⑫

**Naval Ocean Research and Development Activity**
June 1985 Report 103

# Ocean Wave Slope Statistics from Automated Analysis of Sun Glitter Photographs

Matthew Lybanon
Ocean Sensing and Prediction Division
Ocean Science Directorate

# Foreword

The relationship between sun glitter in the sea surface and the statistical distribution of sea surface slopes allows oceanographers to gather information on the physical conditions of the sea surface by analyzing the glitter. Aerial photographs have been a convenient means for recording the glitter pattern, but analysis of the photographs is often labor-intensive.

Current digital image processing systems and techniques have reduced the labor factor considerably and have increased practicality of sun glitter analysis. This report describes the approach that was used by the Naval Ocean Research and Development Activity, Remote Sensing Branch, to perform sun glitter analysis.

**R. P. Onorati, Captain, USN**
**Commanding Officer, NORDA**

# Executive summary

The image of the sun reflected from the rough surface of the sea forms a diffuse pattern, whose details depend on the nature of the surface waves and swell. Consequently, statistics of the slope distribution of the sea surface are related to the statistics of the sun's glitter on the sea surface. Aerial photographs are a convenient medium for recording the glitter pattern. The mathematical relationship to derive the sea surface slope statistics can be determined from an analysis of the imaging geometry. However, analysis of the photographs can be a labor-intensive procedure.

The problem was first studied over thirty years ago. Now, there are modern digital image processing systems and techniques that greatly increase the practicality of the analysis. This report derives the relevant equations and describes an implementation on the Interactive Digital Satellite Image Processing System (IDSIPS). The IDSIPS system is operated by the Remote Sensing Branch of the Naval Ocean Research and Development Activity (NORDA). The report includes full formal documentation of the computer software.

NORDA's Remote Sensing Branch uses image processing methods to obtain quantitative information on oceanographic parameters from analyses of remotely sensed data. This work is directed toward satisfaction of U.S. Navy requirements for accurate oceanographic information.

# Acknowledgments

The work described in this report was performed with the guidance and consultation of Dr. Peter Smith of NORDA's Remote Sensing Branch. The cover photograph was provided by Paul La Violette. The work was sponsored by OP-006 (formerly OP-952) under Program Element 63704N.

# Contents

# Ocean wave slope statistics from automated analysis of sun glitter photographs

## 1. Introduction

The statistical distribution of sea-surface slopes is related to the statistics of the sun's glitter on the sea surface, so information on the physical conditions at the sea surface can be obtained by analyzing the glitter. Aerial photographs are a convenient medium for recording the glitter pattern. The problem was studied several years ago by Cox and Munk [1, 2]; their analysis is included in Kinsman's book [3].

Modern digital image processing equipment and techniques afford a method of performing the analysis in a way that is much less labor-intensive. Another application of digital image processing to photographic photometry is reported in Reference 4. This report describes the approach that was used by the Naval Ocean Research and Development Activity (NORDA) Remote Sensing Branch to perform sun glitter analysis. A detailed development of the formulation is followed by formal documentation of the computer program.

## 2. Derivations of equations

Geometric considerations relate each point in the glitter image to the particular sea-surface tilt and orientation that are required to reflect the sun's rays toward that image point. The brightness of a picture element (pixel) is related to the fraction of the sea-surface area in the field of view that satisfies the geometric conditions for imaging to that pixel. For a given position of the sun, the pixel value (brightness) distribution is determined by the sea conditions. The Cox and Munk analysis infers information about sea conditions from analysis of the pixel value distribution [1-3].

### 2.1 Geometry

Consider a right-hand Cartesian coordinate system $X_1$-$X_2$-$X_3$ with origin at the sea surface, $X_3$ vertically upward, and $X_2$ horizontal and toward the sun (Refs. 2 and 3 and Fig. 1 illustrate this geometry). An incident ray from the sun is reflected from the origin to a point on a horizontal photographic plate above the sea surface. (The case of a photographic plate that is not horizontal is considered in Section 2.9.) An "image" coordinate system $x$-$y$ in the plane of the photographic plate has the $y$-axis pointing away from the sun. The origin of the image system is the intersection of a vertical through the center of the camera's aperture with the photographic plate.

Define the following angles:

$\alpha$ = direction of steepest ascent of the (element of) sea surface, measured clockwise from the sun direction $X_2$.

$\beta$ = tilt of sea surface.

$\phi$ = sun's elevation (or altitude) above the horizontal.

$\mu$ = angle between vertical and vector from center of camera's aperture to image point.

$\nu$ = angle in image plane between $y$-axis and vector from $x$-$y$ origin to image point, positive clockwise from y-axis.

$\omega$ = angle of incidence or reflection.

By applying the geometrical optics law of reflection [5] to the reflection by the sea surface of a ray from the sun to the photographic plate, it is found that [2, 3]

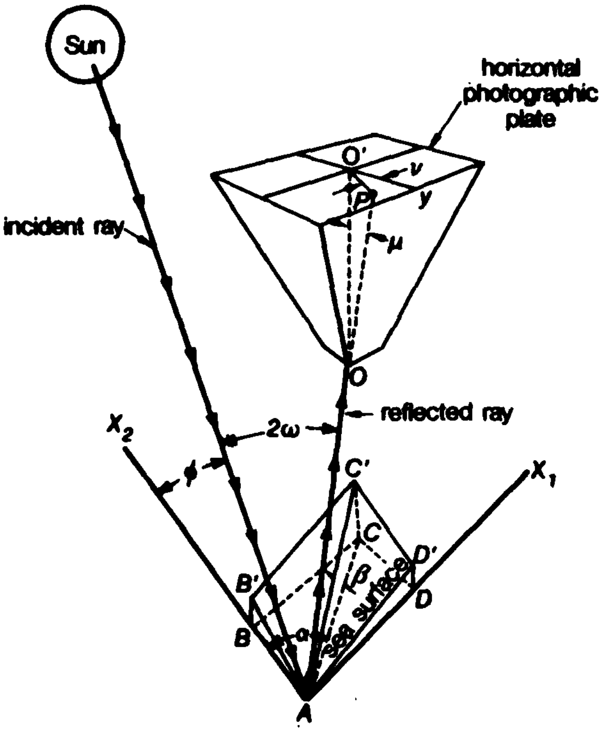

*Figure 1. The geometry of a glitter photograph.*

$$\cos\omega = \cos\beta \sin\phi - \cos\alpha \sin\beta \cos\phi, \quad (1)$$

$$\cos\mu = 2\cos\beta \cos\omega - \sin\phi, \quad (2)$$

$$\cot\nu = \cot\alpha + \tfrac{1}{2}\csc\alpha \csc\beta \sec\omega \cos\phi. \quad (3)$$

The angles $\mu$ and $\nu$ can easily be calculated for any image point. The solar elevation $\phi$ can be calculated from external conditions by standard techniques. It will be shown in Section 2.4 that the angle of incidence $\omega$ can be determined without reference to $\alpha$ and $\beta$. So the mathematical problem consists of finding $\alpha$ and $\beta$ for each image point from the other conditions, and constructing the histograms that approximate the desired probability distributions. It will be shown in Section 2.7 and Appendix A that References 2 and 3 have a sign error in the equation corresponding to Equation (3).

It is convenient to define another coordinate system in the image plane. The system $x_1$-$y_1$ has the same origin as $x$-$y$, but the $x_1$-axis points toward the starboard wing of the aircraft and the $y_1$-axis points toward the nose. Consequently, the $y_1$-axis points toward the top of the image, with the camera mounting geometry that was used. Figure 2 illustrates the two coordinate systems. It is seen that the $x$-$y$ system is oriented at an angle $(Az + Hdg)$ counterclockwise with respect to the $x_1$-$y_1$ system, where

$Az$ = solar azimuth, measured counterclockwise from S,
$Hdg$ = aircraft heading, measured clockwise from N.

Figure 2 also illustrates an image point $P$ and its associated angle $\nu$.

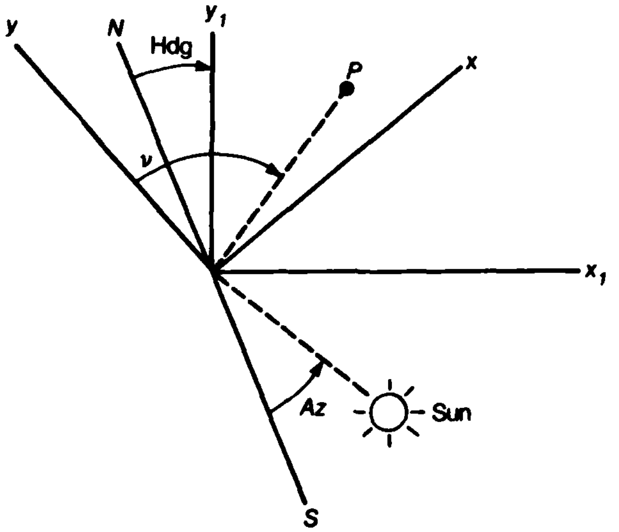

*Figure 2. Relationship between x-y and $x_1$-$y_1$ image plane coordinate systems.*

## 2.2 Solar elevation and azimuth

The first step in the analysis is to locate the sun in the imaging geometry. Let

$t$ = the time of day (GMT) past midnight in minutes,
$\lambda$ = the longitude west of Greenwich in degrees.

The local hour angle $h$ of the sun (the angular distance of that body west of the local celestial meridian) [6] is, in degrees,

$$h = [(t - 720)/4] - \lambda. \quad (4)$$

The sun's elevation and azimuth can be found in terms of the hour angle by solution of the navigational triangle (see Fig. 3). The navigational triangle is a spherical triangle defined by three points on the surface of the earth and formed by the arcs of the great circles that connect these points [6]. The three points are the position of the observer, $M$, the geographical position of the celestial body (the sun in this case) being observed, $GP$, and the earth's pole nearer the observer, $P$. Figure 3 shows the navigational triangle with the sides and one of the vertex angles labeled. In Figure 3

$\delta$ = declination angle of the sun,
$L$ = latitude of the observer.

Then the result of applying one of the standard formulas for the solution of spherical triangles gives [7]

$$\sin\phi = \sin\delta \sin L + \cos\delta \cos L \cos h. \quad (5)$$

The angle at $M$ is $Az - 180°$, as azimuth was defined in Section 2.1. So the application of another spherical trigonometry formula gives [7]

$$\sin Az = -\cos\delta \sin h/\cos\phi. \quad (6)$$

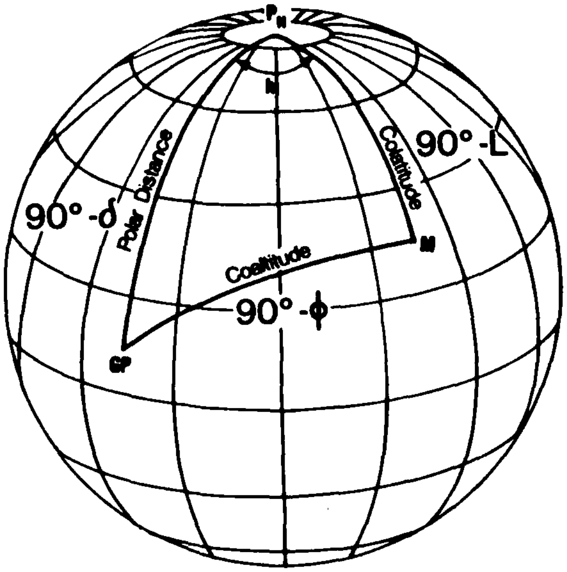

*Figure 3. The navigational triangle.*

## 2.3 Image point position

Let $f$ = the camera lens focal length = distance from lens to image plane, since the camera is assumed to be focused at infinity. Then it is easy to see that

$$\tan\mu = (x_1^2 + y_1^2)^{1/2}/f, \tag{7}$$

where $(x_1, y_1)$ are the convenient set of image plane coordinates. However, $\nu$ is referenced to the $x$-$y$ axis system, so it is necessary to perform the transformation between the two systems. Reference to Figure 2 shows that

$$\nu = Az + Hdg + 90° - \tan^{-1}(y_1/x_1). \tag{8}$$

## 2.4 Angle of incidence or reflection

Consider a vector $\vec{A}$ that points from the reflecting surface toward the sun (opposite to the direction of the incident ray) and another vector $\vec{B}$ that points along the reflected ray (from the sea surface to the image). Then the angle between $\vec{A}$ and the normal to the surface is the incidence angle, and the angle between $\vec{B}$ and the surface normal is the reflection angle. From the geometrical optics law of reflection $\vec{A}$, $\vec{B}$, and the surface normal all lie in the same plane, and the angle of incidence equals the angle of reflection [5]. Let this angle be called $\omega$, as defined in Section 2.1. Then

$$\vec{A} \bullet \vec{B} = |\vec{A}|\,|\vec{B}| \cos 2\omega, \tag{9}$$

since $2\omega$ is the angle between $\vec{A}$ and $\vec{B}$. If the components of $\vec{A}$ and $\vec{B}$ are written in the $X_1$-$X_2$-$X_3$ (sea surface) coordinate system,

$$\begin{array}{ll} A_1 = 0 & B_1 = -|\vec{B}| \sin\mu \sin\theta \\ A_2 = |\vec{A}| \cos\phi & B_2 = |\vec{B}| \sin\mu \cos\theta \\ A_3 = |\vec{A}| \sin\phi & B_3 = |\vec{B}| \cos\mu \end{array} \tag{10}$$

where $\theta$ is the angle between the sun and the image point $P$, measured counterclockwise from the sun. Since the image coordinates $x$ and $y$ are reflections of $X_1$ and $X_2$, respectively, it is easily seen that

$$\theta = 180° - \nu \tag{11}$$

Then the result of performing the operations implied by Equation (9) is

$$\begin{aligned} \cos 2\omega &= \cos\phi \sin\mu \cos\theta + \sin\phi \cos\mu \\ &= -\cos\phi \sin\mu \cos\nu + \sin\phi \cos\mu. \end{aligned} \tag{12}$$

Application of the trigonometric identity

$$\cos^2\omega = \tfrac{1}{2}(\cos 2\omega + 1) \tag{13}$$

allows Equation (12) to be put in terms of a function of $\omega$, instead of $2\omega$. It should be noted that only $\cos\omega$ is needed for the subsequent analysis, so it is not necessary to invert the above equation to find $\omega$ explicitly.

## 2.5 Sea-surface tilt

The sea-surface tilt angle $\beta$ can be found from Equation (2), which can be rewritten

$$\cos\beta = (\cos\mu + \sin\phi)/2 \cos\omega. \tag{14}$$

## 2.6 Direction of steepest ascent of sea surface

From the foregoing we have two equations for the local azimuth of ascent of sea surface $\alpha$, Equations (1) and (3). Appendix A shows how Equation (3) may be solved. Let

$$A = 1/\tan\nu, \tag{15a}$$

$$B = \cos\phi/2 \sin\beta \cos\omega. \tag{15b}$$

Then

$$\alpha = \tan^{-1}\left[\frac{AB \pm (A^2 + 1 - B^2)^{1/2}}{-B \pm A(A^2 + 1 - B^2)^{1/2}}\right], \tag{16}$$

where the ± signs are paired, + with + or − with −. There are two solutions for $\alpha$, both of which can be proved to satisfy Equation (3). Equation (1) gives a method of solving for $\cos\alpha$; again there are two solutions for $\alpha$, since $\cos(-\alpha) = \cos\alpha$. The two equations, (1) and (3), resolve the ambiguity. One of the solutions of Equation (3), given by Equation (16), matches one of the solutions of Equation (1); that one is the correct solution.

It should be noted that Cox and Munk [2], as well as Kinsman [3], show Equation (3) with a minus sign affixed to the last term on the right side. This is not borne out by the derivation. Furthermore, if the equation as given in those references is solved, giving a pair of solutions similar to those given by Equation (16), it is found that neither of those solutions satisfies Equation (1). So, clearly, there is a sign error in those references.

Equation (1) relates $\alpha$ to $\beta$, $\omega$, and $\phi$. Equation (3) also relates $\alpha$ to $\beta$, $\omega$, and $\phi$, and to $\nu$ as well. This may be looked at in two ways:

- Equation (3) has two solutions for $\alpha$ in general, only one of which is consistent with the physics of the problem. For some choices of $\beta$, $\omega$, $\phi$, and $\nu$, Appendix A shows there are no solutions for $\alpha$. But in this analysis those parameters are not all free. The angles $\beta$, $\omega$, and $\phi$, along with an $\alpha$ that satisfies Equation (3), are constrained by Equation (1).

- For a set of values $\alpha$, $\beta$, $\omega$, and $\phi$ that satisfy Equation (1), Equation (3) fixes the value of $\nu$ (within 180°). But $\nu$ is given by Equation (8).

