Covariance Analysis of a Charge Carrier Device Processing Algorithm for Stellar Sensors

Burton Boxenhorn*
The Charles Stark Draper Laboratory, Inc., Cambridge, Massachusetts

This paper addresses the problem of modeling a charge carrier device imager array, develops equations for propagating the signal, and develops equations to propagate the noise. Equations are used for a window processor with a comparator for stellar detection. Finally, a numerical example is presented that shows how the probability of detection varies as a function of the signal-to-noise ratio and charge-transfer inefficiency. Some surprising results are explained.

Introduction

THIS paper deals with the problem of stellar detection using a charge carrier device (CCD) as the sensor in a telescope. An analysis is presented that enables one to determine the effect of signal-to-noise ratio (SNR) and charge-transfer efficiency (CTE) on the probability of stellar detection. A special CCD[1] is used for implementation of the processing algorithm. Since the effect of the CTE is to smear out the charge in the pixels, information becomes correlated and it is necesssary to compute a covariance matrix to determine the correlations between the pixel charges.

Several assumptions are implicit in the subsequent sections. These are

1) A perfect low-pass filter at the CCD output removes the bias.

2) The final output statistics are Gaussian—this is believed to be valid because of the central limit theorem. The summing process described will tend to drive the distribution to a Gaussian formulation. The propagation of statistics is independent of the actual distribution.

3) The spatial noise introduced by the pixel-to-pixel gain variation is assumed negligible—this depends upon the acceptable amount of gain variation for a specific application and device quality.

4) All stellar images appear to be the same size on the array—this must be approximately true when designing any practical system in order to solve problems introduced by device saturation and dynamic range.

The CCD is assumed to encompass a relatively small field of view, and it is assumed that the star image is spread over a number of pixels, at least a 3×3 pixel square. The processing scheme essentially scans the CCD with a moving window during each frame. The contents of the window are examined to determine if the star is present.

Problem Definition

A CCD is essentially an array of delay lines. Each delay line is made up of a fixed number of discrete elements called pixels. The contents of these pixels is charge. The charge can be shifted unidirectionally from one adjacent pixel to another along the delay line. The charge can be inserted either electrically or at some specific pixel along the delay line; by using appropriate photoelectric material over the pixel, it can also be generated at each pixel site by light. In addition, the charge is inserted into each pixel at a constant rate by some random artifact of the construction at each pixel site. This rate is the dark current. The charge can be read out by shifting it down the delay line to an output amplifier. To build an imaging CCD, these delay lines are arranged in a rectangular array. To operate a CCD imager, the sequence of events is as follows: 1) expose the image of the photosensitive surface, 2) shift the resulting charge packets to the output register, 3) shift the charge out of the output register, and 4) use the resulting pulses for subsequent processing or transmit them to a television monitor for viewing. Figure 1 is a sketch of the processing system.

A scanning window is formed using two one-line delays to form a 3×3 window. The contents of the window are read and interpreted to provide the detection level signal Q. The stellar image is then detected using a comparator with a threshold. The detection signal Q is computed from the charge in the window and the appropriate weights as

$$Q=\sum_{i=1}^{3}\sum_{j=1}^{3}w_{ij}r_{ij} \tag{1}$$

where w_{ij} is the window element weight and r_{ij} the window element charge. The weights are chosen to match the expected shape of the stellar signal, so that Eq. (1) forms a matched filter.

The charge of the pixel is due to the signal, noise from the dark current, or noise from background radiation. The dark current fluctuates randomly from pixel to pixel and with time, and depends on the device fabrication. The background radiation has a random component that varies with time. The

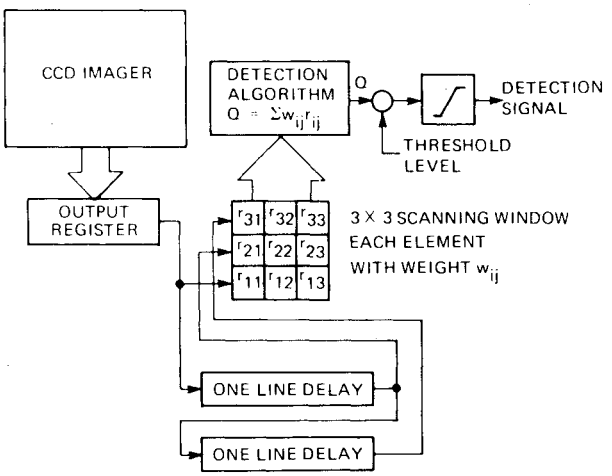

Fig. 1 Schematic of processor.

Presented as Paper 81-1825 at the AIAA Guidance and Control Conference, Albuquerque, N. Mex., Aug. 19-21, 1981; submitted Sept. 30, 1981; revision received Feb. 2, 1983.

