defined as the distance between the two outermost photons. Six curves are shown for the reported number of co-linear photons from 11 to 16. Based on these curves one can estimate the significance of a particular line found by the analysis algorithm. For example, there is a roughly 10 percent chance that a line of length 1500 pixels and consisting of 12 photons is due to background, thus making such a line not significant.

5. DISCUSSIONS AND FUTURE DEVELOPMENT

The present simulation shows that small debris objects can be detected with high probability with computing speed $\sim 10^3$ slower than real-time on a SPARC 2. Improvements in algorithms and processing hardware can increase the speed to within the required factor of 10 of real time.

Presently, our simulation generates a data set representing 1 sec of observation. In reality, the data form a continuous stream. Our algorithm can be generalized to handle continuous data. Furthermore, current simulations assume a perfect detector without defects such as gain variation and spatial non-linearity. The effect of seeing is not included, though it should be small at a good site (1 detector pixel $\sim$ 2 arcsec). These are effects that need to be taken into account in future work.

There are many possibilities for improving the algorithm performance measured by the processing time, pick-up probability, and also cost. 1: The algorithm is currently implemented in C. It is conceivable that the algorithm could be accelerated through machine code optimization. 2: The possibility that this algorithm could be adopted to a massively parallel platform such as the Connection Machine should be explored. 3: There are portions of the algorithm that are simple yet heavily used. They could, in principle, be implemented in hardware, such as an Application Specific IC,

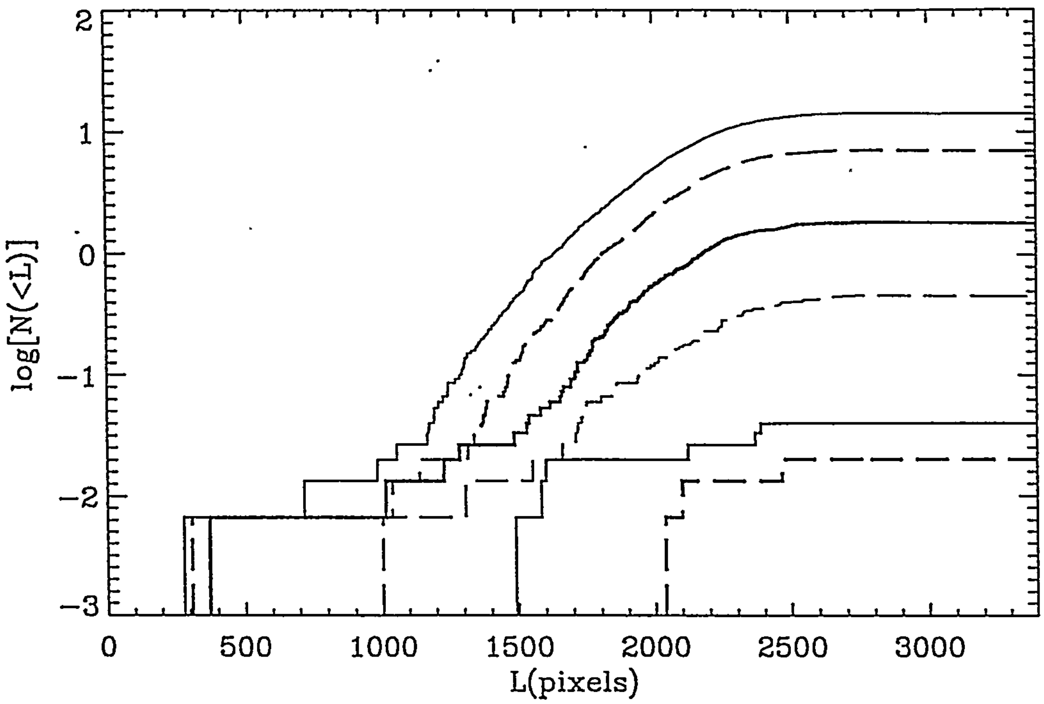

Fig. 4 – Number of lines due to random background as a function of line length L and the number of co-linear points. The y-axis gives the number of lines due to background with length smaller than L. From top to bottom, the curves are for lines with 11, 12, 13, 14, 15, and 16 co-linear photons. The curves are obtained after averaging over 150 independent simulations.

to speed up the processing. 4: Faster processing systems are continually appearing on the market. Given these avenues for improvements, the goal of keeping up on a daily basis (data analyzed/day = data collected/day) or even real-time processing is probably achievable.

We are currently in the process of assembling a brass-board system, including the detector tube, detector electronics, and data acquisition system, to acquire real data. We are planning an end-to-end demonstration of this detection scheme in the near future.

6. ACKONWLEDGEMENTS

We thank Paul Higbie, Steve Kleban, and Khanh Tran for helpful discussions. This work is performed under the auspices of US Department of Energy.

7. REFERENCES

1. Mark B. Williams, Stanley E. Sobottka, and John A. Shepherd, *Nuclear Instruments and Methods in Physics Research*, Vol. A302 105-112, 1991

2. Miles H. Baron and William C. Priedhorsky, 1993, in preparation.