This last observation makes it clear that all of the angles are functionally related in a complicated way. Equations (1), (3), (12), and (14) (or (2)) relate $\alpha$, $\beta$, $\mu$, $\nu$, $\omega$, and $\phi$. But $\phi$, $\mu$, and $\nu$ are given in terms of external conditions by Equations (5), (7), and (8), respectively. So this leaves three variables related by four equations, apparently one more than necessary. But the equations all involve trigonometric functions of the angles. All the inverse trigonometric functions have two solutions (apart from the periodicity outside the range $-180°$ to $+180°$). The physics of the problem resolves much of the ambiguity. The angles $\beta$, $\mu$, $\omega$, and $\phi$ are restricted to the first quadrant. The value of $\nu$ can be found uniquely from Equation (8); since both $x_1$ and $y_1$ are known separately, the inverse tangent is not ambiguous. But $\alpha$ may take on any value $-180° < \alpha < -180°$, and it is not immediately clear how to eliminate one of the solutions of Equation (1) or (3) without using the other equation.

The solution process for Equation (3) derived in Appendix A gives an expression for $\cos\alpha$:

$$\cos\alpha = \frac{-B \pm A(A^2+1-B^2)^{1/2}}{A^2+1} . \tag{17a}$$

This may be equated to the expression for $\cos\alpha$ that is found from Equation (1) to give an identity. The identity is

$$\begin{aligned} &\pm(4 \sin^2\beta \cos^2\omega - \sin^2\nu \cos^2\phi)^{1/2} \\ &= \mathrm{sign}\ (\sin\nu) \times (\sin\mu - \cos\nu \cos\phi), \end{aligned} \tag{17b}$$

in terms of the original variables. The ± sign on the left side is the same as that in Equation (16). So the sign of the expression on the right side of Equation (17b) tells which sign to pick in Equation (16), thus resolving the ambiguity.

## 2.7 Distribution over $\alpha$ and $\beta$

The histogram that approximates the bivariate distribution as a function of $\alpha$ and $\beta$ is very easily found by setting up an array, each of whose cells (or "bins") is indexed by a specific range of $\alpha$ and $\beta$, and accumulating the corresponding pixel values in each cell. The choice of those "class intervals" for $\alpha$ and $\beta$ is of some concern. Because the histogram is constructed from a finite number of samples, if the subdivision is too fine the shape of the distribution will be masked by statistical fluctuations. On the other hand, if the subdivision is too coarse, the shape of the distribution will be blurred (an extreme case would be a histogram with only a single bin that contains all the points). Between the extremes there is a fairly wide range of reasonable values.

Sturges [8] has given a criterion for choosing a class interval. For a statistical series of range $R$ with $N$ items, the Sturges criterion for the optimal class interval is

$$C = R/(1 + 3.322 \log_{10} N). \tag{18}$$

It is based on the principle that the proper distribution into classes is given, for all numbers that are powers of 2, by a series of binomial coefficients. (Note that $3.322 = 1/\log_{10}2$.) For example, 16 items would be divided normally into 5 classes, with class frequencies 1, 4, 6, 4, 1. So if a statistical series had 16 items with values ranging from 20 to 70, or a range of 50 points, it should be divided into five classes of 10 points each; the class interval would be 10.

In the present case, each digitized image consists of $512 \times 512 = 262{,}144 = 2^{18}$ pixels. Thus, there are 262,144 values of $\alpha$ and 262,144 values of $\beta$. From Equation (18), $C = R/19$. Now, in practice, the class interval is chosen to be some convenient value near the theoretically optimum interval. If we use $C = R/18$, then $C = 360°/18 = 20°$ for $\alpha$ and $C = 90°/18 = 5°$ for $\beta$. However, it is clear that $\beta = 90°$ is physically impossible, and very large (near 90°) values are extremely unlikely. In fact, Cox and Munk report that $\beta = 25° - 30°$ may be a practical maximum [2]. So the range $R$ within which values occur is less than 90°; consequently, it is reasonable to choose a class interval or "bin size" smaller than 5°. An interval of 1° was chosen for $\beta$. For $\alpha$, an interval of 10° (instead of 20°) was chosen to obtain better resolution. "Bins" of these widths were set up to cover $-180° < \alpha \leqslant 180°$ and $0° \leqslant \beta < 90°$. So the histogram array contained $36 \times 90 = 3240$ cells.

## 2.8 Distribution over components of slope

Cox and Munk [1-3] related the statistics of the sea surface to the statistics of the distribution of brightness over components of slope. Those components are

$$Z_x = \tan\beta \sin\alpha^* \tag{19a}$$
$$Z_y = \tan\beta \cos\alpha^* \tag{19b}$$

where $\beta$ = sea-surface tilt as defined before, and $\alpha^*$ = azimuth of ascent measured clockwise from some axis. When that axis points toward the sun, $\alpha^* = \alpha$. But Cox and Munk found that the principal axes of the distribution are in the direction of the wind and crosswind [1-2].

So when $\alpha^*$ is measured relative to the wind direction, the principal axes are the coordinate axes. It is easily shown that this $\alpha^*$ is given by

$$\alpha^* = \alpha + (180° - \Omega - Az) \tag{20}$$

where $\Omega$ = direction of wind, measured clockwise from N (similarly to $Hdg$) and $\alpha$ and $Az$ were defined previously. The angle $\Omega$ gives the direction from which the wind is blowing, in agreement with common usage: a west wind is one that blows from the west.

It is clear that $[-1, 1]$ is likely to be a sufficient range for both $Z_x$ and $Z_y$, although in principle they are unbounded. This is so because $|\sin\alpha^*| \leqslant 1$ and $|\cos\alpha^*| \leqslant 1$, and even when $|\sin\alpha^*| = 1$ or $|\cos\alpha^*| = 1$, $\tan\beta \leqslant 1$ except for $\beta > 45°$, which is unlikely. So the histogram was set up for $-1 \leqslant Z_x \leqslant 1$ and $-1 \leqslant Z_y \leqslant 1$. The $\alpha$-$\beta$ histogram has 36 × 45 = 1620 cells over the range of $\beta$ up to 45°. The $Z_x$-$Z_y$ histogram was given approximately the same number of cells by dividing the $Z_x$ and $Z_y$ ranges into 41 intervals each (the odd number causes one interval—the central one—to be centered at zero). The histogram is constructed by calculating $Z_x$ and $Z_y$ for each pixel, then adding the pixel value to the appropriate cell.

## 2.9 Correction for nonzero roll

The equations for the sun glitter analysis that were derived in preceding sections are based on a geometry in which the photographic film is horizontal. Figure 4 illustrates that when the film is not horizontal the intercept of a light ray with the film changes, in general. The axis of rotation, which passes through the center of the camera's aperture, is normal to the plane of the figure. The dashed lines represent the axis of the camera's imaging geometry, normal to the film in each case. The solid arrow shows a light ray, which intersects the horizontal and tilted image planes in different positions in the respective coordinate systems. It is clear that there is a similar projective effect on the component perpendicular to the plane of the figure (except along the line of intersection of the two film planes).

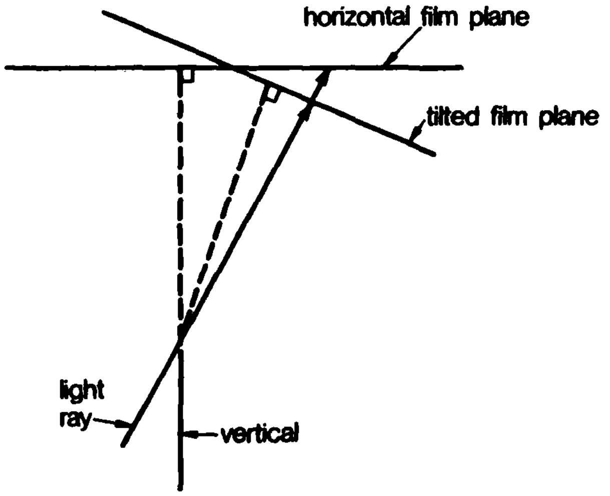

*Figure 4. Effect of camera tilt on image point position.*

The effect enters the analysis through $\mu$ and $\nu$, and alters the values found for $\omega$, $\beta$, and $\alpha$. If the plane of the film is tilted and $\mu$ and $\nu$ are calculated from the (tilted) image point position by the procedure of Section 2.3, the $\alpha$ and $\beta$ that result will be in error.

A solution is to calculate a corrected set of $\mu$, $\nu$ that correspond to the position on a hypothetical horizontal film that the same light ray would reach. To do this, define two coordinate systems $x_1$-$y_1$-$z_1$ and $x_2$-$y_2$-$z_2$ with a common origin O at the camera's aperture. The former system has the $x_1$-$y_1$ plane horizontal with $z_1$ pointing upward. The $x_2$-$y_2$-$z_2$ system is rotated an angle $r$ about the common $y$ axis with respect to the other coordinate system; $r$ is positive for a clockwise rotation from system 1 to system 2.

The $x_2$-$y_2$-$z_2$ system is oriented according to the attitude of the aircraft with nonzero roll angle $r$: the $x_2$-axis points toward the starboard wing and the $y_2$-axis points toward the nose. Figure 5 illustrates the two coordinate systems.

The (hypothetical) film plane in the first system is the plane $z_1 = f$; in the second system the (true) film plane is $z_2 = f$. Here $f$ is the camera's focal length. The $x$-$y$ coordinates of points in the film plane (either one) will be denoted by an extra subscript "f." Thus, $(x_{1f}, y_{1f})$ correspond to the $(x_1, y_1)$ that were used in Section 2.3 to calculate $\mu$ and $\nu$. They are the quantities needed here.

The transformation between arbitrary coordinates in the two systems is

$$\begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} = \begin{bmatrix} \cos & 0 & \sin r \\ 0 & 1 & 0 \\ -\sin r & 0 & \cos r \end{bmatrix} \begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix} . \tag{21}$$

Now consider a light ray that passes through O and strikes the actual film plane $z_2 = f$ at the point $P_2 = (z_2, y_2, f)$. We wish to find the point where the same ray would strike the hypothetical film plane $z_1 = f$. This can be easily done in three stages. First, from Equation (21), the coordinates of $P_2$ in the $x_1$-$y_1$-$z_1$ reference frame are

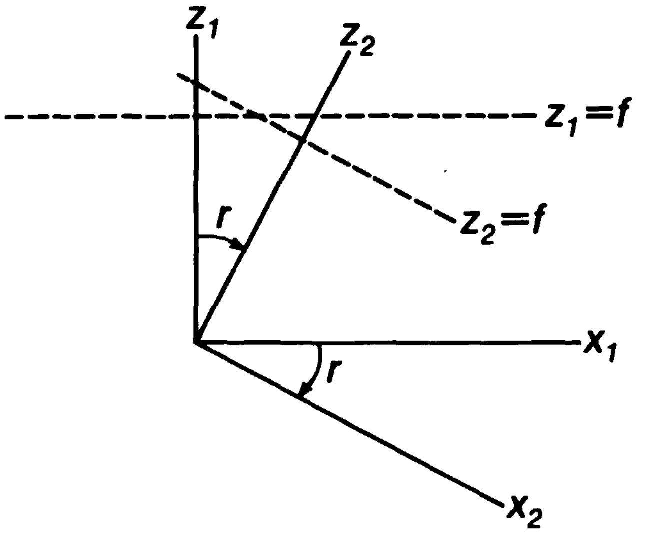

*Figure 5. Relationship between horizontal and rolled coordinate system.*

$$\begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} = \begin{bmatrix} x_{2f}\, cos\ r + f\, sin\ r \\ y_{2f} \\ -x_{2f}\, sin\ r + f\, cos\ r \end{bmatrix} . \qquad (22)$$

These are not the coordinates of the point $P_1$ where the light ray strikes the hypothetical film plane $z_1 = f$. To find those coordinates we next write the equation of the line that passes through the points $O$ and $P_2$. Since $O$ is the origin, substituting from Equation (22), that equation is

$$\frac{x_1}{x_{2f}\, cos\ r + f\, sin\ r} = \frac{y_1}{y_{2f}} = \frac{z_1}{f\, cos\ r - x_{2f}\, sin\ r} . \qquad (23)$$

This is the equation of the light ray in the $x_1$-$y_1$-$z_1$ system. From it, the coordinates of $P_1$ are

$$x_{1f} = \frac{f\,(f\, sin\ r + x_{2f}\, cos\ r)}{(f\, cos\ r - x_{2f}\, sin\ r)} , \qquad (24a)$$

$$y_{1f} = \frac{f\, y_{2f}}{(f\, cos\ r - x_{2f}\, sin\ r)} , \qquad (24b)$$

$$z_{1f} = f . \qquad (24c)$$

From this we can calculate values for $\mu$ and $\nu$ consistent with the geometry that was originally assumed.

From Equation (7)

$$tan\mu = (x_{1f}^2 + y_{1f}^2)^{1/2}/f$$

$$= \frac{\left\{ (x_{2f}^2 + y_{2f}^2) + f\, sin\ r\, [f\, sin\ r + x_{2f}(cos\ r + D)] \right\}^{1/2}}{fD} , \qquad (25)$$

where $D = cos\ r - x_{2f}\, sin\ r/f$. Squaring Equation (25),

$$tan^2\mu = \frac{1}{D} \left\{ \left( \frac{x_{2f}^2 + y_{2f}^2}{f^2} \right) \frac{sin\ r\, [f\, sin\ r + x_{2f}(cos\ r + D)]}{f} \right\} . \qquad (26)$$

The value of $\nu$ is found from Equation (8), where Equations (24a) and (24b) must be substituted for $x_1$ and $y_1$, respectively.

# 3. Limitations of analysis

The foregoing text has described the derivation of the sun glitter analysis algorithms. The discussion will conclude with a summary of the approximations involved, some of which are not explicit.

Three of the approximations relate to the geometry of the problem:

- A spherical earth was assumed.
- No refraction correction was applied to the apparent position of the sun.
- Mean solar time was used in calculating the sun's position.

All have some effect on the sun's position in the assumed geometry. The first causes relatively negligible effects. The second causes an error of up to about ½° in the sun's elevation. The error due to the third can cause an error of up to 4° in the local hour angle. Both errors are then propagated to the quantities that depend on $\phi$ and $h$.

Two other approximations involve photometry:

- No correction of the digitized values to obtain original luminance values was performed.
- No background correction was performed.

With respect to the first of these, the digitized positive image brightness values are almost certainly not directly proportional to the reflected brightness field that illuminated the film. It is the latter that should be used to construct the brightness distributions. While the digitized values are probably related to the original brightness values by a monotonically increasing function, the form of that function is unknown. It involves the properties of the camera's lens, the response of the film (D − log E curve), the uniformity of illumination of the transparency when it is digitized, the response of the digitizing camera, and the law by which that camera's output is transformed to digitized values. The background correction refers to compensation for the radiation from the reflection of skylight at the sea surface and from sunlight scattered by particles beneath the sea surface. It was implicitly assumed in the foregoing discussion that all of the light that reaches the film is from specular reflection of sunlight from the sea surface. Cox and Munk devoted considerable attention to background correction [2].

# 4. Program documentation

This section contains formal documentation of a main program and 15 subroutines that were written (in FORTRAN IV) to perform sun glitter analysis of aerial photographs. Each routine's documentation is self-contained. Thus, in a few cases, equations are numbered independently starting with Equation (1). The written documentation is followed by user instructions, program listings, and a sample run stream.

## 1. Name

GMAIN2

## 2. Purpose

The purpose of GMAIN2 is to calculate histograms that approximate distributions of sea-surface slope by analyzing aerial photographs of the sun's reflection on the water.

## 3. Calling sequence

GMAIN2 is a main program.

## 4. Input—output

### 4.1 Input

The following items are read from logical unit 5 (user terminal):

LAT = latitude (real)
LONG = longitude (real)
DECL = sun's declination (real)
TIME = Zulu time (real)
HDG = aircraft heading (real)
ROLL = aircraft roll angle (real)
WIND = wind direction (real)
F = camera focal length (real)
HEIGHT = image height (or width) (real)
IR1 = first row of region to process (integer)
IR2 = last row of region to process (integer)
IE1 = first column of region to process (integer)
IE2 = last column of region to process (integer)
FILE = image file name (CHARACTER*8)

The above items are read once per run. All angles are in degrees.

In addition, records containing image data are read from logical unit 1. Each record consists of

BUFFER = 512-word array that contains two consecutive scan lines, 1 pixel per byte. (integer)

### 4.2 Output

Messages that constitute the other half of an interactive dialogue are displayed on logical unit 6 (user terminal). The messages solicit the items that are read from logical unit 5, as described in 4.2.

All items read from logical unit 5 are printed on logical unit 7 (line printer). In addition, the following items are printed.