*Member, Technical Staff.

question of how the statistics of this random charge change as the charge transfers along the CCD delay line is of primary interest in this study.

The charge-transfer process is not perfect. Some charge is left behind with every transfer. This loss is the prime source of signal degradation. The model for this process is very simple. The direct model is derived next for clarity. If $q_{j,n}$ is the charge in the jth pixel at the nth transfer (time), then the charge at the $n+1$ transfer is

$$q_{j,(n+1)} = aq_{(j-1),n} + (1-a)q_{j,n} \tag{2}$$

where a is the CTE, aq the fraction of charge from the preceding pixel transferred to the jth pixel, and $(1-a)q_{j,n}$ the fraction of charge in the jth pixel that stays behind. Thus, Eq. (2) expresses that the charge in a pixel after a transfer is the sum of the charge left behind and the charge transferred in. This equation is the definition for the CTE a which is, of course, always less than unity and is considered constant in this paper.

Equation Development

The equation development is presented in the following order: 1) signal propagation, 2) covariance matrix, 3) windowing equations, 4) detection signal variance, and 5) computation of the probability of detection.

The following notation conventions are used:

1) Matrices are denoted by upper case letters, such as A.

2) Scalars are lower case unsubscripted letters or Greek lower case letters, such as b, c, β.

3) Vectors are lower case letters. This will not cause confusion with convention 2.

4) In partitioning a matrix, if it is partitioned into vectors, then a lower case letter corresponding to the upper case matrix designation is used with a subscript. *Example*: A is a matrix with $a_1, a_2, \ldots, a_n$ vector elements, i.e., if $A = [a_1, a_2, \ldots, a_n]$, than a_i is a column vector.

5) The elements of a matrix A are denoted by a_{ij}. The usual notation $A = [a_{ij}]$ for a matrix will also be used.

6) If a matrix is partitioned into submatrices, then these submatrices will be denoted by upper case letters with double subscripts. For example,

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$$

where A_{ij} are submatrices. Then the element of A_{ij} would be A_{ijpq} or $(A_{ij})_{pq}$. Note that a_{ij} is still an element of A.

7) The expectation of a variable x is denoted by a superscript bar, i.e., $E(x) = \bar{x}$.

Equation (2) will now be rederived more carefully. Referring to Fig. 2, the jth pixel after the nth shift has a packet of charge in it of q_{jn}. After the $(n+1)$ shift, aq_{jn} charge has been shifted to the $(j+1)$ pixel, leaving behind $(1-a)q_{j,n}$ of charge. A fresh packet of charge from the $(j-1)$ pixel is shifted into the jth pixel that is equal to $aq_{(j-1),n}$. Then, the charge in the jth pixel at shift $(n+1)$ is

$$q_{j,(n+1)} = aq_{(j-1),n} + (1-a)q_{j,n} \tag{3}$$

where a is the CTE. Equations (2) and (3) are, of course, the same.

This recursive equation is direct and can be implemented easily on a digital computer. The problem arises when one wishes to derive the covariance matrix. Assume that the charge in each pixel is a state. Then, there are as many states as there are pixels. In a large array, there may be 100,000 pixels. Then, the covariance matrix would be of dimension $100{,}000 \times 100{,}000$, or 10^{10} elements—an absurd requirement.

The solution to this problem is to shift viewpoints.† Instead of observing what happens to the charge in a particular pixel as the charge shifts down the delay line, observe one charge packet as it shifts down the delay line. Consider a train of charge packets such as that shown schematically in Fig. 3.

Note that the first packet is the leading one traveling down the delay line. There are m packets in the train. Examining the typical jth packet, the transfer in the $(n+1)$ shift in is $(1-a)q_{(j-1),n}$,‡ where a is the CTE. The flow out of the packet is $(1-a)q_{jn}$. The amount present at $(n+1)$ is $q_{j(n+1)}$. Then, applying charge conservation, Eq. (4) is

$$\begin{aligned} q_{j(n+1)} &= q_{jn} - (1-a)q_{jn} + (1-a)q_{(j-1)n} \\ &= aq_{jn} + (1-a)q_{(j-1)n} \end{aligned} \tag{4}$$

Replacing $(n+1)$ by n, Eq. (4) becomes

$$q_{jn} = aq_{j(n-1)} + (1-a)q_{(j-1)(n-1)} \tag{5}$$

Equation (5) is the fundamental recursive equation.