AZ = solar azimuth (degrees) (real)
HIST = distribution of integrated image intensity vs. $\alpha$ and $\beta$ (real)
A1 = minimum $\alpha$ (real)
A2 = maximum $\alpha$ (real)
DALPHA = $\alpha$ increment (real)
B1 = minimum $\beta$ (real)
B2 = maximum $\beta$ (real)
DBETA = $\beta$ increment (real)
HISTB = HIST summed over $\alpha$, as a function of $\beta$ (real)
XMEANB = mean of $\beta$ with respect to distribution HISTB (real)
VARB = variance of $\beta$ with respect to distribution HISTB (real)
HISTCU = distribution of integrated image intensity vs. crosswind and upwind components of sea-surface slope (real)
ZMIN = minimum value of slope components (real)
ZMAX = maximum value of slope components (real)
ZDELT = slope component increment (real)
SLICE = first crosswind, then upwind slice (along coordinate axes) of HISTCU(both are printed) (real)
RMS = root-mean-square value of slope in slice (computed for both) (real)

In the event of certain error conditions in the calculations performed by subroutines, appropriate messages are printed on logical unit 7.

### 4.3 File Storage

None.

## 5. Exits

There are no nonstandard exits.

## 6. Usage

GMAIN2 is written in FORTRAN IV and is presently implemented on the HP-3000 operating under MPE-III.

## 7. External interfaces

### 7.1 System subroutines

DCOS
DSIN

### 7.2 Other programs called

GETPIX
HDSPLY

MNVAR
NRMLZ
RMSARY
STEP1
STEP2
STEP3
STEP4
STEP5
STEP6
STEP7

### 7.3 External storage used

None.

## 8. Performance specifications

### 8.1 Storage

Stack: 11,765 words
Code: 2686 words

### 8.2 Execution time

TBD.

### 8.3 I/O Load

I/O statements are described in 4.1 and 4.2. The number of records read from unit 1 depends on *IR*1, *IR*2, *IE*1, *IE*2.

### 8.4 Restrictions

Image files are assumed to be in the format produced by CI.

## 9. Method

Constants are read in and program variables are initialized. GMAIN2 then skips to the first record needed from logical unit 1. STEP1 is called to calculate solar elevation and azimuth, and the correction to reference $\alpha$ to the wind direction is computed. The first record is read from unit 1 and unpacked to one pixel per word. Then a loop begins over the rows and columns to be processed. Coordinates of the current image point are calculated, and STEP2-STEP5 are called to calculate (ultimately) the $\alpha$ and $\beta$ for that point. STEP6 and STEP7 are then called to increment HIST and HISTCU. At the end of the loop, another record is read from unit 1 when necessary. The remainder of the program is concerned with the calculation of statistical measures and with output.

## 10. Comments

GMAIN2 does not include a calculation to recover the original light intensity from digitized image values.

## 1. Name

STEP1

## 2. Purpose

The purpose of STEP1 is to calculate local hour angle, solar elevation, and solar azimuth. The last two are returned to the calling routine.

## 3. Calling sequence

CALL STEP1 (TIME, LONG, LAT, DECL, PHI, CDPHI, SDPHI, AZ, $n).

where:

TIME = Zulu time (real)
LONG = longitude W of Greenwich (real)
LAT = latitude N of equator (real)
DECL = solar declination (real)
PHI = solar elevation (real)
CDPHI = cosine of PHI (double precision)
SDPHI = sine of PHI (double precision)
AZ = solar azimuth (real)
n = statement number in calling program to which control is returned if AZ cannot be calculated

## 4. Input—output

### 4.1 Input

There are no input statements.

### 4.2 Output

There are no output statements.

### 4.3 File Storage

None.

## 5. Exits

If the expression for $|\sin (AZ)| > 1$, AZ is set to zero and the error return is taken.

## 6. Usage

STEP1 is written in FORTRAN IV and is presently implemented on the HP-3000 operating under MPE-III.

## 7. External interfaces

### 7.1 System subroutines

DCOS
DSIN

### 7.2 Other programs called

ASIN

### 7.3 External storage used

None.

## 8. Performance specifications

### 8.1 Storage

Stack: 56 words
Code: 231 words

### 8.2 Execution time

TBD.

### 8.3 I/O Load

There are no I/O statements.

### 8.4 Restrictions

None.

## 9. Method

Hour angle is calculated by

$$h = \{(t - 720)/4\} - \lambda \ .$$

Solar elevation is calculated by

$$\sin\phi = \sin\delta \sin L + \cos\delta \cos L \cos h \ .$$

Solar azimuth is calculated by

$$\sin Az = -\cos\delta \sin h / \cos\phi \ .$$

All symbols are defined in Section 2 of this report.

## 10. Comments

None.

## 1. Name

STEP2
Source file STEP2A

## 2. Purpose

The purpose of STEP2 is to calculate image angular coordinates.

## 3. Calling sequence

CALL STEP2 (X, Y, F, HDG, AZ, MU, NU, CDR, SDR, IFLAG).

where:

X = image coordinate toward starboard wing (real)
Y = image coordinate toward aircraft nose (real)
F = camera lens focal length (real)
HDG = aircraft heading (real)
AZ = solar azimuth (real)
MU = angle between reflected ray and vertical (real)
NU = angle of image point measured clockwise from an axis that points away from the sun (real)
CDR = cosine of aircraft roll angle (double precision)
SDR = sine of aircraft roll angle (double precision)
IFLAG = flag that indicates error in MU calculation when roll is nonzero (integer)
= 0—no error
= 1—error

## 4. Input—output

### 4.1 Input

There are no input statements.

### 4.2 Output

There are no output statements.

### 4.3 File Storage

None.

## 5. Exits

There are no nonstandard exits.

## 6. Usage

STEP2 is written in FORTRAN IV and is presently implemented on the HP-3000 operating under MPE-III.

## 7. External interfaces

### 7.1 System subroutines

DATAN
DATAN2
DSQRT

### 7.2 Other programs called

None.

### 7.3 External storage used

None.

## 8. Performance specifications

### 8.1 Storage

Stack: 43 words
Code: 320 words

### 8.2 Execution time

TBD.

### 8.3 I/O Load

There are no I/O statements.

### 8.4 Restrictions

None.

## 9. Method

If the roll angle $r = 0$, the output quantities are calculated from

$$\tan\mu = (x_1^2 + y_1^2)^{1/2}/f, \text{ and}$$

$$= Az + Hdg + 90^\circ - \tan^{-1}(y_1/x_1)$$

where the symbols are defined in Section 2 of this report. In the above equations $x_1 = X$ and $y_1 = Y$, cartesian coordinates in the image.

If $r \neq 0$, then

$$\tan\mu = (x_{1f}^2 + y_{1f}^2)^{1/2}/f$$

$$= \frac{\left\{(x_{2f}^2 + y_{2f}^2) + f \sin r\,[f \sin r + x_{2f}(\cos r + D)]\right\}^{1/2}}{fD},$$

where

$$x_{2f} = X, \; y_{2f} = Y,$$

$$x_{1f} = \frac{f\,(f \sin r + x_{2f} \cos r)}{(f \cos r - x_{2f} \sin r)} \;,$$

$$y_{1f} = \frac{f\, y_{2f}}{(f \cos r - x_{2f} \sin r)} \;,$$

$$D = \cos r - x_{2f} \sin r/f.$$

The value of $\nu$ is found from

$$\nu = Az + Hdg + 90^\circ - \tan^{-1} (y_{1f}/x_{1f}).$$

It is necessary to use the numerators of the expressions for $x_{1f}$ and $y_{1f}$ only in the calculation of $\nu$. The denominators are always greater than zero on physical grounds. If the quantity whose square root is (normally) taken in the formula for $\tan\mu$ is negative, MU is set to zero and IFLAG is set to 1. This should only occur from roundoff error when $\mu \approx 0$.

10. Comments

None.

1. Name

STEP3

2. Purpose

The purpose of STEP3 is to calculate the cosine of the angle of incidence or reflection.

3. Calling sequence

CALL STEP3 (CDPHI, SDPHI, MU, NU, COSOMG).

where:

CDPHI = cosine of solar elevation (double precision)
SDPHI = sine of solar elevation (double precision)
MU = angle between reflected ray and vertical (real)
NU = angle of image point measured clockwise from an angle that points away from the sun (real)
COSOMG = cosine of angle of incidence or reflection (double precision)

4. Input—output

4.1 Input

There are no input statements.

4.2 Output

There are no output statements.

4.3 File Storage

None.

5. Exits

There are no nonstandard exits.

6. Usage

STEP3 is written in FORTRAN IV and is presently implemented on the HP-3000 operating under MPE-III.

7. External interfaces

7.1 System subroutines

DCOS
DSIN
DSQRT

7.2 Other programs called

None.

7.3 External storage used

None.

8. Performance specifications

8.1 Storage

Stack: 26 words
Code: 133 words

8.2 Execution time

TBD.

8.3 I/O Load

There are no I/O statements.

8.4 Restrictions

None.

9. Method

The equations that are implemented are

$$\cos 2\omega = -\cos\phi \sin\mu \cos\nu + \sin\phi \cos\mu,$$

$$\cos\omega = \{½ (\cos 2\omega + 1)\}^{½}.$$

The symbols are defined in Section 2 of this report. It is assumed that ω is in the first quadrant.

10. Comments

None.

## 1. Name

STEP4

## 2. Purpose

The purpose of STEP4 is to calculate sea-surface tilt.

## 3. Calling sequence

CALL STEP4 (MU, SDPHI, COSOMG, BETA)

where:

MU = angle between reflected ray and vertical (real)
SDPHI = sine of solar elevation (double precision)
COSOMG = cosine of angle of incidence or reflection (double precision)
BETA = sea-surface tilt (real)

## 4. Input—output

### 4.1 Input

There are no input statements.

### 4.2 Output

There are no output statements.

### 4.3 File Storage

None.

## 5. Exits

There are no nonstandard exits.

## 6. Usage

STEP4 is written in FORTRAN IV and is presently implemented on the HP-3000 operating under MPE-III.

## 7. External interfaces

### 7.1 System subroutines

DCOS

### 7.2 Other programs called

ACOS

### 7.3 External storage used

None.

## 8. Performance specifications

### 8.1 Storage

Stack: 14 words
Code: 64 words

### 8.2 Execution time

TBD.

### 8.3 I/O Load

There are no I/O statements.

### 8.4 Restrictions

None.

## 9. Method

The sea-surface tilt angle is found from

$$\cos\beta = (\cos\mu + \sin\phi)/2 \cos\omega,$$

where the symbols are defined in Section 2 of this report. If $\cos\omega = 0$, which implies glancing incidence, $\beta$ is set to zero.

## 10. Comments

None.

## 1. Name

STEP5
Source file STEP5A

## 2. Purpose

The purpose of STEP5 is to calculate the local azimuth of ascent of the sea surface.

## 3. Calling sequence

CALL STEP5 (NU, BETA, PHI, CDPHI, COSOMG, MU, ALPHA, $n).

where:

NU = angle of image point measured clockwise (real)
BETA = sea-surface tilt (real)
PHI = solar elevation (real)
CDPHI = cosine of PHI (double precision)
COSOMG = cosine of angle of incidence or reflection (double precision)
MU = angle between reflected ray and vertical (real)
ALPHA = local azimuth of ascent of sea surface (real)
n = statement number in calling program to which control is returned if ALPHA cannot be calculated

## 4. Input—output

### 4.1 Input

There are no input statements.

### 4.2 Output

There are no output statements.

### 4.3 File Storage

None.

## 5. Exits

In case the square root in the expression for ALPHA (see Section 9) involves a negative number, that square root is set to zero in the ALPHA calculation and the error return is taken.

## 6. Usage

STEP5 is written in FORTRAN IV and is presently implemented on the HP-3000 operating under MPE-III.

## 7. External interfaces

### 7.1 System subroutines

DATAN2
DCOS
DSIN
DSQRT
DTAN

### 7.2 Other programs called

ACOS

### 7.3 External storage used

None.

## 8. Performance specifications

### 8.1 Storage

Stack: 71 words
Code: 359 words

### 8.2 Execution time

TBD.

### 8.3 I/O Load

There are no I/O statements.

### 8.4 Restrictions

None.

## 9. Method

The local azimuth of sea-surface ascent is found from

$$\alpha = \tan^{-1}\left[\frac{AB \pm (A^2+1-B^2)^{1/2}}{-B \pm A(A^2+1-B^2)^{1/2}}\right]$$

where

$$A = 1/\tan \nu,$$

$$B = \cos\phi/2 \sin\beta \cos\omega,$$

and the other symbols are defined in Section 2 of this report, which also shows that the sign to use in both the numerator and the denominator of $\tan\alpha$ is the sign of the expression

$$\sin\nu \, (\sin\mu - \cos\nu \cos\phi).$$

Special cases are handled as described in Appendix A. When $A^2 + 1 - B^2 < 0$, the square root cannot be calculated. This should occur only as a result of roundoff error when $A^2 + 1 - B^2$ is very near zero, so in this case $\alpha$ is calculated from

$$\alpha = \tan^{-1}(AB/-B)$$

and the error return is taken.

## 10. Comments

None.

## 1. Name

STEP6

## 2. Purpose

The purpose of STEP6 is to increment the $\alpha$-$\beta$ histogram.

## 3. Calling sequence

CALL STEP6 (ALPHA, BETA, POINT, HIST, NALPHA, NBETA, DALPHA, DBETA).

where:

| | | | |
|---|---|---|---|
| ALPHA | = | azimuth of ascent of sea surface | (real) |
| BETA | = | sea-surface tilt | (real) |
| POINT | = | image point intensity value | (integer) |
| HIST | = | histogram array | (real) |
| NALPHA | = | number of ALPHA increments | (integer) |
| NBETA | = | number of BETA increments | (integer) |
| DALPHA | = | size of an ALPHA increment | (integer) |
| DBETA | = | size of a BETA increment | (integer) |

## 4. Input—output

### 4.1 Input

There are no input statements.

### 4.2 Output

There are no output statements.

### 4.3 File Storage

None.

## 5. Exits

There are no nonstandard exits.

## 6. Usage

STEP6 is written in FORTRAN IV and is presently implemented on the HP-3000 operating under MPE-III.

## 7. External interfaces

### 7.1 System subroutines

None.

### 7.2 Other programs called

None.

### 7.3 External storage used

None.

## 8. Performance specifications

### 8.1 Storage

Stack: 4 words
Code: 48 words

### 8.2 Execution time

TBD.

### 8.3 I/O Load

There are no I/O statements.

### 8.4 Restrictions

None.

## 9. Method

The indices of the appropriate histogram cell are calculated from ALPHA and BETA, and the pixel value is added to that cell.

## 10. Comments

None.

## 1. Name

STEP7

## 2. Purpose

The purpose of STEP7 is to increment the crosswind-upwind histogram.

## 3. Calling sequence

CALL STEP7 (ALPHA, BETA, DELTA, POINT, HIST, N, ZDELT, ZMAX).

where:

ALPHA = azimuth of ascent of sea surface (real)
BETA = sea-surface tilt (real)
DELTA = correction to reference ALPHA to wind direction (real)
POINT = image point intensity value (integer)
HIST = histogram array (real)
N = number of increments along each histogram axis (integer)
ZDELT = size of an increment (real)
ZMAX = upper limit for each axis (real)

## 4. Input—output

### 4.1 Input

There are no input statements.

### 4.2 Output

There are no output statements.

### 4.3 File Storage

None.

## 5. Exits

There are no nonstandard exits.

## 6. Usage

STEP7 is written in FORTRAN IV and is presently implemented on the HP-3000 operating under MPE-III.

## 7. External interfaces

### 7.1 System subroutines

COS
SIN
TAN

### 7.2 Other programs called

None.

### 7.3 External storage used

None.

## 8. Performance specifications

### 8.1 Storage

Stack: 14 words
Code: 91 words

### 8.2 Execution time

TBD.

### 8.3 I/O Load

There are no I/O statements.

### 8.4 Restrictions

None.

## 9. Method

DELTA is used to reference ALPHA to the wind direction. Then the two histogram coordinates are found from

$$Z_x = \tan\beta \cdot \sin\alpha^*,$$

$$Z_y = \tan\beta \cos\alpha^*,$$

The symbols are defined in Section 2 of this report. Then the indices of the histogram cell that contains $Z_x$ and $Z_y$ are calculated, and the pixel value is added to that cell. Since $\tan\beta$ is unbounded in principle, the cell indices are corrected if necessary to lie within the assumed range.

## 10. Comments

None.

1. Name

GETPIX
(GET PIXels)

2. Purpose

The purpose of GETPIX is to unpack a read buffer that contains two image lines packed one pixel per byte into a two-row array in which each word contains a pixel.

3. Calling sequence

CALL GETPIX (BUFFER, ROW).

where:

| | | |
|---|---|---|
| BUFFER | = read buffer | (integer) |
| ROW | = two-row array | (integer) |

4. Input—output

4.1 Input

There are no input statements.

4.2 Output

There are no output statements.

4.3 File Storage

None.

5. Exits

There are no nonstandard exits.

6. Usage

GETPIX is written in FORTRAN IV and is presently implemented on the HP-3000 operating under MPE-III.

7. External interfaces

7.1 System subroutines

None.

7.2 Other programs called

None.

7.3 External storage used

None.

8. Performance specifications

8.1 Storage

Stack: 3 words
Code: 48 words

8.2 Execution time

TBD.

8.3 I/O Load

There are no I/O statements.

8.4 Restrictions

None.

9. Method

The first 256 words of BUFFER are unpacked into the first row of ROW, with each consecutive byte from the former going in order into words in the latter. The last 256 words of BUFFER are unpacked similarly into the second row of ROW.

10. Comments

None.

## 1. Name

HDSPLY

## 2. Purpose

The purpose of HDSPLY is to display a univariate histogram on the line printer.

## 3. Calling sequence

CALL HDSPLY (HIST, N, NAME, X0, STEP).

where:

HIST = histogram array (real)
N = number of histogram cells (integer)
NAME = name of independent variable (CHARACTER*10)
X0 = lower end of independent variable range (real)
STEP = independent variable step size (real)

## 4. Input—output

### 4.1 Input

There are no input statements.

### 4.2 Output

A heading and the histogram itself are printed on logical unit 7. The heading is

(value of NAME) OCCUPANCY

The histogram is printed by giving "breakpoint" values of the independent variable and cell occupancies. The items printed are:

X = current breakpoint value (real)
HIST(I) = histogram occupancy in cell I (real)
STAR = ASCII "*" (CHARACTER*1)

X and HIST(I) are interleaved in such a way that each HIST(I) is printed between values of the independent variable that mark the boundaries of that cell. Along with each HIST(I), STAR is printed a number of times that is proportional to HIST(I).

### 4.3 File Storage

None.

## 5. Exits

There are no nonstandard exits.

## 6. Usage

HDSPLY is written in FORTRAN IV and is presently implemented on the HP-3000 operating under MPE-III.

## 7. External interfaces

### 7.1 System subroutines

None.

### 7.2 Other programs called

None.

### 7.3 External storage used

None.

## 8. Performance specifications

### 8.1 Storage

Stack: 11 words
Code: 174 words

### 8.2 Execution time

TBD.

### 8.3 I/O Load

Each call of HDSPLY produces 2N + 2 lines of output on logical unit 7.

### 8.4 Restrictions

None.

## 9. Method

HIST is searched to find the maximum occupancy, which is used to calculate a scale factor such that a row of 100 asterisks is printed for the maximum-occupancy cell and a proportionate number of asterisks for each other cell. Next, the title is printed. This is followed by lines of independent variable values alternating with lines giving cell occupancy and a row of asterisks for illustration. Between 0 and 100 asterisks are printed. See 4.2 for more information on the output.

## 10. Comments

None.

## 1. Name

ACOS
Source file MLACOS

## 2. Purpose

The purpose of ACOS is to calculate the single-precision inverse cosine of a value.

## 3. Calling sequence

Y = ACOS (X)

where:

| | | |
|---|---|---|
| ACOS | = function value (radians) | (real) |
| X | = cosine of angle to be determined | (real) |

## 4. Input—output

### 4.1 Input

There are no input statements.

### 4.2 Output

There are no output statements.

### 4.3 File Storage

None.

## 5. Exits

There are no nonstandard exits.

## 6. Usage

ACOS is written in FORTRAN IV and is presently implemented on the HP-3000 operating under MPE-III.

## 7. External interfaces

### 7.1 System subroutines

None.

### 7.2 Other programs called

ASIN

### 7.3 External storage used

None.

## 8. Performance specifications

### 8.1 Storage

Stack: 0 words
Code: 11 words

### 8.2 Execution time

TBD.

### 8.3 I/O Load

There are no I/O statements.

### 8.4 Restrictions

None.

## 9. Method

The arccosine is calculated from

$$\cos^{-1} x = \pi/2 - \sin^{-1} x.$$

This method is used in Reference 9. ACOS returns the principal value of the arccosine. That is, if $y = \cos^{-1} x$,

$$0 \leqslant y \leqslant \pi/2 \quad \text{for} \quad x \geqslant 0$$

$$\pi/2 < y \leqslant \pi \quad \text{for} \quad x < 0$$

## 10. Comments

See the documentation for ASIN in order to complete the above discussion.

### 1. Name

ASIN
Source file MLASIN

### 2. Purpose

The purpose of ASIN is to calculate the single-precision inverse sine of a value.

### 3. Calling sequence

Y = ASIN (X)

where:

ASIN = function value (radians) (real)
X = sine of angle to be determined (real)

### 4. Input—output

### 4.1 Input

There are no input statements.

### 4.2 Output

There are no output statements.

### 4.3 File Storage

None.

### 5. Exits

There are no nonstandard exits.

### 6. Usage

ASIN is written in FORTRAN IV and is presently implemented on the HP-3000 operating under MPE-III.

### 7. External interfaces

### 7.1 System subroutines

ATAN
SQRT

### 7.2 Other programs called

None.

### 7.3 External storage used

None.

### 8. Performance specifications

### 8.1 Storage

Stack: 2 words
Code: 75 words

### 8.2 Execution time

TBD.

### 8.3 I/O Load

There are no I/O statements.

### 8.4 Restrictions

None.

### 9. Method

The arcsine is calculated from a formula based on [7],

$$\sec^2 y - \tan^2 y = 1. \qquad (1)$$

Manipulation of this equation produces

$$\tan^2 y = \sec^2 y - 1 = \frac{1}{\cos^2 y} - 1 = \frac{\sin^2 y}{1 - \sin^2 y}$$

If x = sin y then y = $\sin^{-1}$ x, and

$$\tan^2 (\sin^{-1} x) = \frac{x^2}{1 - x^2}$$

from which we obtain

$$\sin^{-1} x = \tan^{-1} \left[ \left( \frac{x^2}{1 - x^2} \right)^{1/2} \right] \qquad (2)$$

This gives the correct result for $x \geqslant 0$. When $x < 0$, the relation [7]

$$\sin^{-1} (-x) = -\sin^{-1} x \qquad (3)$$

is used.

It is noted that when $|x| \approx 1$ the denominator on the right side of Equation (2) is very small. Considerable loss of precision due to roundoff error (in the denominator itself) is expected in that case. To improve the precision of the result as $|x| \to 1$, the argument range is reduced to $[0, \frac{1}{2}]$ by the identity [9]

$$sin^{-1}\, x = \pi/2 - 2\, sin^{-1} \left[ \left( \frac{1-x}{2} \right)^{\frac{1}{2}} \right] , \qquad (4)$$

along with Equation (3) for negative arguments.

The sine of any argument is restricted to the range $[-1, 1]$. If $x > 1$ or $x < -1$, ASIN uses

$$ASIN = \pi/2$$

along with Equation (3).

ASIN returns the principal value of the arcsine. That is, if $y = sin^{-1}\, x$,

$$0 < y \leqslant \pi/2 \quad \text{for} \quad x \geqslant 0$$

$$-\pi/2 \leqslant y < 0 \quad \text{for} \quad x < 0$$

## 10. Comments

None.

## 1. Name

MNVAR

## 2. Purpose

The purpose of MNVAR is to calculate the mean and variance of a normalized univariate distribution.

## 3. Calling sequence

CALL MNVAR (HIST, N, XMIN, XMAX, XMEAN, VAR).

where:

HIST = empirical density function (real)
N = number of cells in histogram HIST (integer)
XMIN = lower end of variable range (real)
XMAX = upper end of variable range (real)
XMEAN = mean of distribution (real)
VAR = variance of distribution (real)

## 4. Input—output

### 4.1 *Input*

There are no input statements.

### 4.2 Output

There are no output statements.

### 4.3 File Storage

None.

## 5. Exits

There are no nonstandard exits.

## 6. Usage

MNVAR is written in FORTRAN IV and is presently implemented on the HP-3000 operating under MPE-III.

## 7. External interfaces

### 7.1 System subroutines

None.

### 7.2 Other programs called

None.

### 7.3 External storage used

None.

## 8. Performance specifications

### 8.1 Storage

Stack: 10 words
Code: 52 words

### 8.2 Execution time

TBD.

### 8.3 I/O Load

There are no I/O statements.

### 8.4 Restrictions

None.

## 9. Method

The mean $m$ of a discrete probability distribution is the first moment of the distribution,

$$m = \sum_s x_s \, p_s \tag{1}$$

where $x_s$ is one of the values taken on by the random variable $X$ and $p_s$ is a value of the probability density function of the distribution.

$$p_s = Pr \{X = x_s\} \tag{2a}$$

$$\sum_s p_s = 1 \tag{2b}$$

The variance $\sigma^2$ is the second moment about the mean of the distribution,

$$\sigma^2 = \sum_s (x_s - m)^2 \, p_s \tag{3}$$

It is easily shown that this is equivalent to

$$\sigma^2 = \sum_s x_s^2 p_s - \Big(\sum_s x_s p_s\Big)^2 . \qquad (4)$$

Equations (1) and (4) are implemented, where HIST plays the role of the density function. The notation used above follows that of Reference 7.

## 10. Comments

None.

## 1. Name

NORM

## 2. Purpose

The purpose of NORM is to transform an angle outside the range $-180° < \text{angle} \leqslant 180°$ into the comparable angle within that range.

## 3. Calling sequence

X = NORM (ANGLE).

where:

| | | |
|---|---|---|
| ANGLE | = original angle | (real) |
| NORM | = transformed angle | (real) |

## 4. Input—output

### 4.1 Input

There are no input statements.

### 4.2 Output

There are no output statements.

### 4.3 File Storage

None.

## 5. Exits

There are no nonstandard exits.

## 6. Usage

NORM is written in FORTRAN IV and is presently implemented on the HP-3000 operating under MPE-III.

## 7. External interfaces

### 7.1 System subroutines

None.

### 7.2 Other programs called

None.

### 7.3 External storage used

None.

## 8. Performance specifications

### 8.1 Storage

Stack: 0 words
Code: 30 words

### 8.2 Execution time

TBD.

### 8.3 I/O Load

There are no I/O statements.

### 8.4 Restrictions

None.

## 9. Method

If ANGLE $> 180°$ then 360 is subtracted, repeatedly if necessary, until ANGLE $\leqslant 180°$. Similarly, if ANGLE $\leqslant -180°$ then 360 is added as many times as necessary. No action is taken if ANGLE is already in the correct range.

## 10. Comments

None.

## 1. Name

NRMLZ

## 2. Purpose

The purpose of NRMLZ is to normalize a two-dimensional histogram.

## 3. Calling sequence

CALL NRMLZ (HIST, NI, NJ).

where:

HIST = histogram array (real)
NI = number of rows in HIST (integer)
NJ = number of columns in HIST (integer)

## 4. Input—output

### 4.1 Input

There are no input statements.

### 4.2 Output

There are no output statements.

### 4.3 File Storage

None.

## 5. Exits

There are no nonstandard exits.

## 6. Usage

NRMLZ is written in FORTRAN IV and is presently implemented on the HP-3000 operating under MPE-III.

## 7. External interfaces

### 7.1 System subroutines

None.

### 7.2 Other programs called

None.

### 7.3 External storage used

None.

## 8. Performance specifications

### 8.1 Storage

Stack: 6 words
Code: 50 words

### 8.2 Execution time

TBD.

### 8.3 I/O Load

There are no I/O statements.

### 8.4 Restrictions

None.

## 9. Method

All of the elements of HIST are summed. The sum is used to normalize HIST so that the sum of all the normalized elements = 1.

## 10. Comments

None.

## 1. Name

RMSARY

## 2. Purpose

The purpose of RMSARY is to find the rms value of a discrete univariate distribution.

## 3. Calling sequence

CALL RMSARY (ARRAY, N, XMIN, XDELT, ARMS).

where:

ARRAY = unnormalized univariate distribution (real)
N = number of cells in distribution (integer)
XMIN = lower end of variable range (real)
XDELT = independent variable increment (real)
ARMS = rms value (real)

## 4. Input—output

### 4.1 Input

There are no input statements.

### 4.2 Output

There are no output statements.

### 4.3 File Storage

None.

## 5. Exits

There are no nonstandard exits.

## 6. Usage

RMSARY is written in FORTRAN IV and is presently implemented on the HP-3000 operating under MPE-III.

## 7. External interfaces

### 7.1 System subroutines

None.

### 7.2 Other programs called

None.

### 7.3 External storage used

None.

## 8. Performance specifications

### 8.1 Storage

Stack: 8 words
Code: 51 words

### 8.2 Execution time

TBD.

### 8.3 I/O Load

There are no I/O statements.

### 8.4 Restrictions

None.

## 9. Method

The rms value is calculated from

$$\sigma = \left[\sum_{s} x_s^2 f_s/N\right]^{1/2}$$

where $f_s$ is a frequency distribution that satisfies

$$\sum_{s} f_s = N.$$

Here ARRAY is taken to represent $\{f_s\}$.