The next section deals with obtaining the state vector q as a function of the initial condition vector q_0 and a transition matrix F. Taking the Z transform of Eq. (5)

$$\phi(z)_j = az^{-1}\phi(z)_j + q_{j,0} + (1-a)z^{-1}\phi(z)_{j-1} \tag{6}$$

or

$$\phi(z)_j = \frac{(1-a)z^{-1}\phi_{j-1}}{1-az^{-1}} + \frac{q_{j,0}}{(1-az^{-1})} \tag{7}$$

Note that $q_{j,0}$ refers to q_0 at the jth packet. A general matrix can be written for the train using Eq. (7) as

$$\begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ \vdots \\ \phi_m \end{bmatrix} = \begin{bmatrix} \dfrac{1}{1-az^{-1}} & 0 & \cdots & \cdots & 0 \\ \dfrac{\alpha z^{-1}}{(1-az^{-1})^2} & \dfrac{1}{(1-az^{-1})} & 0 & \cdots & 0 \\ \dfrac{(\alpha z^{-1})^2}{(1-az^{-1})^3} & \dfrac{\alpha z^{-1}}{(1-az^{-1})^2} & \dfrac{1}{(1-az^{-1})} & 0 \cdots & 0 \\ \hline \dfrac{(\alpha z^{-1})^{m-1}}{(1-az^{-1})^m} & \dfrac{(\alpha z^{-1})^{m-2}}{(1-az^{-1})^{m-1}} & \cdots & \dfrac{\alpha z^{-1}}{(1-az^{-1}} & \dfrac{1}{1-az^{-1}} \end{bmatrix} \begin{bmatrix} q_{1,0} \\ q_{2,0} \\ q_{3,0} \\ \vdots \\ q_{m,0} \end{bmatrix} \tag{8}$$

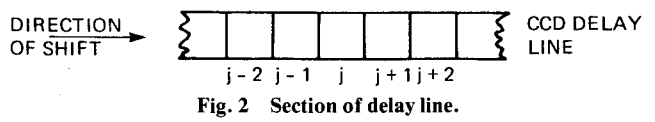

Fig. 2 Section of delay line.

†This was suggested by Brock S. Dew of The Charles Stark Draper Laboratory, Inc.

‡Note that from here on, the subscript notation $x_{n,m}$ may be replaced by x_{nm}, omitting the comma. Any ambiguities should be clarified by the context.

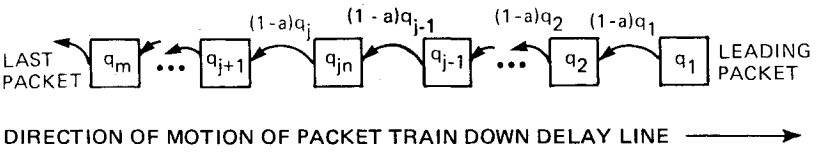

Fig. 3 Train of charge packets.

This neglects input to the leading packet, and

$$\alpha = 1 - a \tag{9}$$

where α is the charge-transfer inefficiency (CTI). Note that m is the last packet in the train.

The matrix Eq. (8) can be inverted to obtain the sampled time function (below is the transform pair from Ref. 2.),

$$\binom{n+k}{k} c^n \leftrightarrow \frac{z^{k+1}}{(z-c)^{k+1}}$$

The result is the matrix

$$F = a^n \begin{bmatrix} 1 & 0 & \cdots & 0 \\ \alpha(n+1)u(n-1) & 1 & \cdots & 0 \\ \alpha^2(n+2)(n+1)u(n-2) & \alpha(n+1)u(n-1) & \cdots & 0 \\ \hline \dfrac{\alpha^{m-1}(n+m-1)(n+m-2)\cdots(n+1)u[n-(m-1)]}{(m-1)!} & & \cdots & 1 \end{bmatrix} \tag{10}$$

where n is the sampled time and $u(\cdot)$ the unit step function.

For practical application in this paper, it is assumed that n is large. This means that the length of the packet train is small compared to the number of shifts, so that F approaches

$$F = a^n \begin{bmatrix} 1 & 0 & 0 & \cdots & \cdots & 0 \\ \lambda & 1 & 0 & \cdots & \cdots & 0 \\ \dfrac{\lambda^2}{2} & \lambda & 1 & 0 & \cdots & 0 \\ \hline \dfrac{\lambda^{m-1}}{(m-1)!} & \dfrac{\lambda^{m-2}}{(m-2)!} & \cdots & \dfrac{\lambda^2}{2!} & \lambda & 1 \end{bmatrix} = a^n F_\infty \tag{11}$$

where $\lambda = \alpha n$.

The dynamical equation is then

$$q = Fq_o \tag{12}$$

where q is a column vector of charge, and q_o a column vector of the original charge.

Before further development, the array geometry must be defined. This is done in Fig. 4. The array is assumed to contain a square patch. The charge on the patch is the matrix (x_{ij}). The charge patch, consisting of n^2 charge packets, propagates up to the output register. Once there, the charge packets become y_{ij}. The charge is then propagated out of the register into the first delay line denoted u, and then to the second delay line denoted v. The window is formed from the output of the y, u, v delay line.