## 10. Comments

None.

```
                                    MARCH 5, 1980

THE SUN GLITTER ANALYSIS PROGRAM IS NAMED GLITTER2.
IT IS CONSTRUCTED FROM THE MAIN PROGRAM IN THE USL GMAIN2RB
AND THE SUBROUTINES IN RL GSUBSRL AS FOLLOWS:

:PREP GMAIN2RB,GLITTER2;RL=GSUBSRL
:SAVE GLITTER2

THE SOURCE FILE FOR THE MAIN PROGRAM IS GMAIN2.
THE SOURCE FILES FOR THE SUBROUTINES HAVE THE SAME NAMES AS
THE ENTRY POINTS IN GSUBSRL WITH THE FOLLOWING EXCEPTIONS:

     STEP2A INSTEAD OF STEP2
     STEP5A INSTEAD OF STEP5
     MLACOS INSTEAD OF ACOS
     MLASIN INSTEAD OF ASIN

THE SUBROUTINES ARE ALSO IN THE USL FILES SUBSRB AND SUBS2RB.

GLITTER2 READS IMAGE DATA FROM LOGICAL UNIT 1 AND PRINTS ON
UNIT 7.  SO THE JOB SETUP IS

:FILE FTN07;DEV=LP;CCTL
:FILE FTN01=FORMAL FILE DESIGNATOR,OLD
:RUN GLITTER2
     ENTER DATA IN RESPONSE TO PROMPTS FROM PROGRAM.

DATA ITEMS ARE (ALL ANGLES IN DEGREES):

     LATITUDE, LONGITUDE, DECLINATION, TIME
          TIME IS GMT, 24-HOUR SYSTEM, FORM HHMM.MM...
          THUS 1810.75 MEANS 18 HR. 10.75 MIN. PAST
          MIDNIGHT, OR 6:10:45 PM, GREENWICH TIME.
     AIRCRAFT HEADING, ROLL
     WIND DIRECTION
     CAMERA FOCAL LENGTH, PICTURE HEIGHT (BOTH SAME UNITS)
     1ST ROW, LAST ROW, 1ST COLUMN, LAST COLUMN
          (IMAGE AREA TO PROCESS)
     IMAGE FILE NAME (TO LABEL PRINTOUT)

FOLLOWING ARE PROGRAM FILE LISTINGS AND A SAMPLE RUN STREAM.
```

```
$CONTROL MAP,CROSSREF,LABEL,FILE=1
C
C     MAIN PROGRAM TO CALCULATE SEA SURFACE SLOPE HISTOGRAMS:
C          1. ALPHA-BETA HISTOGRAM.
C          2. CROSSWIND-UPWIND HISTOGRAM.
C
C     MATTHEW LYBANON, CSC, FEBRUARY 15, 1980.
C     MODIFIED FEBRUARY 21, 1980.
C     MODIFIED FEBRUARY 27, 1980.
C
      PARAMETER NALPHA=36, NBETA=90
      PARAMETER NZ=41, ZMAX=1.0
      INTEGER BUFFER(512), ROW(2,512), DALPHA, DBETA
      REAL LONG, LAT, MU, NU, HIST(NALPHA,NBETA), HISTB(NBETA)
      REAL HISTCU(NZ,NZ), SLICE(NZ)
      DOUBLE PRECISION COSOMG, DROLL, DRROLL, DDR, CDR, SDR,
     +     CDPHI, SDPHI
      CHARACTER*8 FILE
      CHARACTER*10 NAMEB, NAMEZ
      DATA NHDR /3/
      DATA DDR /57.295779513082321D0/
      DATA A1, A2, B1, B2 /-180., 180., 0., 90./
      DATA NAMEB /"   BETA   "/
      DATA NAMEZ /"    Z     "/
C
C     READ IN CONSTANTS.
      WRITE (6,1000)
 1000 FORMAT (" ENTER ALL ANGLES IN DEGREES."//
     +     " ENTER LATITUDE, LONGITUDE, DECLINATION, TIME (REAL) ")
      ACCEPT LAT, LONG, DECL, TIME
      WRITE (6,1001)
 1001 FORMAT (" ENTER AIRCRAFT HEADING, ROLL (REAL) ")
      ACCEPT HDG, ROLL
      WRITE (6,1005)
 1005 FORMAT (" ENTER DIRECTION FROM WHICH WIND IS BLOWING,"/
     +     " MEASURED CLOCKWISE FROM NORTH (REAL) ")
      ACCEPT WIND
      WRITE (6,1002)
 1002 FORMAT (" ENTER CAMERA FOCAL LENGTH, PICTURE HEIGHT=WIDTH"/
     +     " (REAL, BOTH IN SAME UNITS) ")
      ACCEPT F, HEIGHT
      WRITE (6,1003)
 1003 FORMAT (" ENTER FIRST & LAST ROW, FIRST & LAST COLUMN",
     +     " (INTEGER) ")
      ACCEPT IR1, IR2, IE1, IE2
      WRITE (6,1004)
 1004 FORMAT (" ENTER IMAGE FILE NAME ")
      ACCEPT FILE
C     INITIALIZE.
      DO 10 J=1,NBETA
        HISTB(J) = 0.
        DO 10 I=1,NALPHA
          HIST(I,J) = 0.
 10   CONTINUE
      DO 15 I=1,NZ
        DO 15 J=1,NZ
          HISTCU(I,J) = 0.
```

```
   15   CONTINUE
        IREC1 = (IR1 + 1) / 2
        ISKIP = NHDR + IREC1 - 1
        SCALE = HEIGHT / 512.
        DALPHA = 360 / NALPHA
        DBETA = 90 / NBETA
        DROLL = ROLL
        DRROLL = DROLL / DDR
        COR = COS (DRROLL)
        SOR = SIN (DRROLL)
        ZDELT = 2. * ZMAX / FLOAT (NZ)
        NCNTR = (NZ + 1) / 2
C       CONVERT F TO SCAN LINE UNITS.
        FF = F
        F = F / SCALE
C       SKIP TO START.
        REWIND 1
        DO 20 I=1,ISKIP
          READ (1)
   20   CONTINUE
C
        WRITE (7,2000) LAT, LONG, DECL, TIME, HDG, ROLL, WIND, FF, HEIGHT
 2000 FORMAT ("1",61X"CONDITIONS"/
     +       "0LATITUDE =",F8.2,10X"LONGITUDE =",F8.2,
     +            10X"DECLINATION =",F7.2,10X"TIME =",F8.2,
     +            " HOURS (GMT)"/
     +       "0AIRCRAFT HEADING =",F8.2,10X"ROLL ANGLE =",F7.2/
     +       "0WIND DIRECTION =",F8.2/
     +       "0CAMERA FOCAL LENGTH =",F4.1,10X"PICTURE HEIGHT =",F4.1)
        WRITE (7,2001) IR1, IR2, IE1, IE2, FILE
 2001 FORMAT ("0ROWS",I4," -",I4,", COLUMNS",I4," -",I4,
     +       " OF IMAGE",A9)
        WRITE (7,2002)
 2002 FORMAT (///60X"ERROR MESSAGES")
C       CALCULATE SOLAR ELEVATION AND AZIMUTH.
        CALL STEP1 (TIME, LONG, LAT, DECL, PHI, CDPHI, SDPHI, AZ, $300)
C       CALCULATE CORRECTION TO REFERENCE ALPHA TO WIND.
        DELTA = 180. - (WIND + AZ)
C       READ A RECORD (2 SCAN LINES).
   30   READ (1) BUFFER
C       UNPACK SCAN LINES.
        CALL GETPIX (BUFFER, ROW)
C
C       START LOOP ON ROWS.
        DO 100 I=IR1,IR2
C       CALCULATE Y.
          Y = 256.5 - FLOAT (I)
C       START LOOP ON COLUMNS.
          DO 70 J=IE1,IE2
C       CALCULATE X.
            X = FLOAT (J) - 256.5
C       CALCULATE MU AND NU.
            CALL STEP2 (X, Y, F, HDG, AZ, MU, NU, COR, SOR, IFLAG)
C
            IF (IFLAG .EQ. 1) GO TO 301
C       CALCULATE COS (OMEGA).
   40       CALL STEP3 (CDPHI, SDPHI, MU, NU, COSOMG)
C       CALCULATE BETA.
            CALL STEP4 (MU, SDPHI, COSOMG, BETA)
C       CALCULATE ALPHA.
            CALL STEP5 (NU, BETA, PHI, CDPHI, COSOMG, MU, ALPHA, $302)
C
   50       IROW = MOD (I - 1, 2) + 1
C       INCREMENT ALPHA-BETA HISTOGRAM CELL.
            CALL STEP6 (ALPHA, BETA, ROW(IROW,J), HIST, NALPHA,
     +           NBETA, DALPHA, DBETA)
C       INCREMENT CROSSWIND-UPWIND HISTOGRAM CELL.
            CALL STEP7 (ALPHA, BETA, DELTA, ROW(IROW,J), HISTCU,
     +           NZ, ZDELT, ZMAX)
C       END LOOP ON COLUMNS.
   70     CONTINUE
```

```
C     READ NEXT RECORD WHEN NECESSARY.
        IF (MOD (I, 2) .EQ. 1) GO TO 100
        READ (1,END=100) BUFFER
C     UNPACK SCAN LINES.
        CALL GETPIX (BUFFER, ROW)
C     END LOOP ON ROWS.
 100  CONTINUE
C     NORMALIZE HISTOGRAMS.
      CALL NRMLZ (HIST, NALPHA, NBETA)
C
      CALL NRMLZ (HISTCU, NZ, NZ)
C     FIND DISTRIBUTION OVER BETA.
      DO 130 J=1,NBETA
        DO 130 I=1,NALPHA
          HISTB(J) = HISTB(J) + HIST(I,J)
 130  CONTINUE
C     FIND MEAN AND VARIANCE OF HISTB.
      CALL MNVAR (HISTB, NBETA, B1, B2, XMEANB, VARB)
C     FIND LARGEST SIGNIFICANT BETA.
      JMAX = 0
      JJ = NBETA + 1
      DO 150 I=1,NALPHA
        DO 140 J=1,NBETA
          K = JJ - J
          IF (HIST(I,K) .LT. 1.0E-5) GO TO 140
          JMAX = MAX (JMAX, K)
          GO TO 150
 140    CONTINUE
        JMAX = MAX (JMAX, 1)
 150  CONTINUE
      B2 = JMAX * DBETA
C     PRINT OUT CONDITIONS AND HISTOGRAMS.
      WRITE (7,2000) LAT, LONG, DECL, TIME, HDG, ROLL, WIND, FF, HEIGHT
      WRITE (7,2001) IR1, IR2, IE1, IE2, FILE
      WRITE (7,2004) AZ
 2004 FORMAT ("0ALPHA IS MEASURED CLOCKWISE FROM AN AXIS THAT ",
     +        "POINTS TOWARD THE SUN."/
     +     10X"SOLAR AZIMUTH =",F7.1," COUNTER-CLOCKWISE FROM SOUTH.")
      WRITE (7,2003)
 2003 FORMAT (///50X"NORMALIZED ALPHA-BETA DISTRIBUTION"//65X"BETA")
      I1 = NALPHA / 2  -  2
      I2 = I1 + 6
      DO 160 I=1,I1
        WRITE (7,2007) (HIST(I,J),J=1,JMAX)
 2007   FORMAT (/(6X18F7.5))
 160  CONTINUE
      WRITE (7,2008) (HIST(I1+1,J),J=1,JMAX)
 2008 FORMAT (3X"A"/(6X18F7.5))
      WRITE (7,2010) (HIST(I1+2,J),J=1,JMAX)
 2010 FORMAT (3X"L"/(6X18F7.5))
      WRITE (7,2011) (HIST(I1+3,J),J=1,JMAX)
 2011 FORMAT (3X"P"/(6X18F7.5))
      WRITE (7,2012) (HIST(I1+4,J),J=1,JMAX)
 2012 FORMAT (3X"H"/(6X18F7.5))
      WRITE (7,2008) (HIST(I1+5,J),J=1,JMAX)
      DO 170 I=I2,NALPHA
        WRITE (7,2007) (HIST(I,J),J=1,JMAX)
 170  CONTINUE
      WRITE (7,2009) A1, A2, DALPHA, B1, B2, DBETA
 2009 FORMAT (//" ALPHA RANGES FROM",F6.0," TO",F6.0,
     +           " IN STEPS OF",I3//
     +    " BETA RANGES FROM",F6.0," TO",F6.0," IN STEPS OF",I3)
      WRITE (7,2005) (HISTB(J),J=1,JMAX)
 2005 FORMAT ("1",55X"DISTRIBUTION OVER BETA"//
     +    (6X18F7.5))
      XSTEP = DBETA
C
      CALL HDSPLY (HISTB, NBETA, NAMEB, B1, XSTEP)
C
      WRITE (7,2006) XMEANB, VARB
 2006 FORMAT (//10X"MEAN =",E15.8,10X"VARIANCE =",E15.8)
```

```
      WRITE (7,2013)
 2013 FORMAT ("1",46X"NORMALIZED CROSSWIND-UPWIND DISTRIBUTION"//
     +     64X"UPWIND")
      I1 = NCNTR - 5
      I2 = NCNTR + 5
      DO 180 I=1,I1
        WRITE (7,2007) (HISTCU(I,J),J=1,NZ)
  180 CONTINUE
      WRITE (7,2014)
 2014 FORMAT (/3X"C")
      WRITE (7,2015) (HISTCU(I1+1,J),J=1,NZ)
 2015 FORMAT ("+",5X18F7.5/(6X18F7.5))
      WRITE (7,2016)
 2016 FORMAT (/3X"R")
      WRITE (7,2015) (HISTCU(I1+2,J),J=1,NZ)
      WRITE (7,2017)
 2017 FORMAT (/3X"O")
      WRITE (7,2015) (HISTCU(I1+3,J),J=1,NZ)
      WRITE (7,2018)
 2018 FORMAT (/3X"S")
      WRITE (7,2015) (HISTCU(I1+4,J),J=1,NZ)
      WRITE (7,2018)
      WRITE (7,2015) (HISTCU(I1+5,J),J=1,NZ)
      WRITE (7,2019)
 2019 FORMAT (/3X"W")
      WRITE (7,2015) (HISTCU(I1+6,J),J=1,NZ)
      WRITE (7,2020)
 2020 FORMAT (/3X"I")
      WRITE (7,2015) (HISTCU(I1+7,J),J=1,NZ)
      WRITE (7,2021)
 2021 FORMAT (/3X"N")
      WRITE (7,2015) (HISTCU(I1+8,J),J=1,NZ)
      WRITE (7,2022)
 2022 FORMAT (/3X"D")
      WRITE (7,2015) (HISTCU(I1+9,J),J=1,NZ)
      DO 190 I=I2,NZ
        WRITE (7,2007) (HISTCU(I,J),J=1,NZ)
  190 CONTINUE
      ZMIN = -ZMAX
      WRITE (7,2023) ZMIN, ZMAX, ZDELT
 2023 FORMAT (//" BOTH ZX AND ZY RANGE FROM",F6.2," TO",F6.2,
     +     " IN STEPS OF",F6.4)
C     CROSSWIND SLICE.
      DO 200 I=1,NZ
        SLICE(I) = HISTCU(I,NCNTR)
  200 CONTINUE
      WRITE (7,2024)
 2024 FORMAT ("1",55X"CROSSWIND DISTRIBUTION"/)
C
      CALL HDSPLY (SLICE, NZ, NAMEZ, ZMIN, ZDELT)
C
      CALL RMSARY (SLICE, NZ, ZMIN, ZDELT, RMS)
C
      WRITE (7,2025) RMS
 2025 FORMAT (//10X"RMS SLOPE",E15.8)
C     UPWIND SLICE.
      DO 210 J=1,NZ
        SLICE(J) = HISTCU(NCNTR,J)
  210 CONTINUE
      WRITE (7,2026)
 2026 FORMAT ("1",56X"UPWIND DISTRIBUTION"/)
C
      CALL HDSPLY (SLICE, NZ, NAMEZ, ZMIN, ZDELT)
C
      CALL RMSARY (SLICE, NZ, ZMIN, ZDELT, RMS)
C
      WRITE (7,2025) RMS
C
      STOP
C
C     ERROR MESSAGES.
```

```
300  WRITE (7,3000) PHI. AZ
3000 FORMAT ("0ELEVATION =",E15.8,10X,
    +      "AZIMUTH CAN'T BE CALCULATED. SET TO",E15.8)
     GO TO 30
301  II = 257 - Y
     JJ = 257 + X
     WRITE (7,3010) II. JJ. MU. NU
3010 FORMAT ("0BAD MU CALCULATION FOR ROW",I4.10X"COLUMN",I4,
    +      10X"MU =",E15.8.10X"NU =",E15.8)
     GO TO 40
302  II = 257 - Y
     JJ = 257 + X
     WRITE (7,3020) II. JJ. NU. BETA. PHI. COSOMG. MU. ALPHA
3020 FORMAT ("0NO SOLUTION FOR ALPHA EXISTS FOR ROW",I4,
    +          10X"COLUMN",I4/
     +      10X"NU =",E15.8,10X"BETA =",E15.8,10X"PHI =",E15.8/
     +      10X"COS (OMEGA) =",D15.8,10X"MU =",E15.8/
     +      10X"ALPHA SET TO",E15.8)
      GO TO 50
C
      END
```

```
$CONTROL MAP,CROSSREF,LABEL
      SUBROUTINE STEP1 (TIME, LONG, LAT, DECL, PHI, CDPHI, SDPHI, AZ, *)
C
C     SUBROUTINE TO CALCULATE LOCAL HOUR ANGLE (USED IN SUBSEQUENT
C          CALCULATIONS), SOLAR ELEVATION, AND SOLAR AZIMUTH.
C
C     MATTHEW LYBANON, CSC, FEBRUARY 6, 1980.
C
C     TIME  = GREENWICH MEAN TIME (MILITARY FORMAT)
C     LONG  = LONGITUDE (+ FOR WEST OF GREENWICH)
C     LAT   = LATITUDE (+ FOR NORTH OF EQUATOR)
C     DECL  = SUN'S DECLINATION (+ FOR NORTH OF EQUATOR)
C     PHI   = SOLAR ELEVATION
C     CDPHI = COS (PHI) -- DOUBLE PRECISION.
C     SDPHI = SIN (PHI) -- DOUBLE PRECISION.
C     AZ    = SOLAR AZIMUTH
C     NOTE:  ALL ANGLES IN ARGUMENT LIST ARE IN DEGREES.
C     *     = ERROR RETURN IF AZIMUTH CAN'T BE CALCULATED
C
      REAL LONG, LAT
      DOUBLE PRECISION DLAT, DRLAT, DDECL, DRDECL, DHA, DRHA,
     +     NUM, DENOM, DDR, DRPHI, CDPHI, SDPHI
      DATA DR, DDR /57.29578, 57.295779513082321D0/
C
C     CALCULATE LOCAL HOUR ANGLE.
      IHR = TIME / 100.
      XMIN = MOD (TIME, 100.)
C     XTIME IS TIME PAST MIDNIGHT IN MINUTES.
      XTIME = FLOAT (60 * IHR) + XMIN
C     HA IS HOUR ANGLE.
      HA = (XTIME - 720.) / 4.  -  LONG
C     CALCULATE SOLAR ELEVATION.
C     CONVERT ANGLES TO DOUBLE PRECISION.
      DLAT = LAT
      DDECL = DECL
      DHA = HA
C     CONVERT ANGLES TO RADIAN MEASURE.
      DRLAT = DLAT / DDR
      DRDECL = DDECL / DDR
      DRHA = DHA / DDR
C
      Z = SIN (DRLAT) * SIN (DRDECL)  +  COS (DRLAT) *
     +      COS (DRDECL) * COS (DRHA)
C     PHI IS SOLAR ELEVATION (RADIAN MEASURE).
      PHI = ASIN (Z)
C     CALCULATE SOLAR AZIMUTH.
C     CONVERT PHI TO DOUBLE PRECISION.
      DRPHI = PHI
      CDPHI = COS (DRPHI)
      SDPHI = SIN (DRPHI)
      NUM = -COS (DRDECL) * SIN (DRHA)
      DENOM = CDPHI
      IF (ABS (NUM) .LE. ABS (DENOM)) GO TO 10
      IF (DENOM .NE. 0.0D0) GO TO 10
C     AZIMUTH CANNOT BE CALCULATED (SUN IS PROBABLY AT ZENITH).
C     SET TO ZERO AND TAKE ERROR EXIT.
      AZ = 0.
      PHI = PHI * DR
      RETURN 1
C     AZIMUTH CAN BE CALCULATED.
 10   Z = NUM / DENOM
C     AZ IS SOLAR AZIMUTH.
      AZ = ASIN (Z)
C     CONVERT PHI AND AZ TO DEGREES.
      PHI = PHI * DR
      AZ = AZ * DR
C
      RETURN
C
      END
```

```
$CONTROL MAP,CROSSREF,LABEL
      SUBROUTINE STEP2 (X, Y, F, HDG, AZ, MU, NU, CDR, SDR, IFLAG)
C
C     SUBROUTINE TO CALCULATE IMAGE ANGULAR COORDINATES.
C     VERSION INCLUDING AIRCRAFT ROLL COMPENSATION.
C
C     MATTHEW LYBANON, CSC, FEBRUARY 6, 1980.
C     MODIFIED FEBRUARY 15, 1980.
C
C     X     = COORDINATE ALONG AXIS POINTING TOWARD STARBOARD WING.
C     Y     = COORDINATE ALONG AXIS POINTING TOWARD AIRCRAFT NOSE.
C     F     = CAMERA LENS FOCAL LENGTH (ASSUMED POSITIVE).
C     NOTE:  X, Y, AND F HAVE THE SAME UNITS.
C     HDG   = AIRCRAFT HEADING.
C     AZ    = SOLAR AZIMUTH.
C     MU    = ANGLE BETWEEN REFLECTED RAY AND VERTICAL.
C     NU    = ANGLE OF IMAGE POINT MEASURED CLOCKWISE FROM AN AXIS
C                 POINTING AWAY FROM SUN.
C     CDR   = COSINE OF AIRCRAFT ROLL ANGLE (DOUBLE PRECISION).
C     SDR   = SINE OF AIRCRAFT ROLL ANGLE (DOUBLE PRECISION).
C     IFLAG = FLAG THAT INDICATES ERROR IN MU CALCULATION
C                   FOR NONZERO ROLL.
C           = 0 -- NO ERROR.
C           = 1 -- ERROR.
C     NOTE:  ALL ANGLES IN ARGUMENT LIST ARE IN DEGREES.
C
      REAL MU, NU, NORM
      DOUBLE PRECISION DRMU, DZ, DX, DY, DF, DDR,
     +     D, NUM, DENOM, CDR, SDR
      DATA DDR /57.295779513082321D0/
C
      IFLAG = 0
C     CONVERT X, Y, AND F TO DOUBLE PRECISION.
      DX = X
      DY = Y
      DF = F
C
      IF (SDR .NE. 0.0D0) GO TO 30
C     ZERO ROLL CALCULATION.
C     CALCULATE MU.
      DZ = SQRT (DX * DX  +  DY * DY) / DF
      DRMU = ATAN (DZ)
C     CONVERT MU TO DEGREES.
      MU = DRMU * DDR
C     CALCULATE NU.
      IF (X .NE. 0.) GO TO 10
C     SPECIAL CASE, CAN'T DIVIDE BY 0.
      IF (Y .GE. 0.) Z = 90.
      IF (Y .LT. 0.) Z = -90.
      GO TO 20
C     NORMAL CALCULATION.
 10   DZ = ATAN (DY, DX)
C     CONVERT Z TO DEGREES.
      Z = DZ * DDR
C
 20   NU = NORM (AZ + HDG + 90. - Z)
C
```

```
      RETURN
C     NONZERO ROLL CALCULATION.
C     CALCULATE MU.
   30 D = CDR  -  DX * SDR / DF
      NUM = DX * DX  +  DY * DY  +
     +          DF *  SDR * (DF * SDR  +  DX * (CDR + D))
      IF (NUM .LT. 0.0D0) IFLAG = 1
      NUM = SQRT (ABS (NUM))
      DENOM = DF * D
      DRMU = ATAN (NUM, DENOM)
C     CONVERT MU TO DEGREES.
      MU = DRMU * DDR
C     CALCULATE NU.
      DENOM = DF * SDR  +  DX * CDR
      IF (DENOM .NE. 0.0D0) GO TO 40
C     SPECIAL CASE.  CAN'T DIVIDE BY 0.
      IF (Y .GE. 0.) Z = 90.
      IF (Y .LT. 0.) Z = -90.
      GO TO 50
C     NORMAL CALCULATION..
   40 DZ = ATAN (DY, DENOM)
C     CONVERT Z TO DEGREES.
      Z = DZ * DDR
C
   50 NU = NORM (AZ + HDG + 90. - Z)
C
      RETURN
C
      END
```

```
$CONTROL MAP,CROSSREF,LABEL
      SUBROUTINE STEP3 (COPHI, SDPHI, MU, NU, COSOMG)
C
C     SUBROUTINE TO CALCULATE COS (OMEGA), WHERE OMEGA IS THE
C          ANGLE OF INCIDENCE OR REFLECTION.
C
C     MATTHEW LYBANON, CSC, FEBRUARY 6, 1980.
C
C     COPHI  = COSINE OF SOLAR ELEVATION -- DOUBLE PRECISION.
C     SDPHI  = SINE OF SOLAR ELEVATION -- DOUBLE PRECISION.
C     MU     = ANGLE BETWEEN REFLECTED RAY AND VERTICAL.
C     NU     = ANGLE OF IMAGE POINT MEASURED CLOCKWISE FROM AN
C                   AXIS THAT POINTS AWAY FROM SUN.
C     NOTE:  ALL ANGLES IN ARGUMENT LIST ARE IN DEGREES.
C     COSOMG = COS (OMEGA) -- DOUBLE PRECISION.
C
      REAL MU, NU
      DOUBLE PRECISION COPHI, SDPHI, DMU, DRMU, DNU, DRNU, DZ,
     +     DDR, COSOMG
      DATA DDR /57.2957795130823210D/
C
C     CONVERT ANGLES TO DOUBLE PRECISION.
      DMU = MU
      DNU = NU
C     CONVERT ANGLES TO RADIAN MEASURE.
      DRMU = DMU / DDR
      DRNU = DNU / DDR
C
      DZ = -COPHI * SIN (DRMU) * COS (DRNU) +
     *     SDPHI * COS (DRMU)
C     MAKE SURE.
      DZ = MAX (DZ, -1.0D0)
      DZ = MIN (DZ, 1.0D0)
      COSOMG = SQRT (0.5D0 * (DZ + 1.0D0))
C
      RETURN
C
      END
```

```
$CONTROL MAP,CROSSREF.LABEL
      SUBROUTINE STEP4 (MU, SDPHI, COSOMG, BETA)
C     SUBROUTINE TO CALCULATE SEA SURFACE TILT.
C
C     MATTHEW LYBANON, CSC, FEBRUARY 6, 1980.
C
C     MU     = ANGLE BETWEEN REFLECTED RAY AND VERTICAL.
C     SDPHI  = SINE OF SOLAR ELEVATION -- DOUBLE PRECISION.
C     COSOMG = COSINE OF ANGLE OF INCIDENCE OR REFLECTION
C                   -- DOUBLE PRECISION.
C     BETA   = SEA SURFACE TILT.
C     NOTE:  ALL ANGLES IN ARGUMENT LIST ARE IN DEGREES.
C
      REAL MU
      DOUBLE PRECISION DMU, DRMU, SDPHI, COSOMG, DDR
      DATA DR, DDR /57.29578, 57.29577951308232100/
C
      IF (COSOMG .NE. 0.0D0) GO TO 10
C     GLANCING INCIDENCE -- NO SULUTION FOR BETA.
C     ASSIGN ARBITRARY VALUE AND RETURN.
      BETA = 0.0
C
      RETURN
C     CONVERT MU TO DOUBLE PRECISION.
 10   DMU = MU
C     CONVERT MU TO RADIAN MEASURE.
      DRMU = DMU / DDR
C
      Z = (COS (DRMU) + SDPHI) / (2.0D0 * COSOMG)
      BETA = ACOS (Z)
C     CONVERT BETA TO DEGREES.
      BETA = BETA * DR
C
      RETURN
C
      END
```

```
$CONTROL MAP,CROSSREF,LABEL
      SUBROUTINE STEPS (NU, BETA, PHI, COPHI, COSOMG, MU, ALPHA, *)
C
C     SUBROUTINE TO CALCULATE LOCAL AZIMUTH OF ASCENT
C           OF SEA SURFACE.
C     MODIFICATION 1. SOLUTION OF KINSMAN'S 2ND GRID EQUATION
C          USING CRITERION TO RESOLVE AMBIGUITY.
C
C     MATTHEW LYBANON, CSC, FEBRUARY 7, 1980.
C     MODIFIED FEBRUARY 13, 1980.
C
C     NU     = ANGLE OF IMAGE POINT MEASURED CLOCKWISE FROM AN
C                       AXIS THAT POINTS AWAY FROM THE SUN.
C     BETA   = SEA SURFACE TILT.
C     PHI    = SOLAR ELEVATION.
C     COPHI  = COS (PHI) -- DOUBLE PRECISION.
C     COSOMG = COSINE OF ANGLE OF INCIDENCE OR REFLECTION
C                       -- DOUBLE PRECISION.
C     MU     = ANGLE BETWEEN REFLECTED RAY AND VERTICAL.
C     ALPHA  = SOLUTION OF SECOND GRID RELATION.
C     NOTE:  ALL ANGLES IN ARGUMENT LIST ARE IN DEGREES.
C     *      = ERROR RETURN IN CASE ALPHA CAN'T BE CALCULATED.
C
      REAL MU, NU
      DOUBLE PRECISION DNU, DRNU, DBETA, DRBETA, COPHI,
     +     COSOMG, DRALP, DTEST, A, B, DZ, NUM, DENOM, DDR,
     +     DMU, DRMU
      DATA DR, DDR /57.29578, 57.2957795130823210D0/
C
C     CONVERT ANGLES TO DOUBLE PRECISION.
      DNU = NU
      DBETA = BETA
      DMU = MU
C     CONVERT ANGLES TO RADIAN MEASURE.
      DRNU = DNU / DDR
      DRBETA = DBETA / DDR
      DRMU = DMU / DDR
C     CHECK FOR SPECIAL CASES.
      IF (COPHI .NE. 0.0D0) GO TO 10
C     SUN IS AT ZENITH, SO ORIENTATION OF COORDINATES IS
C          UNDEFINED.
      ALPHA = NU
C
      RETURN
C
 10   IF (SIN (DRBETA) .NE. 0.0D0) GO TO 20
C     NO WAVE TILT, SO ALPHA IS NOT DEFINED.
      ALPHA = 0.
C
      RETURN
C
 20   IF (COSOMG .NE. 0.0D0) GO TO 30
C     GLANCING INCIDENCE, NO SOLUTION FOR ALPHA.
      ALPHA = 0.
C
      RETURN
C
```

```
   30 IF (SIN (DRMU) .NE. 0.0D0) GO TO 40
C     REFLECTED BEAM IS IN ZENITH DIRECTION.
      ALPHA = 180.
C
      RETURN
C
   40 IF (SIN (DRNU) .NE. 0.0D0) GO TO 70
C     IMAGE POINT IS ON Y-AXIS.
      IF (NU .NE. 180.) GO TO 50
      ALPHA = 180.
C
      RETURN
C     NU = 0.
   50 Z = COSOMG
      OMEGA = ACOS (Z) * DR
      IF (ABS (PHI + OMEGA - BETA) .LE. ABS (PHI + OMEGA + BETA))
     +     GO TO 60
      ALPHA = 180.
C
      RETURN
C
   60 ALPHA = 0.
C
      RETURN
C     CALCULATE ALPHA.
   70 A = 1.D0 / TAN (DRNU)
      B = CDPHI / (2.0 * SIN (DRBETA) * COSOMG)
C     TEST FOR EXISTENCE OF SOLUTION.
      DZ = A * A  + 1.0D0  - B * B
      IF (DZ .GE. 0.0D0) GO TO 80
C     NO SOLUTION.
      ALPHA = ATAN (A * B, -B) * DDR
C
      RETURN 1
C     SOLUTION EXISTS.
   80 DZ = SQRT (DZ)
      DTEST = SIN (DRNU) * (SIN (DRMU)  -  COS (DRNU) * CDPHI)
      NUM = A * B  +  SIGN (DZ, DTEST)
      DENOM = -B  +  A * SIGN (DZ, DTEST)
      DRALP = ATAN (NUM, DENOM)
C     CONVERT ALPHA TO DEGREES.
      ALPHA = DRALP * DDR
C
      RETURN
C
      END
```

```
SCONTROL MAP,CROSSREF.LABEL
      SUBROUTINE STEP6 (ALPHA, BETA, POINT, HIST, NALPHA, NBETA,
     +     DALPHA, DBETA)
C
C     SUBROUTINE TO INCREMENT ALPHA-BETA HISTOGRAM.
C
C     MATTHEW LYBANON, CSC, FEBRUARY 15, 1980.
C
C     ALPHA  = AZIMUTH OF ASCENT OF SEA SURFACE.
C     BETA   = TILT OF SEA SURFACE.
C     POINT  = IMAGE POINT INTENSITY VALUE.
C     HIST   = HISTOGRAM ARRAY.
C     NALPHA = ROW DIMENSION OF HIST.
C     NBETA  = COLUMN DIMENSION OF HIST.
C     DALPHA = WIDTH OF AN ALPHA BIN.
C     DBETA  = WIDTH OF A BETA BIN.
C
      INTEGER POINT, DALPHA, DBETA
      REAL HIST(NALPHA,NBETA)
C
      I = NALPHA  -  IFIX (180. - ALPHA) / DALPHA
      J = NBETA   -  IFIX (90. - BETA) / DBETA
      J = MAX (J, 1)
      HIST(I,J) = HIST(I,J) + POINT
C
      RETURN
C
      END
```

```
$CONTROL MAP,CROSSREF,LABEL
      SUBROUTINE STEP7 (ALPHA, BETA, DELTA, POINT, HIST,
     +     N, ZDELT, ZMAX)
C
C     SUBROUTINE TO INCREMENT CROSSWIND-UPWIND HISTOGRAM.
C
C     MATTHEW LYBANON, CSC, FEBRUARY 26, 1980.
C
C     ALPHA = AZIMUTH OF ASCENT OF SEA SURFACE.
C     BETA  = TILT OF SEA SURFACE.
C     DELTA = CORRECTION TO REFERENCE ALPHA TO (UPWIND)
C                 WIND DIRECTION.
C     POINT = IMAGE POINT INTENSITY VALUE.
C     HIST  = HISTOGRAM ARRAY.
C     N     = ROW AND COLUMN DIMENSION OF HIST (SHOULD BE ODD
C                 SO ORIGIN IS AT CENTER OF A BIN).
C     ZDELT = WIDTH OF A BIN (EITHER AXIS).
C     ZMAX  = UPPER LIMIT FOR EACH AXIS.
C
      INTEGER POINT
      REAL HIST(N,N)
      DATA DR /57.29578/
C
C     CORRECT ALPHA TO WIND DIRECTION.
      AR = ALPHA + DELTA
C     CONVERT ALPHA AND BETA TO RADIANS.
      AR = AR / DR
      BR = BETA / DR
C     CALCULATE WAVE SLOPE.
      SLOPE = TAN (BR)
C     CALCULATE PROJECTIONS ON HISTOGRAM AXES.
      ZX = SLOPE * SIN (AR)
      ZY = SLOPE * COS (AR)
C     CALCULATE INDICES OF HISTOGRAM CELL.
      I = N - IFIX ((ZMAX - ZX) / ZDELT)
      I = MAX (I, 1)
      I = MIN (I, N)
      J = N - IFIX ((ZMAX - ZY) / ZDELT)
      J = MAX (J, 1)
      J = MIN (J, N)
C     INCREMENT HISTOGRAM CELL.
      HIST(I,J) = HIST(I,J) + FLOAT (POINT)
C
      RETURN
C
      END
```

```
$CONTROL MAP, CROSSREF, LABEL
      SUBROUTINE GETPIX (BUFFER, ROW)
C
C     SUBROUTINE TO UNPACK READ BUFFER.
C
C     MATTHEW LYBANON, CSC, FEBRUARY 15, 1980.
C
C     BUFFER = READ BUFFER CONTAINING 2 SCAN LINES, 1 BYTE/PIXEL.
C     ROW    = 2-ROW ARRAY TO HOLD 2 SCAN LINES, 1 WORD/PIXEL.
C
      INTEGER BUFFER(512), ROW(2,512)
C
      IOFSET = 1
      DO 100 IROW=1,2
        DO 50 ICOL=1,512,2
          ROW(IROW,ICOL) = BUFFER((ICOL+IOFSET)/2)[0:8]
          ROW(IROW,ICOL+1) = BUFFER((ICOL+IOFSET)/2)[8:8]
 50     CONTINUE
        IOFSET = 513
 100  CONTINUE
C
      RETURN
C
      END
```

```
$CONTROL MAP,CROSSREF,LABEL
      SUBROUTINE HDSPLY (HIST, N, NAME, X0, XSTEP)
C
C     SUBROUTINE TO DISPLAY A UNIVARIATE HISTOGRAM.
C
C     MATTHEW LYBANON, CSC, FEBRUARY 22, 1980.
C
C     HIST  = HISTOGRAM ARRAY.
C     N     = NUMBER OF ENTRIES IN HIST.
C     NAME  = NAME OF INDEPENDENT VARIABLE.
C     X0    = LOWER END OF VARIABLE RANGE.
C     XSTEP = WIDTH OF A BIN.
C
      REAL HIST(N)
      CHARACTER*10 NAME
      CHARACTER*1 STAR
      DATA STAR /"*"/
C
C     FIND MAXIMUM-OCCUPANCY CELL AND CALCULATE SCALE FACTOR.
      HMAX = HIST(1)
      DO 10 I=2,N
        HMAX = MAX (HMAX, HIST(I))
 10   CONTINUE
      FCTR = 100. / HMAX
C     DISPLAY HISTOGRAM.
      WRITE (7,1000) NAME
 1000 FORMAT ("0",A10," OCCUPANCY"/)
      X = X0
      WRITE (7,1001) X
 1001 FORMAT (1XF9.3)
      DO 40 I=1,N
        NSTAR = FCTR * HIST(I)  +  0.5
        IF (NSTAR .LE. 0) GO TO 20
        WRITE (7,1002) HIST(I), (STAR,K=1,NSTAR)
 1002   FORMAT (11XF9.7,1X100A1)
        GO TO 30
 20     WRITE (7,1002) HIST(I)
 30     X = X + XSTEP
      WRITE (7,1001) X
 40   CONTINUE
C
      RETURN
C
      END
```

```
$CONTROL MAP,CROSSREF,LABEL
      FUNCTION ACOS (X)
C
C     CALCULATES THE INVERSE COSINE OF X.
C
C     MATTHEW LYBANON, CSC, FEBRUARY 11, 1980.
C
      ACOS = 1.570796 - ASIN (X)
C
      RETURN
C
      END
```

```
$CONTROL MAP,CROSSREF,LABEL
      FUNCTION ASIN (X)
C
C     CALCULATES THE INVERSE SINE OF X.
C
C     MATTHEW LYBANON, CSC, FEBRUARY 11, 1980.
C
      IF (ABS (X) .GE. 1.0) GO TO 20
C     REDUCE ARGUMENT RANGE TO [0, 0.5].
      IF (ABS (X) .LE. 0.5) GO TO 10
C     ABS (X) .GT. 0.5 BUT .LT. 1.0.
      Y = (1. - ABS (X)) / 2.
      ASIN = 1.570796  -  2. * ATAN (SQRT (Y / (1. - Y)))
      GO TO 30
C     ABS (X) .LE. 0.5.
 10   ASIN = ATAN (SQRT (X * X / (1.  -  X * X)))
      GO TO 30
C     ABS (X) = 1.0.
 20   ASIN = 1.570796
C     TAKE CARE OF NEGATIVE ARGUMENTS.
 30   IF (X .LT. 0.) ASIN = -ASIN
C
      RETURN
C
      END
```

```
$CONTROL MAP,CROSSREF,LABEL
      SUBROUTINE MNVAR (HIST, N, XMIN, XMAX, XMEAN, VAR)
C
C     SUBROUTINE TO CALCULATE THE MEAN AND VARIANCE OF A
C          NORMALIZED UNIVARIATE DISTRIBUTION.
C
C     MATTHEW LYBANON, CSC, FEBRUARY 21, 1980.
C
C     HIST  = EMPIRICAL DENSITY FUNCTION.
C     N     = NUMBER OF ELEMENTS IN HIST.
C     XMIN  = LOWER END OF VARIABLE RANGE.
C     XMAX  = UPPER END OF VARIABLE RANGE.
C     XMEAN = MEAN OF DISTRIBUTION.
C     VAR   = VARIANCE OF DISTRIBUTION.
C
      REAL HIST(N)
C
      SUM1 = 0.
      SUM2 = 0.
      DX = (XMAX - XMIN) / FLOAT (N)
      X = XMIN  +  0.5 * DX
      DO 10 I=1,N
        SUM1 = SUM1  +  HIST(I) * X
        SUM2 = SUM2  +  HIST(I) * X**2
        X = X + DX
 10   CONTINUE
      XMEAN = SUM1
      VAR = SUM2 - SUM1**2
C
      RETURN
C
      END
```

```
$CONTROL MAP,CROSSREF,LABEL
      REAL FUNCTION NORM (ANGLE)
C
C     TRANSFORMS ANGLE TO THE RANGE (-180, 180).
C
C     MATTHEW LYBANON, CSC, FEBRUARY 7, 1980.
C
      NORM = ANGLE
 10   IF (NORM .LE. 180.0) GO TO 20
      NORM = NORM - 360.0
      GO TO 10
C
 20   IF (NORM .GT. -180.0) RETURN
C
      NORM = NORM + 360.0
      GO TO 20
C
      END
```

```
$CONTROL MAP,CROSSREF,LABEL
      SUBROUTINE NRMLZ (HIST, NI, NJ)
C
C     SUBROUTINE TO NORMALIZE A 2-DIMENSIONAL HISTOGRAM.
C
C     MATTHEW LYBANON, CSC, FEBRUARY 26, 1980.
C
C     HIST = HISTOGRAM ARRAY.
C     NI   = ROW DIMENSION OF HIST.
C     NJ   = COLUMN DIMENSION OF HIST.
C
      REAL HIST(NI,NJ)
C
      SUM = 0.
      DO 10 I=1,NI
        DO 10 J=1,NJ
          SUM = SUM + HIST(I,J)
 10   CONTINUE
      DO 20 I=1,NI
        DO 20 J=1,NJ
          HIST(I,J) = HIST(I,J) / SUM
 20   CONTINUE
C
      RETURN
C
      END
```

```
SCONTROL MAP.CROSSREF.LABEL
      SUBROUTINE RMSARY (ARRAY, N, XMIN, XDELT, ARMS)
C
C     SUBROUTINE TO FIND THE RMS VALUE OF AN ARRAY.
C
C     MATTHEW LYBANON, CSC, FEBRUARY 26, 1980.
C
C     ARRAY = UNIVARIATE DISTRIBUTION (MAY NOT BE NORMALIZED).
C     N     = NUMBER OF ELEMENTS IN ARRAY.
C     XMIN  = LOWER END OF VARIABLE RANGE.
C     XDELT = STEP SIZE.
C     ARMS  = RMS VALUE OF ARRAY.
C
      REAL ARRAY(N)
C
      D = ARRAY(1)
      X = XMIN  +  0.5 * XDELT
      SUM = ARRAY(1) * X**2
      DO 10 I=2,N
        D = D + ARRAY(I)
        X = X + XDELT
        SUM = SUM  +  ARRAY(I) * X**2
 10   CONTINUE
C
      ARMS = SQRT (SUM / D)
C
      RETURN
C
      END
```