The propagation of the patch to the y register is

$$X_1 = a^{n_x} F_\infty(\lambda_x) X_0 \tag{13}$$

where n_x is the number of shifts of the patch to the top, λ_x the parameter for the patch, and X_0 the original charge matrix.

The assumptions are that 1) F_∞ can be used, 2) there are no other inputs to the packet, and 3) the number of shifts is the same for all packets in the train to the output register y.

Similar equations can be written for the output of the y, u, and v registers

$$Y' = a^{n_y} F_\infty(\lambda_y) X_1^T \tag{14}$$

$$U' = a^{n_u} F_\infty(\lambda_u) X_1^T \tag{15}$$

$$V' = a^{n_v} F_\infty(\lambda_v) X_1^T \tag{16}$$

Some explanations are now in order. The matrix X_0 is considered to be made up of a set of packet trains. The vertical columns are the trains. Equation (13) propagates these trains to the y register. After transfer to the y register, the trains must be reorganized to form the y register trains, which are propagated transverse to the array. To make the rows of

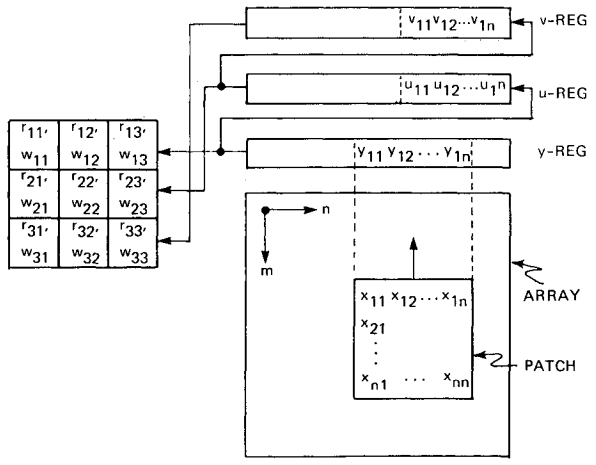

Fig. 4 Geometry of coordinate definitions.

the Y, U, and V matrices correspond to the packet train, form

$$Y=(Y')^T \tag{17}$$

$$U=(U')^T \tag{18}$$

$$V=(V')^T \tag{19}$$

Then finally, combining Eqs. (14-19)

$$Y=a^{n_y}X_1F(\lambda_y)^T \tag{20}$$

$$U=a^{n_u}X_1F(\lambda_u)^T \tag{21}$$

$$V=a^{n_v}X_1F(\lambda_v)^T \tag{22}$$

where the elements are row vectors

$$Y=\begin{bmatrix} y_1 \\ y_2 \\ \cdot \\ \cdot \\ \cdot \\ y_n \end{bmatrix};\ U=\begin{bmatrix} u_1 \\ u_2 \\ \cdot \\ \cdot \\ \cdot \\ u_n \end{bmatrix};\ V=\begin{bmatrix} v_1 \\ v_2 \\ \cdot \\ \cdot \\ \cdot \\ v_n \end{bmatrix} \tag{23}$$

and $y_1=[y_{11}y_{12}\ldots y_{1n}]$, etc. The rows correspond to the packet trains along these registers. Equations (20-22) are the final signal propagation equations.

The covariance matrices for each register output can now be computed. (Without loss of generality, the assumption is made of zero mean for the variables.) The covariance matrix for the y register is

$$C_y=\overline{\begin{bmatrix} y_1^T \\ y_2^T \\ \cdot \\ \cdot \\ \cdot \\ y_m^T \end{bmatrix}[y_1y_2\ldots y_m]} \tag{24}$$

recalling that y_i is a row vector, Eq. (24) becomes

$$C_y=\begin{bmatrix} \overline{y_1^Ty_1} & \overline{y_1^Ty_2} & \ldots & \overline{y_1^Ty_n} \\ \overline{y_2^Ty_1} & \overline{y_2^Ty_2} & \ldots & \overline{y_2^Ty_n} \\ \hline \overline{y_n^Ty_1} & \overline{y_n^Ty_2} & \ldots & \overline{y_n^Ty_n} \end{bmatrix}$$

$$=\begin{bmatrix} \overline{C_{y11}} & \overline{C_{y12}} & \ldots & \overline{C_{y1n}} \\ \overline{C_{y21}} & \overline{C_{y22}} & \ldots & \overline{C_{y2n}} \\ \hline \overline{C_{yn1}} & \overline{C_{yn2}} & \ldots & \overline{C_{ynn}} \end{bmatrix} \tag{25}$$

If the assumption is made that the correlations between elements in different rows are negligible, then only the matrices along the main diagonal in Eq. (25) are of interest. Combining Eq. (13) with Eq. (20)

$$Y=a^{(n_y+n_x)}F_\infty(\lambda_x)X_0F(\lambda_y)^T \tag{26}$$

Eq. (26) can be written as

$$Y=a^{(n_y+m_x)}\begin{bmatrix} f_{x1} \\ f_{x2} \\ \cdot \\ \cdot \\ f_{xn} \end{bmatrix}[x_1x_2\ldots x_m]\,[f_{y1}^Tf_{y2}^T\ldots f_{yn}^T] \tag{27}$$

where $f(\cdot)$ are the row vectors of the F_∞ matrix, and $x(\cdot)$ the column vectors of the initial condition matrix X_0. Multiplying Eq. (27) and extracting one term y_1 as an example results in

$$y_1=a^{n_y+n_x}[f_{x1}x_1f_{x1}x_2\ldots f_{x1}x_n]\,[f_{y1}^Tf_{y2}^T\ldots f_{yn}^T] \tag{28}$$

and then

$$C_{y11}=\overline{y_1^Ty_1}=a^{2(n_y+n_x)}F_y\,[\overline{X_0^Tf_{x1}^Tf_{x1}X_0}]F_y^T \tag{29}$$

In general, the expression for terms along the main diagonal is

$$C_{yii}=a^{2(n_y+n_x)}F_y\,[\overline{X_0^Tf_{xi}^Tf_{xi}X_0}]F_y^T \tag{30}$$

Note that the notation $F_y=F_\infty(y)$ and that of $(\,)_0$ on x have been dropped for simplicity.

Continuing the development, another way of writing an expression for C_{y11} is

$$C_{y11}=a^{2(n_y+n_x)}$$

$$\times F_y\begin{bmatrix} \overline{(f_{x1}x_1)^2} & \overline{f_{x1}x_1f_{x1}x_2} & \ldots & \overline{f_{x1}x_1f_{x1}x_m} \\ \overline{f_{x1}x_2f_{x1}x_1} & \overline{(f_{x1}x_2)^2} & \ldots & \overline{f_{x1}x_2f_{x1}x_m} \\ \overline{f_{x1}x_nf_{x1}x_1} & \overline{f_{x1}x_nf_{x1}x_2} & \ldots & \overline{(f_{x1}x_m)^2} \end{bmatrix}F_y^T \tag{31}$$

Rewriting Eq. (30), the general expression for the covariance of interest is

$$C_{yii}=a^{2(n_y+n_x)}F_y\,[\overline{X_0^Tf_{xi}f_{xi}X_0}]F_y^T \tag{32}$$

$$C_{yii}=a^{2(n_y+n_x)}F_yT_{ii}F_y^T \tag{33}$$

where

$$T_{ii}=[t_{iipq}]=\sum_{s=1}^{m}\sum_{r=1}^{m}f_{xir}f_{xis}\overline{x_{rp}x_{sq}} \tag{34}$$

The complete solution for the covariance has been derived. Great simplification is possible if it is assumed that X_0 is uncorrelated and that the variances of all the terms are the same, equal to σ^2. Then $\sigma_{rp}^2=\overline{x_{rp}^2}=\sigma^2$ and Eq. (34) can be reduced to

$$T_{ii}=[t]=\sigma^2\sum_{r=1}^{i}f_{xir}^2 \tag{35}$$

Then Eq. (32) becomes

$$C_{yii}=a^{2(n_y+n_x)}\sigma^2\left[\sum_{r=1}^{i}f_{xir}\right]F_yF_y^T \tag{36}$$

From the expression F_x given in Eq. (11), Eq. (36) becomes

$$C_{yii}=a^{2(n_y+n_x)}\sigma^2\left[1+\lambda_x^2+\left(\frac{\lambda_y^2}{2}\right)+\cdots\left(\frac{\lambda_x^{i-1}}{(i-1)!}\right)^2\right]F_yF_y^T \tag{37}$$

Equation (37) is the result that will be used for numerical computations.

For the output of the other two delay lines, the answer can be written immediately from observation of Eq. (37)

$$C_{uii}=a^{2(n_u+n_x)}\sigma^2\left[1+\lambda_x^2+\left(\frac{\lambda_x^2}{2}\right)^2+\cdots\left(\frac{\lambda_x^{i-1}}{(i-1)!}\right)^2\right]F_uF_u^T \tag{38}$$

$$C_{vii}=a^{2(n_v+n_x)}\sigma^2\left[1+\lambda_x^2+\left(\frac{\lambda_x^2}{2}\right)^2+\cdots\left(\frac{\lambda_x^{i-1}}{(i-1)!}\right)^2\right]F_vF_v^T \tag{39}$$

The development of the covariance matrix is complete. The next task is to develop equations which describe the statistics of the detection signal. The previous results from the covariance matrix will be used to obtain the necessary statistics for computing the detection signal-to-noise ratio. Assuming a 3×3 pixel window, assign weights to each pixel in the window. Then a weighting matrix can be defined as

$$W=\begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \\ w_{31} & w_{32} & w_{33} \end{bmatrix}=\begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} \tag{40}$$

The charge in each pixel of the window is denoted by m_{ij}, and a measurement matrix may be defined as

$$M=\begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix}=\begin{bmatrix} m_1 \\ m_2 \\ m_3 \end{bmatrix} \tag{41}$$

The detection signal is

$$q=\sum_{j=1}^{3}\sum_{i=1}^{3}w_{ij}m_{ij}=w_1m_1^T+w_2m_2^T+w_3m_3^T \tag{42}$$

The object is to find the variance of q and σ_q^2, given σ_m^2.

The measurement is signal plus noise

$$m_{ij}=s_{ij}+n_{ij} \tag{43}$$

where s_{ij} is the signal and n_{ij} the correlated noise with zero mean. Using Eqs. (42) and (43) and using vector notation, the mean of q is

$$\bar{q}=w_1s_1^T+w_2s_2^T+w_3s_3^T \tag{44}$$

and the variance is

$$\sigma_q^2=\overline{[\bar{q}-q]^2} \tag{45}$$

which can be written as [from Eqs. (42-44)]

$$\sigma_q^2=\overline{(w_1n_1^T+w_2n_2^T+w_3n_3^T)^2} \tag{46}$$

By examining Eq. (46), one can see that a set of covariance matrices are generated in the form $\overline{n_i^Tn_j}$. The terms where $i=j$ are those that correlate elements in a window row with each other. The terms where $i\neq j$ correlate elements in one row to elements in another row. There are 9 covariance matrices and 9 terms in each for a total of 81 terms. However, in this paper it has been assumed generally that the correlations between rows are zero. Applying this rule to Eq. (46) the variance of q becomes

$$\sigma_q^2=w_1\overline{n_1^Tn_1}w_1^T+w_2\overline{n_2^Tn_2}w_2^T+w_3\overline{n_3^Tn_3}w_3^T$$

$$\overline{n_i^Tn_j}=0,\ i\neq j \tag{47}$$

The noise covariance elements are found from computation of the delay line covariance derived previously.

The parameter of specific interest for threshold detection is the signal-to-noise ratio SNR_q defined as

$$\mathrm{SNR}_q=\bar{q}/\sigma_q \tag{48}$$

referred to as the array signal-to-noise ratio. Using Eqs. (44) and (46) (uncorrelated rows), Eq. (48) is written

$$\mathrm{SNR}_q=\frac{w_1s_1^T+w_2s_2^T+w_3s_3^T}{(w_1\overline{n_1^Tn_1}w_1^T+w_2\overline{n_2^Tn_2}w_2^T+w_3\overline{n_3^Tn_3}w_3^T)^{1/2}} \tag{49}$$

To make these equations amenable to numerical analysis, further normalizing must be done. Then, let the total signal in the window be e and define

$$e=\sum_{j=1}^{3}\sum_{i=1}^{3}s_{ij} \tag{50}$$

$$\psi_{ij}=s_{ij}/e \tag{51}$$

then

$$\sum_{j=1}^{3}\sum_{i=1}^{3}\psi_{ij}=1 \tag{52}$$

and further, define

$$\omega_{ij}=w_{ij}/e \tag{53}$$

$$\eta_{ij}=n_{ij}/e \tag{54}$$

For a matched filter, the weights are proportional or equal to the signal, that is

$$w_{ij}=s_{ij}(\text{matched filter}) \tag{55}$$

and therefore

$$e=\sum_{j=1}^{3}\sum_{i=1}^{3}w_{ij} \tag{56}$$

$$\sum_{j=1}^{3}\sum_{i=1}^{3}\omega_{ij}=1 \tag{57}$$

For the general unmatched case where all elements are uncorrelated, the covariance matrices of Eq. (49) become simply the variance of the pixel noise. Then, the SNR_q becomes

$$\mathrm{SNR}_q=\Sigma w_{ij}s_{ij}/\sigma_m(\Sigma w_{ij}^2)^{1/2} \tag{58}$$

for notational convenience, the double summation ($\Sigma\Sigma$) has been written as a single summation (Σ) when the meaning is obvious. Equation (57) is normalized to

$$\mathrm{SNR}_q=(e/\sigma_m)\Sigma\omega_{ij}\psi_{ij}/(\Sigma\omega_{ij}^2)^{1/2} \tag{59}$$

There are several points to note in Eq. (59). First, the quantity e/σ_m is a signal-to-noise ratio approximately equal to the stellar-energy/pixel-noise ratio. Here it is called the fundamental signal-to-noise ratio (SNR_F). If e is considered to contain all the signal (stellar) energy, and σ_m is the standard deviation of the noise for one pixel, then this quantity is the greatest signal-to-noise ratio the system can have. It is equivalent to the case where all the signal energy is concentrated in one pixel. The probability of detection that corresponds to this value represents the best case. Due to the windowing and the spatial distribution of the charge over the window, the value of e/σ_m is reduced by the quantity k, such

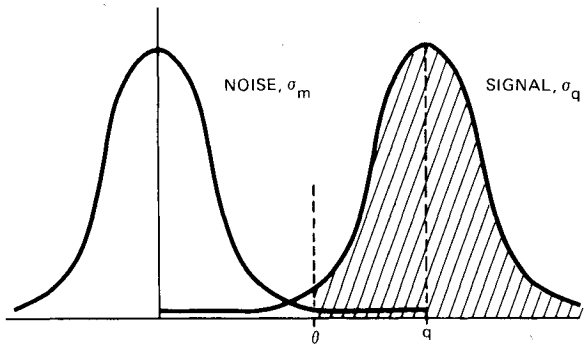

Fig. 5 Threshold definition.

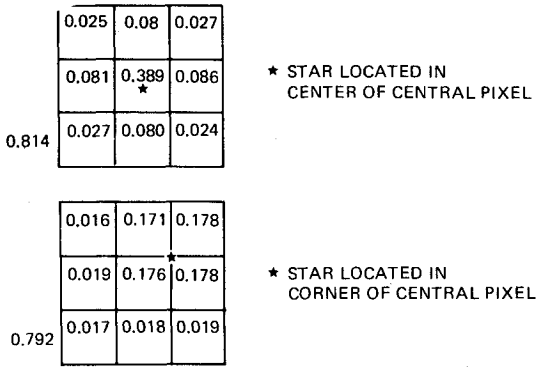

Fig. 6 Normalized star and integrated energy on a 3×3 window for star at center and star at corner.

that

$$\mathrm{SNR}_q = k(e/\sigma_m) = k\mathrm{SNR}_F$$

where

$$k = \Sigma\omega_{ij}\psi_{ij}/(\Sigma\omega_{ij}^2)^{1/2} \tag{60}$$

Given the weights and density of the (stellar) signal over the array, the signal can be integrated over the pixels to finally obtain a normalized signal on each pixel as a function of the position of the star within the central pixel. Thus, k depends on the signal shape and its position within the central pixel of the window. For instance, when the signal is matched to the weights, $\omega_{ij} = \psi_{ij}$, Eq. (60) becomes [using Eqs. (60) and (57)]

$$(k)_{\text{matched}} = (\Sigma\psi_{ij}^2)^{1/2}$$

or if ω_{ij} is a constant, then $k = 1/3$. Thus if the weights are all equal, the array signal-to-noise ratio is reduced by a factor of one-third. Also, if the signal shape is rectangular and flat, then k is invariant with signal position within the window (as long as the signal is larger than one pixel).

The next and last item to be discussed is the computation of the probability of detection, $P(D)$. Calculation of this number uses the well-known threshold detection theory.[3] This theory assumes a Gaussian distribution for the noise. Figure 5 identifies the use of the parameters.

The probability distribution is that for the noise with a mean of q. θ is the comparator threshold. The area to the right of θ, shown hatched in Fig. 5, is the probability of detection.

Computation

A FORTRAN program was written to compute the probability of detection. The equations used were Eqs. (13), (20-22), (37-39), and (49). The stellar signal was separately computed using a ray-trace program to obtain the stellar density. Choosing an array size, the integrated signal on the pixels was computed and the result tabulated in normalized form. The normalized stellar signal used for the results reported here is tabulated in Fig. 6.

Note that the sum of the weights in Fig. 6 does not equal unity. This is because all the stellar energy does not lie in the 3×3 pixel window in this example. Table 1 shows the parameters for this example.

The results are shown in Figs. 7 and 8 which plot the probability of not detecting $P(\bar{D})$ vs the fundamental signal-to-noise ratio (SNR_F) and the CTI (α).

Examining Fig. 7, which is $P(\bar{D})$ vs SNR_F with $\alpha = (1-a)$ as a parameter, the most noteworthy characteristic is the enormous sensitivity of $P(\bar{D})$ to the signal-to-noise ratio. For example, $P(\bar{D})$ drops from 0.76 at $\mathrm{SNR}_A = 8$ to 10^{-3} at $\mathrm{SNR}_A = 16$ for $\alpha = 0.0005$. In other words, doubling the stellar intensity results in the probability of detection going from unlikely to very likely.

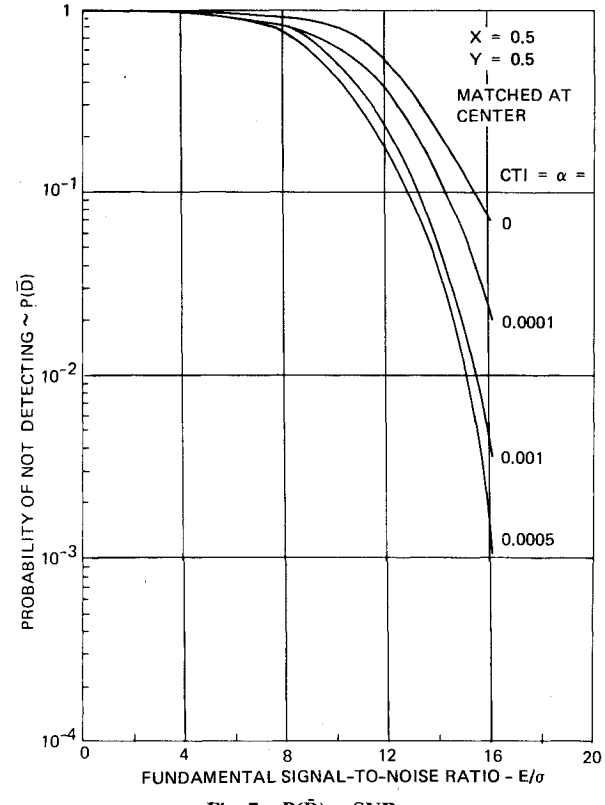

Fig. 7 $P(\bar{D})$ vs SNR_F.

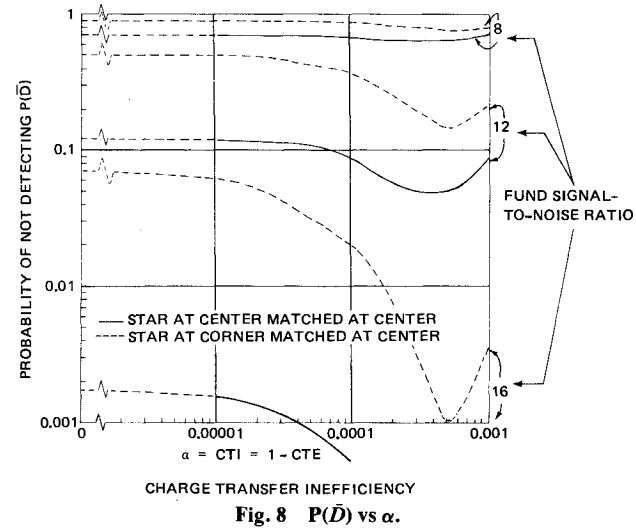

Fig. 8 $\mathbf{P(\bar{D})}$ vs α.

Table 1 Constants for example

Threshold-to-noise ratio $\theta/\sigma = 4.0$
Patch size = 7×7 pixels
Patch location = array center
Array size = 324×324 pixels
CTE, α = range
Fundamental signal-to-noise ratio, SNR_F = range
Weights are stellar signal at center

In Fig. 8, which plots $P(\bar{D})$ vs α, with SNR_F a parameter, a most unexpected characteristic is observed. One's intuition would expect a degradation of performance as the CTE decreases (or CTI increases); instead, performance actually increases over a range of α and then decreases. Thus, some charge-transfer inefficiency is desirable from the point of view of acquisition to provide the minimum $P(\bar{D})$. The explanation for this phenomenon is as follows. The effect of the CTE is to smooth the charge in the delay line. This means that the delay line acts as a filter. The spatial frequency characteristic of the signal is that of a band-pass filter. The noise is white. The CCD delay line acts as a low-pass filter. The result is that as α increases, the pass band of the filter decreases, which cuts off more noise than signal. However, if α gets too large, it begins to cut into the signal as well. Thus, examination of Fig. 8 reveals that as α increases, at first SNR_q increases, resulting in a lower $P(\bar{D})$; but eventually, when α gets large enough, SNR_q starts to decrease resulting in an increase in $P(\bar{D})$.

Figure 8 also shows the effect of where the star is located in the central pixel. The filter weights are chosen for the star located in the pixel center, since it is the only unbiased location. When the star is located in the corner, the weights are incorrect and this results in reduced performance; it is still, however, better than using equal pixel weights for the filter.

Acknowledgment

This article was prepared by The Charles Stark Draper Laboratory, Inc., under Contract N00030-79-C-0096 with the Strategic Systems Project Office of the U.S. Navy. Publication of this paper does not constitute approval by the U.S. Navy of the findings or conclusions contained herein. It is published for the exchange and stimulation of ideas.

References

[1] Hall, J.E., Breitzmann, J.F., Blouke, M.M., and Carlo, J.J., "A Multiple Output CCD Imager for Image Processing Applications," *IEDM Technical Digest,* Dec. 1978.

[2] Jury, E.I., *Theory and Application of the Z-transform Method,* John Wiley & Sons, New York, 1964, p. 280.

[3] "Detection, Resolution and Recognition," *RCA-Electro Optics Handbook,* Sec. 8, RCA Commercial Engineering, Harrison, N.J.