## SAMPLE RUN STREAM

```
!JOB GLINT.C335
!FILE FTN07;DEV=LP;CCTL
!FILE FTN01=I0059711.JO133524.S101,OLD
!RUN GLITTER2
30.84,71.92,6.66,1606.39
0.,0.
100.
3.,4.5
1,512,1,512
I0059711
!FILE FTN01=I0059582.JO133524.S101,OLD
!RUN GLITTER2
30.34,72.00,6.66,1704.44
90.,-23.66
100.
3.,4.5
1,512,1,512
I0059582
!FILE FTN01=I0059614.JO133524.S101,OLD
!RUN GLITTER2
30.34,72.00,6.66,1704.46
90.,-23.66
100.
3.,4.5
1,512,1,512
I0059614
!FILE FTN01=I0059631.JO133524.S101,OLD
!RUN GLITTER2
30.34,72.00,6.66,1704.49
90.,-23.66
100.
3.,4.5
1,512,1,512
I0059631
!EOJ
```

# 5. References

1. Cox, C. and W. Munk, Statistics of the Sea Surface Derived from Sun Glitter, *Journal of Marine Research* 13, pp. 198-227, 1954.

2. Cox, C. and W. Munk, Measurement of the Roughness of the Sea Surface from Photographs of the Sun's Glitter, *Journal of the Optical Society of America* 44, pp. 838-850, 1954.

3. Kinsman, B., *Wind Waves*, Prentice-Hall, Inc., 1965.

4. Craven, P. D., R. V. Hembree, and C. A. Lundquist, Photographic Photometry from Skylab, in *Proceedings of Comet Kohoutek Workshop*(G. A. Gary, ed.), Marshall Space Flight Center, Huntsville, Alabama, June 13-14, 1974, NASA SP-355, 1975.

5. Rossi, B., *Optics*, Addison-Wesley Publishing Company, Inc., 1957.

6. Hill, J. C. et al., *Dutton's Navigation and Piloting*, United States Naval Institute, 1958.

7. Abramowitz, M. and I. A. Stegun, *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*, National Bureau of Standards Applied Mathematics Series No. 55, Ninth Printing, 1970.

8. Sturges, H. A., Choice of a Class Interval, *American Statistical Association Journal* 21, p. 66, 1926.

9. Sperry UNIVAC 1100 Series FORTRAN V Library, UP-7876, Rev. 3, 1974.

10. Mees, C. E. K. and T. H. James, *The Theory of the Photographic Process*, Third Edition, Macmillan, New York, 1966.

# Appendix A: Solution for local azimuth of sea-surface ascent

## A1. Introduction

This appendix discusses the solution for $\alpha$, the local azimuth of sea-surface ascent. The equation for $\alpha$ is sufficiently complex that the properties of the solution are not obvious. One property that will be proved is that (when there is a solution at all) there are two roots in general. This poses a problem in solving for the $(\alpha, \beta)$ that corresponds to each pair of values in the film plane $(\mu, \nu)$. It is to be expected that further consideration of the physics of the situation will be required to resolve the dilemma.

## A2. Analysis

The azimuth angle $\alpha$ is a solution of the equation

$$\cot\nu = \cot\alpha + \tfrac{1}{2} \csc\alpha \csc\beta \sec\omega \cos\phi, \tag{A1}$$

where $\nu$ gives the angle that the image point makes with the film plane y-axis, $\beta$ is the inclination of the sea surface, $\omega$ is the angle of incidence or reflection, and $\phi$ is the solar elevation. The sign of the last term is the opposite of that given in Reference 2 (second of Equations (2)), but a careful check of the derivation gives the result shown above.

Equation (A1) can be simplified by multiplying through by $\sin\alpha$. This gives

$$\sin\alpha \cot\nu = \cos\alpha + \tfrac{1}{2} \csc\beta \sec\omega \cos\phi, \tag{A2}$$

which can be written

$$A \sin\alpha = \cos\alpha + B, \tag{A3}$$

where

$$A = \cot\nu, \tag{A4a}$$

$$B = \tfrac{1}{2} \csc\beta \sec\omega \cos\phi. \tag{A4b}$$

The properties of the solution of Equation (A3) can be derived by writing the equation in a different form. Make the change of variables

$$x = \cos\alpha \tag{A5a}$$

$$y = \sin\alpha \tag{A5b}$$

With these substitutions Equation (A3) becomes

$$y = x/A + B/A \tag{A6}$$

which is in the form

$$y = m\,x + b \tag{A7}$$

with

$$m = 1/A \tag{A8a}$$

$$b = B/A \tag{A8b}$$

From Equations (A5a) and (A5b) it can be seen that the solutions for $\alpha$ given by Equation (A7) (if any) are its intercepts with the unit circle. This is illustrated in Figure A1. The value of $\alpha$ is $\alpha = \tan^{-1} (y/x)$, where $(x, y)$ are the coordinates of an intercept.

Two special cases will be considered first:

- $m = 0$. Then $y = b$ and $x = \pm(1 - b^2)^{1/2}$ if $b \leq 1$.
- $b = 0$. Then $y = m\,x$ passes through the origin, so

$$x^2 + y^2 = x^2 + (m\,x)^2 = 1$$

$$x = \pm 1/(1 + m^2)^{1/2},\ y = m\,x = \pm m/(1 + m^2)^{1/2}$$

In both of these cases (if $b \leq 1$) there are two solutions.

More generally, suppose $m \neq 0$, $b \neq 0$. Draw a perpendicular from the origin O to $y = m\,x + b$, intersecting the straight line at $P = (x_0, y_0)$. The perpendicular line has the equation

$$y = (-1/m)\,x. \tag{A9}$$

First, let us find the distance $d = OP$.

$$m\,x_0 + b = (-1/m)\,x_0$$

$$x_0 = -b\,m/(m^2 + 1) \tag{A10a}$$

$$y_0 = (-1/m)\,[-b\,m/(m^2 + 1)]$$

$$= b/(m^2 + 1) \tag{A10b}$$

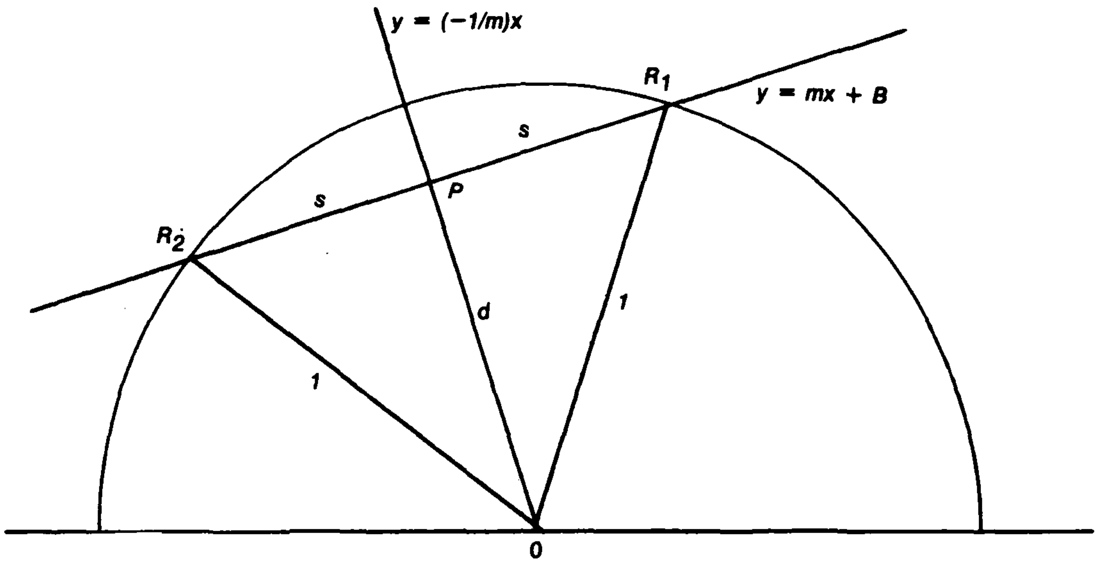

*Figure A1. Equation (A7) displayed on the unit circle.*

$$d = (x_0^2 + y_0^2)^{1/2} = [(b^2 m^2 + b^2) / (m^2 + 1)^2]^{1/2}$$

$$= b/(m^2 + 1)^{1/2} \qquad \text{(A10c)}$$

There are three possibilities:

- If $d > 1$, then Equation (A7) has no solution.
- If $d = 1$, there is exactly one solution, $(x_0, y_0)$.
- If $d < 1$, there are two solutions.

In terms of A and B, $d$ is given by

$$d = B/(1 + A^2)^{1/2} \qquad \text{(A11a)}$$

so the three possibilities above can be expressed in terms of

$$f(A, B) = A^2 + 1 - B^2. \qquad \text{(A11b)}$$

Equation (A7) has 0, 1, or 2 solutions, depending on whether $f(A, B)$ is negative, zero, or positive, respectively.

When $f(A, B) > 0$ the two solutions $R_1 = (x_1, y_1)$ and $R_2 = (x_2, y_2)$ are found by going a distance $s = (1 - d^2)^{1/2}$ from point $P$ in both directions along $y = m\,x + b$. Equivalently, they can be found by applying the identity $\sin^2\alpha + \cos^2\alpha = 1$ to Equation (A3). The result is

$$x_1 = \frac{-B + A(A^2 + 1 - B^2)^{1/2}}{A^2 + 1} \qquad \text{(A12a),}$$

$$y_1 = \frac{A\,B + (A^2 + 1 - B^2)^{1/2}}{A^2 + 1} \qquad \text{(A12b),}$$

$$\alpha_1 = \tan^{-1}\left[\frac{y_1}{x_1}\right] = \tan^{-1}\left[\frac{A\,B + (A^2+1-B^2)^{1/2}}{-B + A(A^2+1-B^2)^{1/2}}\right] \qquad \text{(A12c),}$$

$$x_2 = \frac{-B - A(A^2 + 1 - B^2)^{1/2}}{A^2 + 1} \qquad \text{(A13a),}$$

$$y_2 = \frac{A\,B - (A^2 + 1 - B^2)^{1/2}}{A^2 + 1} \qquad \text{(A13b),}$$

$$\alpha_2 = \tan^{-1}\left[\frac{y_2}{x_2}\right] = \tan^{-1}\left[\frac{A\,B - (A^2+1-B^2)^{1/2}}{-B - A(A^2+1-B^2)^{1/2}}\right]. \qquad \text{(A13c)}$$

The occurrence of $f(A, B)$ in these equations should be noted. When $f(A, B) = 0$ only the first terms in both the numerator and denominator of Equations (A12c) and (A13c) remain, $\alpha_1 = \alpha_2$, and the solution for possibility 2 (discussed after Equation (A10c)) is obtained.

It can be shown that $\alpha_1$ and $\alpha_2$ do indeed satisfy Equation (A1). Written in terms of $A$ and $B$, Equation (A1) becomes

$$A = \cot\alpha + B \csc\alpha \ . \qquad \text{(A14)}$$

By referring to a unit circle diagram it can be seen that

$$\cot\alpha_i = \frac{x_i}{y_i} = \frac{-B \pm A(A^2 + 1 - B^2)^{1/2}}{A\ B \pm (A^2 + 1 - B^2)^{1/2}} \qquad \text{(A15a)}$$

$$\csc\alpha_i = \frac{1}{y_i} = \frac{A^2 + 1}{A\ B \pm (A^2 + 1 - B^2)^{1/2}} \qquad \text{(A15b)}$$

where $i = 1$ or $2$ and the variable signs should be paired, + with + and − with −. When we substitute (A15a) and (A15b) into (A14) we get

$$A = \frac{-B \pm A(A^2+1-B^2)^{1/2}}{A\ B \pm (A^2+1-B^2)^{1/2}} + \frac{B\,(A^2 + 1)}{A\ B \pm (A^2+1-B^2)^{1/2}}$$

$$= \frac{A\,[A\ B \pm (A^2 + 1 - B^2)^{1/2}]}{A\ B \pm (A^2 + 1 - B^2)^{1/2}}$$

$$= A \ . \qquad \text{(A16)}$$

# A3. Discussion

There are several special cases for which the solution to Equation (A1) cannot be performed as discussed in the preceding section. These cases will be taken up now.

## A3.1 $\cos\phi = 0$

$B = 0$ so,
$A \sin\alpha = \cos\alpha$,
$\tan\alpha = 1/A = 1/\cot\nu = \tan\nu$ . (A17)

Physically, $\cos\phi = 0$ means that the sun is at the zenith, so the direction of the $X_2$ and y axes is undefined. Arbitrarily we may pick $\alpha = \nu$ in this case, but it is not expected to occur for the present application.

## A3.2 $\sin\alpha = 0$

Since $\alpha$ is the desired solution this situation is not explicitly apparent in advance. When $\sin\alpha = 0$ then $\cot\alpha$ and $\csc\alpha$ are undefined (infinite) and Equation (A1) is invalid. Returning to the derivation of equation (A1), it is seen that

$$\sin\nu \sin\mu = 0. \qquad \text{(A18)}$$

Physically, it is clear that when $\sin\alpha = 0$ the reflected beam must strike the image plane along the y-axis, $\sin\nu = 0$. The particular situation $\sin\mu = 0$ occurs when the reflected beam is in the zenith direction. Further examination shows that

If $\nu = 0$ and $\phi + \omega - \beta = 0$, then $\alpha = 0°$.
If $\nu = 0$ and $\phi + \omega + \beta = 0$, then $\alpha = 180°$.
If $\nu = 180°$, then $\alpha = 180°$ always.

The cases $\sin\nu = 0$ and $\sin\mu = 0$ will be discussed further below.

## A3.3 $\sin\beta = 0$

$B$ is defined. Physically, $\sin\beta = 0$ means there is no wave slope, so clearly $\alpha$ has no meaning. An arbitrary value such as 0° can be assigned to $\alpha$ in this case.

## A3.4 $\cos\omega = 0$

There is no solution for $\alpha$ from the original geometric relations; also, $B$ is undefined. Physically, $\cos\omega = 0$ corresponds to glancing incidence. An arbitrary value such as 0° can be assigned to $\alpha$ in this case.

## A3.5 $\sin\nu = 0$

$A$ is undefined. The image point is on the film y-axis, which can only happen when $\alpha = 0°$ or $\alpha = 180°$. Refer to Section A3.2.

## A3.6 $\sin\mu = 0$

The value of $\mu$ does not appear explicitly in Equation (A1), so presumably the solution presented in Section A2 could be used. But it was shown above that $\sin\mu = 0$ when $\sin\alpha = 0$, in which case Equation (A1) is not valid. From the basic geometrical relationships, when $\sin\mu = 0$,

$$2 \sin\alpha \sin\beta \cos\omega = 0. \qquad \text{(A19)}$$

The cases $\sin\beta = 0$ and $\cos\omega = 0$ have already been discussed. These possibilities should be checked for first. If $\sin\beta \neq 0$ and $\cos\omega \neq 0$ when $\sin\mu = 0$, then $\sin\alpha = 0$. Physically, $\sin\mu = 0$ occurs when the reflected beam is in the zenith direction. This can only happen when $\alpha = 180°$.

# Conclusion

It has been shown that, in general, there are two solutions for $\alpha$. Both solutions are mathematically valid, so the choice of solution in a particular case must be based on further analysis of the physics underlying the derivation of Equation (A1). It is not immediately clear how to proceed.

The cases $\alpha = 90^\circ$ and $\alpha = -90^\circ$ provide an interesting contrast. Both $cot\alpha$ and $csc\alpha$ are defined, so Equation (A1) is valid. First, consider $\alpha = 90^\circ$. Then, returning to the notation of Equations (A12a) – (A13c), $x_i = 0$ and $y_i = 1$, where $i = 1$ or $2$.

$$y_i = sin\alpha_i = 1 = \frac{A\,B \pm (A^2 + 1 - B^2)^{1/2}}{A^2 + 1} \tag{A20a}$$

$$A^2 - A\,B + 1 = \pm(A^2 + 1 - B^2)^{1/2}$$

$$(A^2 - A\,B + 1)^2 = A^2 + 1 - B^2$$

$$(A - B)^2\,(A^2 + 1) = 0\,, \tag{A20b}$$

after performing the algebra. So $A = B$. (The physical condition $\nu = 45^\circ$, $\beta = \omega = \phi = 30^\circ$ gives $A = B = 1$, an example of this situation.) When $A = B$ is substituted into Equation (A20a) it is seen that only $\alpha_1$ (Eq. (A12c)) gives the correct result. A similar development shows that for $\alpha = -90^\circ$ only $\alpha_2$ (Eq. (A13c)) satisfies the equation comparable to Equation (A20a). So in some cases $\alpha_1$ is the correct solution and in others it is $\alpha_2$.

# Appendix B: Computer program improvements

## B1. Introduction

Some changes have been made to the system of computer programs described in the text of this report, for the purpose of improving the results. This appendix describes two modifications. The first of them is a procedure that provides a better approximation to the original light intensities incident on the film than simply the digitized pixel values themselves. The second modification is a revised calculation of rms slope values from the $Z_x$-$Z_y$ histogram. The new formulations produce some changes both in the programs and in the instructions for use.

## B2. Enhancements to formulation

### B2.1 Correction of digitized values to obtain original luminance

The pixel values that make up the digitized images are related to the reflected brightness field that illuminated the film. It is the latter (reflected brightness) that should properly be used to construct the distributions that are analyzed to provide information on the sea slope distributions. The former (pixel values) is what is actually available. It is almost certain that the latter values are not directly proportional to the former, as the following discussion shows.

An image is produced when light of intensity $I_0(x, y)$ strikes a photographic plate, where the coordinates $x$ and $y$ define the position on the plate. The illumination produces an optical density distribution $D(x, y)$ when the resulting negative is developed [10]. The digitized image is produced by illuminating the negative by an (ideally) uniform light intensity $I_1$. The intensity of the light that is transmitted is given by

$$I_2 (x, y) = I_1 \, 10^{-D(x, y)} \tag{B1}$$

In the NORDA Remote Sensing Branch IDSIPS system the intensity distribution $I_2 (x, y)$ is imaged by a video camera, whose output is digitized to give values $K$ at sampled positions, in the range $0 \leqslant K \leqslant 255$. The functional relationship between $I_2$ and $K$ is unknown. The values $K$, which describe a negative, are further modified to give the pixel values $K'$ for a positive image by

$$K' = 255 - K. \tag{B2}$$

To perform the sun glitter analysis it is necessary to estimate $I_o$ from $K'$. Empirically, it is found that

$$10^{-D} = X = a + bK + cK^2 \tag{B3}$$

describes the relationship between $K$ and $D$ fairly well. The Hurter-Driffield curve that describes the dependence of $D$ on exposure for photographic film ($D - \log E$ curve) has a linear region given by

$$D = \gamma \, (\log I_0 t - \log i) \tag{B4}$$

where $\gamma$ is the slope of the linear part, $t$ is the exposure time, and $i$ is a constant called the "inertia" of the film [10].

If we assume that exposures are restricted to the linear portion of the $D - \log E$ curve, we can estimate $I_o$ from $K'$ as follows:

- Invert Equation (B2) to obtain $K$.
- Use Equation (B3) to get $X = 10^{-D}$ from $K$.
- From Equation (B4),

$$10^{-D} = X = AI_0^{-\gamma} . \tag{B5}$$

Therefore,

$$I_o = B/X^{1/\gamma} . \tag{B6}$$

The choice of $B$ is tantamount to the choice of a unit of light intensity, so $B$ can be chosen for convenience.

The only unknowns in the above procedure are the three coefficients in Equation (B3), a, b, and c. Table B1 lists a set of calibration values that were used to obtain the coefficients. A least-squares fit to the data of Table B1 gave

$a = 0.10138 \times 10^{-1}$
$b = 0.97295 \times 10^{-3}$
$c = 0.21485 \times 10^{-4}$

Table B1. Step wedge calibration values.

| Step | D | $10^{-D}$ | K |
|---|---|---|---|
| 1 | 0 | 1 | 186 |
| 2 | 0.1 | 0.794 | 172 |
| 3 | 0.2 | 0.631 | 159 |
| 4 | 0.38 | 0.417 | 117 |
| 5 | 0.59 | 0.257 | 81 |
| 6 | 0.83 | 0.148 | 53 |
| 7 | 1.04 | 0.0912 | 39 |
| 8 | 1.28 | 0.0525 | 29 |
| 9 | 1.50 | 0.0316 | 21 |
| 10 | 2.27 | 0.00537 | 12 |

D = Density of step
K = Resulting digitized value

### B2.2 RMS slope calculation

In the computer program described in the text of this report, rms slope values were calculated in the following way: Two "slices" of the distribution of brightness over components of slope were taken, one along the upwind = downwind axis and the other along the cross-wind axis. Each slice was taken to be a separate univariate distribution, from which the rms slopes (upwind and cross-wind) were calculated.

To state the above formally, let the bivariate distribution be called $p(x, y)$. Since $p(x, y)$ is a probability density,

$$\iint p(x, y)\, dx\, dy = 1 \tag{B7}$$

where the integral is over the appropriate region of $R_2$. (There is an equivalent formulation for discrete distributions.) The two slices are $Ap(x, 0)$ and $B\,p(0, y)$ where $A$ and $B$ are chosen to normalize the integral over $x$ or $y$, respectively. The rms values were calculated as the square roots of

$$A \int (x - m)^2\, p(x, 0)\, dx \tag{B8}$$

and the corresponding integral over the other slice. In Equation (B8) $m$ is the mean value of $x$. (In fact, $m = 0$ was assumed in SUBROUTINE RMSARY.)

From usual statistical theory, it might be assumed that Equation (B8) should be replaced by

$$\iint (x - m)^2\, p(x, y)\, dx\, dy \tag{B9}$$

(and the equivalent for the $y$ component). The $m$ in Equation (B9) is not necessarily the same as the $m$ in Equation (B8). The discussion by Cox and Munk does not make it clear which formulation is correct [1, 2]. Furthermore, for some distributions (such as the bivariate normal distribution with no correlation between x and y) the results are no different. However, for completeness the formulation of Equation (B9) was implemented and tried.

## B3. Changes to programs and user instructions

### B3.1 Original intensity calculations

Equations (B2), (B3), and (B6) were implemented in a new subroutine, INTENS. The main program was modified to a new version, GMAIN3, which calls INTENS and contains the other necessary changes. INTENS was compiled into the same RL, GSUBSRL, as the other subroutines. GMAIN3 was compiled into GMAIN3RB. The prepared program has the file name GLITTER3.

GLITTER3 requires the same :FILE statements as GLITTER2. The only change is that one extra input item, $\gamma$, is needed. It goes in a separate record immediately following focal length and picture height.

### B3.2 Revised RMS slope calculation

The changes for the calculation described in Section B2.2 are in a new version of the main program, GMAIN4. GMAIN4 is a revision of GMAIN3. Like the latter it calls INTENS, and $\gamma$ is a required input. GMAIN4 was compiled into GMAIN4RB. The prepared program has the file name GLITTER4. (RMSARY is no longer used.) The instructions for use are exactly the same as for GLITTER3.

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

AD - A161 995

## REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION
Unclassified

1b RESTRICTIVE MARKINGS
None

2a. SECURITY CLASSIFICATION AUTHORITY

2b. DECLASSIFICATION/DOWNGRADING SCHEDULE

3. DISTRIBUTION/AVAILABILITY OF REPORT
Approved for public release; distribution is unlimited.

4. PERFORMING ORGANIZATION REPORT NUMBER(S)
NORDA Report 103

5. MONITORING ORGANIZATION REPORT NUMBER(S)
NORDA Report 103

6. NAME OF PERFORMING ORGANIZATION
Naval Ocean Research and Development Activity

7a. NAME OF MONITORING ORGANIZATION
Naval Ocean Research and Development Activity

6c. ADDRESS *(City, State, and ZIP Code)*
Ocean Science Directorate
NSTL, Mississippi 39529-5004

7b. ADDRESS *(City, State, and ZIP Code)*
Ocean Science Directorate
NSTL, Mississippi 39529-5004

8a. NAME OF FUNDING/SPONSORING ORGANIZATION
Naval Ocean Research and Development Activity

8b. OFFICE SYMBOL *(If applicable)*

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER

8c. ADDRESS *(City, State, and ZIP Code)*
Ocean Science Directorate
NSTL, Mississippi 39529-5004

10. SOURCE OF FUNDING NOS.

| PROGRAM ELEMENT NO. | PROJECT NO. | TASK NO. | WORK UNIT NO. |
|---|---|---|---|
| | | | |

11. TITLE *(Include Security Classification)*
Ocean Wave Slope Statistics from Automated Analysis of Sun Glitter Photographs

12. PERSONAL AUTHOR(S)
Matthew Lybanon

13a. TYPE OF REPORT
Final

13b. TIME COVERED
From ________ To ________

14. DATE OF REPORT *(Yr., Mo., Day)*
June 1985

15. PAGE COUNT
63

16. SUPPLEMENTARY NOTATION

17. COSATI CODES

| FIELD | GROUP | SUB. GR. |
|---|---|---|
| | | |
| | | |

18. SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*
sun glitter, wave slope, digital image processing

19 ABSTRACT *(Continue on reverse if necessary and identify by block number)*

The image of the sun reflected from the rough surface of the sea forms a diffuse pattern, whose details depend on the nature of the surface waves and swell. Consequently, statistics of the slope distribution of the sea surface are related to the statistics of the sun's glitter on the sea surface. Aerial photographs are a convenient medium for recording the glitter pattern. The mathematical relationship to derive the sea surface slope statistics can be determined from an analysis of the imaging geometry. However, analysis of the photographs can be a labor-intensive procedure.

The problem was first studied over thirty years ago. Now, there are modern digital image processing systems and techniques that greatly increase the practicality of the analysis. This report derives the relevant equations and describes an implementation on the Interactive Digital Satellite Image Processing System (IDSIPS). The IDSIPS system is operated by the Remote Sensing Branch of the Naval Ocean Research and Development Activity (NORDA). The report includes full formal documentation of the computer software.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT
UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ■ DTIC USERS ☐

21. ABSTRACT SECURITY CLASSIFICATION
Unclassified

22a. NAME OF RESPONSIBLE INDIVIDUAL
Matthew Lybanon

22b. TELEPHONE NUMBER *(Include Area Code)*
(601) 688-5263

22c. OFFICE SYMBOL
Code 321

DD FORM 1473, 83 APR

EDITION OF 1 JAN 73 IS OBSOLETE.

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE