File Copy

Declass Review by NIMA/DOD

STAT

Approved For Release 2003/01/28 : CIA-RDP78B05171A000600030001-0

STEREOCOMPARATOR

OPERATOR TRAINING MANUAL

(Preliminary Submittal)

U. S. Government

STAT

STAT

This document is presented for review by the customer in the form of a Preliminary Operator Training Manual

STAT

STAT

TABLE OF CONTENTS

OPERATOR TRAINING MANUAL

I. VECTORS, TENSORS, and MATRICES

1.1 Vectors*

1.1.1 Need for Coordinate Systems

Vectors have often been defined as entities which have both magnitude and direction. In exploring the concept direction, one finds that it cannot be defined except in relation to reference points or reference directions. Any particular statement or rule which establishes a given set of reference directions is said to thus establish a coordinate system. Thus the above definition of vectors is made mathematically meaningful by relating the magnitude and direction to some particular coordinate system. In any established space, however, a given set of reference points may be used to define a large number of different coordinate systems, any one of which may be suitable for expressing the desired vector. Thus a complete definition of a vector must provide a scheme for expressing the vector with respect to any desired coordinate system.

1.1.2 Properties of Vector Addition

Vector analysis includes a rule for vector addition, and this rule is purposely designed so that vector addition is both "commutative" and "associative." This means that if two vectors A and B are added then the resultant is the same whether the vector addition is performed as

$$A + B \quad \text{(vector addition)}$$

or as

$$B + A \quad \text{(vector addition)}$$

*This discussion is not intended as an introduction to vector analysis. It is written for readers who are already, at least somewhat, familiar with most of the ideas presented.

Furthermore if three vectors A, B, and C are added, then the resultant is the same whether any of the following schemes is employed:

$$A + B + C,$$

$$(A + B) + C,$$

$$A + (B + C),$$

$$A + (C + B),$$

$$(C + A) + B, \text{ etc.}$$

1.1.3 Use of Vector Components

In implementing the ideas presented in the foregoing two paragraphs one arrives at the notion of resolving vectors into other vectors such that the latter may be vectorially added to produce the original vectors as resultants. Thus assume that a coordinate system x, y, z is to be used for the vector addition of two vectors A and B. The vector A is first resolved into the three component vectors A_x, A_y, and A_z such that vectorial addition of these component vectors yields the vector A as resultant. Similarly the vector B is resolved into component vectors B_x, B_y, and B_z which vectorially added produce the vector B as resultant. The above stated properties of vectorial addition imply that the vector sum of the vectors A and B must be the same as the vector sum of resultants obtained by vectorially adding corresponding components. The rule of vector addition is also such that the vector sum of two vectors which have the same direction is a vector with this common direction whose magnitude is the algebraic sum of the two magnitudes. Thus the vector sum of the vectors A and B is the same as the vector sum of the three component vectors

$$A_x + B_x, \; A_y + B_y, \text{ and } A_z + B_z$$

where each component is obtained by algebraic addition of magnitudes. Thus one way of implementing vector addition is by resolving the vectors into components with respect to some established coordinate system and defining the vector sum as that vector whose components are the algebraic sums of corresponding components of the vectors being added. This turns out to be a practical scheme for computing vector sums.

1.1.4 Arbitrariness of Coordinate Systems

From all of the foregoing it follows that there is considerable arbitrariness in selecting a coordinate system but that any vector sum must be, in some sense, independent of which particular coordinate system is chosen. Thus if two coordinate systems x, y, z and u, v, w are equally suitable, then the vector sum of A and B is the vector whose components in the two coordinate systems are

$$A_x + B_x, \ A_y + B_y, \ A_z + B_z$$

and

$$A_u + B_u, \ A_v + B_v, \ A_w + B_w$$

respectively. Vector addition of these two sets of component vectors must therefore yield resultants which are equal in magniture and equivalent in direction. The concepts "magnitude" and "direction" must then be so related to any set of components as to have equal (or at least equivalent) values regardless of which suitable coordinate system is being employed.

If the "suitable" coordinate systems were to be limited to only those which are called Cartesian coordinate systems (i.e., systems with 3 mutually perpendicular axes with <u>fixed</u> directions) then magnitude might be defined by the Pythagorean rule:

$$|V| = \sqrt{V_x^{\ 2} + V_y^{\ 2} + V_z^{\ 2}} = \sqrt{V_u^{\ 2} + V_v^{\ 2} + V_w^{\ 2}}$$

where V is any vector (including those obtained by adding other vectors). It is desirable, however, to include more general co-ordinate systems as being "suitable" (for example, spherical coordinates, which may consist of the distance from earth's center, longitude, and geocentric latitude, respectively). Thus the definition of "magnitude" must be a generalized version of the Pythagorean relation which is suitable for the most general coordinate system to be employed (a corresponding discussion of "direction" will be omitted).

1.1.5 Examples of "Directed Strokes" Which Are Not Vectors

Before continuing, some examples will be given of entities which have magnitude and direction but which do not conform to the mathematical relations which are customarily employed in vector analysis. (The point is that, in actual usage, the term "vector" is properly used to designate only a certain subclass of all possible entities which have magnitude and direction). Suppose there are three non-coplanar vectors A, B, and C which are so established as to have their tails all conjoined in one point. Then one may draw three directed strokes to represent the "vector angles" between the three pairs of vectors.

Thus the directed stroke α has a direction perpendicular to both of the vectors A and B and a magnitude equal to the angle between these two vectors (this angle being in the plane of the two vectors).

Similarly the directed stroke β has a direction perpendicular to the two vectors B and C and a magnitude equal to the angle between these two vectors. Finally the directed stroke γ has a direction perpendicular to the two vectors A and C and a magnitude equal to the angle between these two vectors. With these definitions it may be demonstrated that the directed stroke γ is, in general, not equal to the vector sum of the directed strokes α and β. Thus it is not strictly proper to call the directed strokes α, β and γ by the name "vectors."

Somewhat similarly, directed strokes which join various points in space behave like "true" vectors only if the allowable coordinate systems are restricted to those in the same class as Cartesian coordinate systems. As was noted previously, this is not, in general, a desirable restriction.

1.1.6 Basic Importance of 'Components' in Defining Vectors

Evidently then it is not entirely satisfactory to begin a deductive discussion of vectors by defining them as entities having magnitude and direction (although the concept of the directed stroke is, nevertheless, extremely useful in qualitative visualization of vectors). Present day general discussions of vector analysis are often built around the calculus of partial derivatives which may, at first, seem very far afield. This, however, turns out to be a very satisfying procedure. Most practical computations of vector relations operate by manipulating the components of the vectors - hence the latter are of fundamental importance. It turns out then, that, if needed, the direction and magnitude can be obtained from the components.

1.1.7 Vectors Defined by Components

Thus a vector is, first of all, an entity which has as many components as there are dimensions in the space being considered (i.e., 3 components in 3 dimensional space or N components in N dimensional space, where N is a positive integer). In some computations nothing more is said about the vectors being used, but usually the vectors can be assumed to have magnitudes (and directions) even though it may not be of interest to compute these properties. Usual computing practice is to use one or more coordinate systems and to state vector components relative to these coordinate systems. If all the computations are carried out in only one coordinate system then it is not necessary to evaluate the components relative to any other coordinate system. Vector theory assumes, however, that any one coordinate system defines a special case and that (at least potentially) there are a large number of other equally good coordinate systems. Thus there must be a set of rules whereby, if the components of a vector are known in any one coordinate system, they can be evaluated in any other "suitable" coordinate system. The various relations which are used for operating on and/or combining vectors must then be equally valid in all "suitable" coordinate systems. This concept is known as "invariance" (sometimes "covariance") and is fundamental in much of present day mathematics. The answer to the question "What constitutes a "suitable" coordinate system?" has a long history of evolution, but presently it includes about any coordinate system which can be imagined.

Suppose then that there is an x, y, z coordinate system (defined somehow) and that relative to this a u, v, w coordinate system can be computed by the (quite general) functional relations

$$u = f_1 (x, y, z)$$
$$v = f_2 (x, y, z)$$
$$w = f_3 (x, y, z).$$

Suppose, furthermore, that these functions can be solved to yield the inverse relations

$$x = g_1 (u, v, w)$$
$$y = g_2 (u, v, w)$$
$$z = g_3 (u, v, w).$$

The first set of functions will be assumed to have all of the nine first order partial derivatives:

$$\frac{\partial u}{\partial x} = \frac{\partial f_1}{\partial x}, \ \frac{\partial u}{\partial y} = \frac{\partial f_1}{\partial y}, \ \frac{\partial u}{\partial z} = \frac{\partial f_1}{\partial z},$$

$$\frac{\partial v}{\partial x} = \frac{\partial f_2}{\partial x}, \ \frac{\partial v}{\partial y} = \frac{\partial f_2}{\partial y}, \ \frac{\partial v}{\partial z} = \frac{\partial f_2}{\partial z},$$

$$\frac{\partial w}{\partial x} = \frac{\partial f_3}{\partial x}, \ \frac{\partial w}{\partial y} = \frac{\partial f_3}{\partial y}, \ \frac{\partial w}{\partial z} = \frac{\partial f_3}{\partial z}.$$

Likewise the second set of functions will be assumed to have all of the nine first order partial derivatives

$$\frac{\partial x}{\partial u}, \ \frac{\partial x}{\partial v}, \ \ldots$$

$$\frac{\partial y}{\partial u}, \ \ldots$$

$$\frac{\partial z}{\partial u}, \ \ldots .$$

(Sometimes higher order partial derivatives are also assumed, but these need not be considered here.) Since the second set of functions are inverse to the first set (i.e., substituting the values of x, y, and z given by the second set of functions into the first set of functions will result in identities in u, v, and w), the two sets of partial derivatives must satisfy the nine relations

$$\frac{\partial u}{\partial x}\frac{\partial x}{\partial u} + \frac{\partial u}{\partial y}\frac{\partial y}{\partial u} + \frac{\partial u}{\partial z}\frac{\partial z}{\partial u} = 1$$

$$\frac{\partial u}{\partial x}\frac{\partial x}{\partial v} + \frac{\partial u}{\partial y}\frac{\partial y}{\partial v} + \frac{\partial u}{\partial z}\frac{\partial z}{\partial v} = 0$$

$$\frac{\partial u}{\partial x}\frac{\partial x}{\partial w} + \frac{\partial u}{\partial y}\frac{\partial y}{\partial w} + \frac{\partial u}{\partial z}\frac{\partial z}{\partial w} = 0, \text{ etc.}$$

Evidently there is serious need for some short hand notation in representing these relations. Hence let

$$x_i = (x, y, z); \qquad (i = 1, 2, 3)$$

and

$$u_i = (u, v, w); \qquad (i = 1, 2, 3)$$

Then the above relations may be written in the form:

$$\sum_{j=1}^{3} \frac{\partial u_i}{\partial x_j}\frac{\partial x_j}{\partial u_k} = \begin{cases} 1 \text{ if } i = k \\ 0 \text{ if } i \neq k \end{cases}$$

Similarly, a set of inverse relations must also hold:

$$\sum_{j=1}^{3} \frac{\partial x_i}{\partial u_j}\frac{\partial u_j}{\partial x_k} = \begin{cases} 1 \text{ if } i = k \\ 0 \text{ if } i \neq k \end{cases}$$

(In both cases the relations are true for all combinations of i and k independently taking values 1, 2, or 3.)

1.1.8 Two Vectors Drawn From Practical Applications

In vector analysis two vectors which occur quite frequently are the differential displacement

$$dr = (dx,\ dy,\ dz)$$
$$= (dx_i); \quad (i = 1,\ 2,\ 3)$$

and the gradient of a scalar field

$$\Delta\phi = \left(\frac{\partial\phi}{\partial x}, \frac{\partial\phi}{\partial y}, \frac{\partial\phi}{\partial z}\right)$$
$$= \left(\frac{\partial\phi}{\partial x_i}\right); \quad (i = 1,\ 2,\ 3)$$

Note that these vectors are stated by defining their components relative to the x_i coordinate system. The corresponding components relative to the u_i coordinate system are then (by the rules of differential calculus):

$$du_i = \sum_{j=1}^{3} \frac{\partial u_i}{\partial x_j}\ dx_j; \quad (i = 1,\ 2,\ 3)$$

and

$$\frac{\partial\phi}{\partial u_i} = \sum_{j=1}^{3} \frac{\partial\phi}{\partial x_j}\ \frac{\partial x_j}{\partial u_i}; \quad (i = 1,\ 2,\ 3).$$

These two transformation laws have similar form (linear) but differ in that one uses the partial derivatives

$$\frac{\partial u_i}{\partial x_j}$$

as coefficients of the vector components relative to the x_i system whereas the other uses the <u>inverse</u> partial derivatives

$$\frac{\partial x_j}{\partial u_i}$$

as coefficients of the vector components relative to the x_i coordinate system.

1.1.9 Definition of Contravariant and Covariant Vectors

Using the above two vectors as models we, more generally, define a vector A as having contravariant components

$$(A_x, A_y, A_z) = A_{x_i}; \ (i = 1, 2, 3)$$

if the appropriate transformation law (which gives the components relative to the u_i coordinate system) is

$$A_{u_i} = \sum_{j=1}^{3} \frac{\partial u_i}{\partial x_j} A_{x_j}; \ (i = 1, 2, 3).$$

Similarly we define a vector B as having covariant components

$$(B_x, B_y, B_z) = B_{x_i}; \quad (i = 1, 2, 3)$$

if the appropriate transformation law is

$$B_{u_i} = \sum_{j=1}^{3} \frac{\partial x_j}{\partial u_i} B_{x_j}; \quad (i = 1, 2, 3).$$

Thus our calculus leads to two basic types of vector: the contravariant and the covariant type.* For convenience, a notation is commonly used which marks any particular components as to which type they are. This notation uses superscripts (which should not be confused with exponents) for contravariant components and it uses subscripts for covariant components. Since the differential displacement dr takes contravariant components the latter are represented by dx^i $(i = 1, 2, 3)$ rather than by dx_i. Correspondingly the coordinates themselves are therefore represented by x^i and u^i $(i = 1, 2, 3)$ rather than by x_i and u_i as was done previously.** Thus the two transformation laws may be written in the forms:

---

*In some discussions it is assumed that the vector entity exists above and beyond its components and that it may be represented by components of either type. Here, components of one type (only) will be used for some vectors and components of the other type (only) will be used for other vectors.

**Note that the change in notation does not imply any change whatsoever in meaning.

$$A^i = \sum_{j=1}^{3} \frac{\partial u^i}{\partial x^j} A^j; \quad \text{(contravariant)}$$

and

$$B_i = \sum_{j=1}^{3} \frac{\partial x^j}{\partial u^i} B_j; \quad \text{(covariant)}.$$

1.1.10 Use of Special Index Letters to Denote Coordinate Systems

While it is somewhat traditional to use the letters i, j, and k for indices (subscripts and superscripts) there is really no reason for not using other letters (or even more general symbols) as well, if some purpose is served by doing so. In succeeding chapters there will be occasion to use a number of different coordinate systems and it will avoid ambiguities if a notation is used which identifies the coordinate system which any particular set of components is relative to. Hence a unique set of letters will be used as indices for each different coordinate system. For now, only two different coordinate systems will be distinguished, but these two are to be thought of as generalized representations of all pairs of coordinate systems which may be introduced later.

The letters a, b, c, ··· will be used as indices for one representative coordinate system, and the letters m, n, p, ··· will be used as indices for the other representative coordinate system (note that any index is assumed to represent any positive integer from 1 to N, where N is the number of dimensions in the space being considered). With this convention the two sets of coordinates are represented by:

$$x^a = x,\ y,\ z; \qquad a = 1,\ 2,\ 3$$

and

$$x^m = u,\ v,\ w; \qquad m = 1,\ 2,\ 3.$$

At times x^a will be replaced by x^b or x^c (to avoid ambiguities which would otherwise occur). This does not imply any change in meaning. Similarly x^m will sometimes be replaced by x^n or x^p, but no different meaning is intended.

With this notation the two transformation laws may be written as:

$$A^m = \sum_{a=1}^{3} \frac{\partial x^m}{\partial x^a} A^a; \qquad (m = 1,\ 2,\ 3)$$

and

$$B_m = \sum_{a=1}^{3} \frac{\partial x^a}{\partial x^m} B_a; \qquad (m = 1,\ 2,\ 3).$$

1.1.11 Short Hand Notation for Partial Derivatives

Because the two sets of partial derivatives $\frac{\partial x^m}{\partial x^a}$ and $\frac{\partial x^a}{\partial x^m}$ occur so frequently it will be convenient to represent them by the short hand symbols X^m_a and X^a_m respectively. Then the transformations become:

$$A^m = \sum_{a=1}^{3} X^m_a A^a$$

and

$$B_m = \sum_{a=1}^{3} X^a_m B_a.$$

1.1.12 Omission of Special Sign for Summation

Finally it may be observed that the summation symbol $\sum_{a=1}^{3}$ is redundent, since the index "a" occurs in such a way that it alone may be considered as signifying the same thing which was heretofore signified by the combined presence of the summation symbol and an index used in this particular manner. Thus the transformation formulae will be written in the form:

$$A^m = X^m_a \ A^a$$

and

$$B_m = X^a_m \ B_a$$

which mean precisely the same things as though the summation symbol $\sum_{a=1}^{3}$ were written immediately after the = sign in each case. Note that there is also an implication that the formulae hold for m = 1, 2, 3, but explicit statement of this fact is customarily omitted.*

1.1.13 Summary of Conventions for Notation

To recapitulate; the short hand notation

$$A^m = X^m_a \ A^a$$

means precisely the same thing as

$$A^{u_i} = \sum_{j=1}^{N} \frac{\partial u_i}{\partial x_j} A^{x_j}; \quad (i = 1, 2, \cdots N).$$

Likewise:

$$B_m = X^a_m \ B_a$$

means precisely the same thing as

$$B_{u_i} = \sum_{j=1}^{N} \frac{\partial x_j}{\partial u_i} B_{x_j}; \quad (i = 1, 2, \cdots N),$$

*In N dimensional space the implied range of all indices is 1 to N.

where the integer N is the number of dimensions in the space being considered.

1.1.14 Inverse Transforms

The two transformation laws may be solved to yield the corresponding inverse transformations

$$A^a = X^a_m A^m$$

and

$$B_a = X^m_a B_m.$$

Evidently these are entirely analagous to the first forms.

1.1.15 Scalar Product of Two Vectors

Returning to the two vectors, differential displacement, and gradient, (which will now be represented by dx^a and $\frac{\partial \phi}{\partial x^a}$ respectively) we note that the dot (scalar) product of these may be written in the form:

$$\frac{\partial \phi}{\partial x^a} dx^a$$

This expression illustrates the general convention that an index which appears both contravariantly and covariantly in the same term signifies summation (just as though the summation symbol $\sum_1^N$ preceded the term). Now substitute the (inverse) transformation laws for both vectors

$$\frac{\partial \phi}{\partial x^a} = X^m_a \frac{\partial \phi}{\partial x^m}$$

and

$$dx^a = X^a_m dx^m.$$

Then

$$\frac{\partial \phi}{\partial x^a} dx^a = (X_a^m \frac{\partial \phi}{\partial x^m}) (X_n^a dx^n)$$

where n has the same meaning as m but runs through its range of values independently of m. This may also be written (without in the least changing its meaning)

$$\frac{\partial \phi}{\partial x^a} dx^a = X_a^m X_n^a \frac{\partial \phi}{\partial x^m} dx^n.$$

As was stated earlier (in different notation),

$$X_a^m X_n^a = \begin{cases} 1 & \text{if } m = n \\ 0 & \text{if } m \neq n. \end{cases}$$

Hence:

$$X_a^m X_n^a \frac{\partial \phi}{\partial x^m} dx^n = \delta_n^m \frac{\partial \phi}{\partial x^m} dx^n$$

$$(= \sum_{m=1}^{N} \sum_{n=1}^{N} \delta_n^m \frac{\partial \phi}{\partial x^m} dx^n)$$

$$= \frac{\partial \phi}{\partial x^m} dx^m,$$

where

$$\delta_n^m = \begin{cases} 1 & \text{if } m = n \\ 0 & \text{if } m \neq n. \end{cases}$$

Thus the expression $\frac{\partial \phi'}{\partial x^a} dx^a$, in the x^a coordinate system, transforms into the expression $\frac{\partial \phi}{\partial x^m} dx^m$, in the x^m coordinate system. This duplication of form before and after transformation is an example of invariance, and in this case it allows us to identify the result with the differential $d\phi$.

In precisely the same way we represent the dot (scalar) product of the vectors A^a and B_a by

$$A^a B_a = (X^a_m A^m) \; (X^n_a B_n)$$
$$= X^a_m X^n_a A^m B_n$$
$$= \delta^n_m A^m B_n = A^m B_m .$$

Again the result is an invariant form.

1.1.16 Alternate Derivation of Inverse Transforms

By applying a somewhat similar procedure to the transformation laws we can obtain the inverse transformations in straight forward fashion. Thus

$$A^a = X^a_m A^m$$
$$X^m_a A^a = X^m_a X^a_n A^n = \delta^m_n A^n$$
$$= A^m .$$

Likewise:

$$B_a = X^m_a B_m$$
$$X^a_m B_a = X^a_m X^n_a B_n = \delta^n_m B_n$$
$$= B_m .$$

At this point it would perhaps be logical to introduce the general definition of the "magnitude" of a vector. This topic will be more easily discussed, however, when the reader is familiar with the subject matter in the next section. Similarly the operation "cross product" can be discussed more meaningfully at a later point.*

*If any part of the discussion to this point is not clear to the reader, then the reader should substitute enough of the longer notation in the various expressions to satisfy himself that all of the results given really do follow logically from the definitions and from the principles of differential calculus. The reader should not be deceived by the brevity of the notation. The underlying meaning of these expressions is fundamental to all that follows.

1.2 Tensors*

1.2.1 Generalization of Vectors

Tensors are a straightforward generalization of vectors. Whereas vectors have N components (in N-dimensional space), tensors may have N^2 or N^3 or, in general, N^M - where M is any non-negative integer - components. M is then called the "order" (sometimes "rank") of the tensor. Thus vectors are first order tensors and scalars are zero order tensors. Like vectors, tensors may have either contravariant or covariant components. Tensors, of order two or greater may, however, also have mixed (i.e., partly contravariant and partly covariant) components.

A simple example of a second order tensor is the algebraic product of two vectors. Since each vector has N components then the product has N^2 components. For example, let the vectors A and B both have contravariant components. Then their transformation laws are:

$$A^m = X_a^m A^a$$

and

$$B^m = X_a^m B^a.$$

Multiplying these together (and changing the indices so as to avoid confusion):

$$A^m B^n = (X_a^m A^a) \quad (X_b^n B^b)$$

$$= X_a^m X_b^n A^a B^b.**$$

*This discussion is intended to discuss only such aspects of tensor analysis as are considered pertinent to the material which follows. Standard texts are available which cover a number of topics not treated here.

**The reader should now be able to recognize that this expression means the same thing as
$$A^m B^n = \sum_{a=1}^{N} \sum_{b=1}^{N} X_a^m X_b^n A^a B^b; \; m=1, \cdots N; \; n = 1, \cdots N.$$

This is an example of a tensor of contravariant order 2. If the two vectors are both covariant then the product takes the form

$$A_m B_n = (X^a_m A_a) \quad (X^b_n B_b)$$
$$= X^a_m X^b_n A_a B_b,$$

and this is an example of a tensor of covariant order 2. Finally the mixed tensor

$$A^m B_n = (X^m_a A^a) \quad (X^b_n B_b)$$
$$= X^m_a X^b_n A^a B_b$$

has contravariant order 1 and covariant order 1.

1.2.2 General Definition of Tensors

Thus a tensor of contravariant order p and covariant order q has its components relative to the first representative coordinate system represented as:

$$T^{a_1, a_2, \cdots a_p}_{b_1, b_2, \cdots b_q}.$$

Its components relative to the second representative coordinate system are then:

$$T^{m_1, m_2, \cdots m_p}_{n_1, n_2, \cdots n_q} = X^{m_1}_{a_1} X^{b_1}_{n_1} X^{m_2}_{a_2} X^{b_2}_{n_2} \cdots T^{a_1 \cdots a_p}_{b_1 \cdots b_q}$$

Thus the number of components in both coordinate systems is N^{p+q}. The notation on the right side of the transformation formula, of course, means independent summation on each of the indices $a_1, a_2, \cdots a_p, b_1, b_2, \cdots b_q$, each of which appears both contravariantly and covariantly in the same term. The tensor property resides in the fact that when the components are defined relative to some one coordinate system then corresponding components relative to any other coordinate system are given by a linear transformation law (such as the one shown) involving the partial derivatives X_a^m and the inverse partial derivatives X_m^a as coefficients.

1.2.3 Addition of Tensors

The following discussion is given using third order tensors but it could equally well be given for tensors of any order, the equations would then differ only in having the required number of indices and of partial derivatives. Let A_c^{ab} and B_c^{ab} be two tensors of contravariant order 2 and covariant order 1. Then their transformation laws are

$$A_p^{mn} = X_a^m X_b^n X_p^c A_c^{ab}$$

and

$$B_p^{mn} = X_a^m X_b^n X_p^c B_c^{ab} .$$

Adding these together then gives

$$A_p^{mn} + B_p^{mn} = (X_a^m X_b^n X_p^c A_c^{ab}) + (X_a^m X_b^n X_p^c B_c^{ab})$$

$$= X_a^m X_b^n X_p^c (A_c^{ab} + B_c^{ab}).$$

Thus the sum of two tensors of equal order is a tensor of the same order as both of them.

1.2.4 Multiplication of Tensors

Likewise multiplying the above two transformations:

$$A_p^{mn} B_s^{qr} = (X_a^m X_b^n X_p^c A_c^{ab}) (X_d^q X_e^r X_s^f B_f^{de})$$

$$= X_a^m X_b^n X_p^c X_d^q X_e^r X_s^f A_c^{ab} B_f^{de}.$$

Thus the product of two tensors is a tensor whose contravariant order is the sum of the contravariant orders, and whose covariant order is the sum of the covariant orders, of the two tensors being multiplied.

1.2.5 Contraction of Tensors

Mixed tensors are also subject to an operation known as "contraction." This operation consists of simultaneously equating and summing on one of the contravariant and one of the covariant indices. Thus consider the transformation formula

$$A_p^{mn} = X_a^m X_b^n X_p^c A_c^{ab}.$$

Now equate and sum on the indices n and p, then:

$$A_n^{mn} = X_a^m X_b^n X_n^c A_c^{ab}$$

$$= X_a^m \delta_b^c A_c^{ab}$$

$$= X_a^m A_b^{ab}.$$

Thus the result is a tensor with its order reduced by two. The above tensor can similarly be contracted by equating and summing on the indices m and p:

$$A_m^{mn} = X_a^m X_b^n X_m^c A_c^{ab}$$

$$= X_b^n \delta_a^c A_c^{ab}$$

$$= X_b^n A_a^{ab}.$$

This result is also a tensor with order reduced by two but it is not, in general, equal to the one above.

1.2.6 Inner Product of Tensors

Two tensors may, of course, be multiplied and then immediately contracted on a contravariant index of one tensor and a covariant index of the other tensor (assuming that the two tensors have such indices). This combination of operations is sometimes known as the "inner product" of the two tensors. If the two tensors are both vectors (one contravariant and the other covariant) then the result is the familiar "dot product" of the two vectors. Most of the applications of tensors in the following material will involve either this dot product of two vectors or else the inner product of a second order tensor by a vector; the latter producing a vector.

1.2.7 Some Special Types of Tensors

Evidently if the components of a tensor relative to some one coordinate system are all zero then the components relative to any other coordinate system are likewise all zero. Evidently, also, if the components of two tensors are respectively equal in one coordinate system then they are also equal in all other coordinate systems: i.e., if

$$A^{ab}_{c} = B^{ab}_{c}$$

then

$$\begin{aligned} A^{mn}_{p} &= X^{m}_{a} X^{n}_{b} X^{c}_{p} A^{ab}_{c} \\ &= X^{m}_{a} X^{n}_{b} X^{c}_{p} B^{ab}_{c} \\ &= B^{mn}_{p}. \end{aligned}$$

Tensors of contravariant order 2 or of covariant order 2, cannot, in general, have their two indices interchanged (without altering the value of the tensor). That is, in general,

$$A^{ab} \neq A^{ba}$$

and

$$A_{ab} \neq A_{ba} .$$

There is, however, a special class of tensor - known as symmetric - in which

$$A^{ab} = A^{ba}$$

(Note - that when this is true

$$A^{mn} = X^m_a X^n_b A^{ab}$$

$$= X^m_a X^n_b A^{ba}$$

$$= X^n_b X^m_a A^{ba}$$

$$= X^n_a X^m_b A^{ab}$$

$$= A^{nm} .)$$

Likewise the covariant symmetric tensor is such that

$$A_{ab} = A_{ba}$$

(hence $A_{mn} = A_{nm}$).

There is also a special class of tensor - known as skew-symmetric - such that

$$A^{ab} = -A^{ba}; \text{ (hence } A^{mn} = -A^{nm})$$

or

$$A_{ab} = -A_{ba}; \text{ (hence } A_{mn} = -A_{nm})$$

1.2.8 The Metric Tensor

An important example of a symmetric tensor is the so-called "metric" tensor. This may have either contravariant or covariant components and is usually represented by the base letter g. Thus:

$$g^{mn} = X^m_a X^n_b g^{ab}$$

$$= X^m_a X^n_b g^{ba}$$

$$= g^{nm}$$

and

$$g_{mn} = X^a_m X^b_n g_{ab}$$

$$= X^a_m X^b_n g_{ba}$$

$$= g_{nm}.$$

This metric tensor, besides being symmetric, has the important property that its components relative to any Cartesian coordinate system are components of the unit tensor. Thus let a and b be indices for a Cartesian coordinate system. Then:

$$g^{ab} = \delta^{ab} = \begin{cases} 1 & \text{if } a = b \\ 0 & \text{if } a \neq b \end{cases}$$

and

$$g_{ab} = \delta_{ab} = \begin{cases} 1 & \text{if } a = b \\ 0 & \text{if } a \neq b \end{cases}$$

Hence

$$g^{mn} = X^m_a X^n_b \delta^{ab}$$

$$\left(= \sum_{a=1}^{N} X^m_a X^n_a\right)$$

and

$$g_{mn} = X^a_m X^b_n \delta_{ab}$$

$$\left(= \sum_{a=1}^{N} X^a_m X^a_n\right)$$

Assume (again) that m and n are indices for a general coordinate system but that a and b are indices for a Cartesian coordinate system. Then:

$$g_{mn}\, g^{np} = (X^a_m\, X^b_n\, \delta_{ab}) \quad (X^n_c\, X^p_d\, \delta^{cd})$$

$$= X^a_m\, X^b_n\, X^n_c\, X^p_d\, \delta_{ab}\, \delta^{cd}$$

$$= X^a_m\, X^p_d\, \delta^b_c\, \delta_{ab}\, \delta^{cd}$$

$$= X^a_m\, X^p_d\, \delta^d_a$$

$$= X^a_m\, X^p_a$$

$$= \delta^p_m\ .$$

Likewise:

$$g^{mn}\, g_{np} = (X^m_a\, X^n_b\, \delta^{ab}) \quad (X^c_n\, X^d_p\, \delta_{cd})$$

$$= X^m_a\, X^n_b\, X^c_n\, X^d_p\, \delta^{ab}\, \delta_{cd}$$

$$= X^m_a\, X^d_p\, \delta^c_b\, \delta^{ab}\, \delta_{cd}$$

$$= X^m_a\, X^d_p\, \delta^a_d$$

$$= X^m_a\, X^a_p$$

$$= \delta^m_p\ .$$

Thus the contravariant and covariant components of the metric tensor are reciprocal to one another. The symbols δ^{ab}, δ_{ab}, and δ^a_b are all forms of what is known as the "Kronecker delta." All three forms have the values 1 if $a = b$ and 0 if $a \neq b$. The corresponding matrices are often called the unit matrix or the identity matrix. In the general coordinate system δ^{mn} and δ_{mn} do not occur, since they are replaced by g^{mn} and g_{mn}. The particular Kronecker delta δ^m_n is, however, valid even in the most general coordinates (since $\delta^a_b X^m_a X^b_n = X^m_a X^a_n = \delta^m_n$).

1.2.9 Formula for the Magnitude of A Vector

The metric tensor is used to give a general definition of the magnitude of a vector. Thus if the components of a contravariant vector are

$$A^m = X^m_a A^a$$

then the magnitude of this vector is defined as

$$\sqrt{g_{mn} A^m A^n}$$

$$= \sqrt{(X^a_m X^b_n g_{ab})(X^m_c A^c)(X^n_d A^d)}$$

$$= \sqrt{X^a_m X^m_c X^b_n X^n_d g_{ab} A^c A^d}$$

$$= \sqrt{\delta^a_c \delta^b_d g_{ab} A^c A^d}$$

$$= \sqrt{g_{ab} A^a A^b}.$$

Likewise for a covariant vector;

$$B_m = X^a_m B_a,$$

the magnitude is defined as

$$\sqrt{g^{mn} B_m B_n}$$

$$= \sqrt{(X^m_a X^n_b g^{ab})(X^c_m B_c)(X^d_n B_d)}$$

$$= \sqrt{X_a^m X_m^c X_b^n X_n^d g^{ab} B_c B_d}$$

$$= \sqrt{\delta_a^c \delta_b^d g^{ab} B_c B_d}$$

$$= \sqrt{g^{ab} B_a B_b}.$$

In both cases the magnitude is thus an invariant form. Hence if a and b are indices for a Cartesian coordinate system then the two magnitudes are

$$\sqrt{\delta_{ab} A^a A^b} = \sqrt{\sum_{a=1}^{N} (A^a)^2}$$

and

$$\sqrt{\delta^{ab} B_a B_b} = \sqrt{\sum_{a=1}^{N} (B_a)^2}$$

where the 2's are both exponents. Thus the magnitude of a vector is defined so as to be a generalization of the Pythagorean theorem.

1.2.10 Example of Use of the Magnitude Formula

As an example of the above, consider a vector relative to a spherical coordinate system. Let the spherical coordinates be r - the distance from the origin, θ - the colatitude, and ϕ - the longitude. Let x, y, and z be Cartesian coordinates related to r, θ, and ϕ by the relations

$$x = r \sin \theta \cos \phi$$

$$y = r \sin \theta \sin \phi$$

$$z = r \cos \theta.$$

From these we calculate the nine partial derivatives:

$$\frac{\partial x}{\partial r} = \sin\theta\cos\phi$$

$$\frac{\partial x}{\partial \theta} = r\cos\theta\cos\phi$$

$$\frac{\partial x}{\partial \phi} = -r\sin\theta\sin\phi$$

$$\frac{\partial y}{\partial r} = \sin\theta\sin\phi$$

$$\frac{\partial y}{\partial \theta} = r\cos\theta\sin\phi$$

$$\frac{\partial y}{\partial \phi} = r\sin\theta\cos\phi$$

$$\frac{\partial z}{\partial r} = \cos\theta$$

$$\frac{\partial z}{\partial \theta} = -r\sin\theta$$

$$\frac{\partial z}{\partial \phi} = 0$$

For convenience these may be listed in matrix form:

$$X^a_m = \begin{bmatrix} \sin\theta\cos\phi & r\cos\theta\cos\phi & -r\sin\theta\sin\phi \\ \sin\theta\sin\phi & r\cos\theta\sin\phi & r\sin\theta\cos\phi \\ \cos\theta & -r\sin\theta & 0 \end{bmatrix}$$

Using these:

$$g_{mn} = \delta_{ab} X^a_m X^b_n \quad (= \sum_{a=1}^{3} X^a_m X^a_n)$$

$$= \begin{bmatrix} 1 & 0 & 0 \\ 0 & r^2 & 0 \\ 0 & 0 & r^2\sin^2\theta \end{bmatrix}.$$

Let the vector under consideration be the velocity vector - with components relative to the spherical coordinates:

$$\frac{dx^m}{dt} = \left(\frac{dr}{dt}, \frac{d\theta}{dt}, \frac{d\phi}{dt}\right).$$

Hence the magnitude squared of this vector is

$$g_{mn} \frac{dx^m}{dt} \frac{dx^n}{dt} = \left(\frac{dr}{dt}\right)^2 + r^2 \left(\frac{d\theta}{dt}\right)^2 + r^2 \sin^2\theta \left(\frac{d\phi}{dt}\right)^2 .$$

1.2.11 Determinants and the Tensor Densities of Levi Civita

Since the metric tensor g_{mn} is defined to be non-singular it has a determinant. This determinant is customarily designated by the letter g for one coordinate system or by g' for another coordinate system. Since

$$g_{mn} = X^a_m X^b_n g_{ab}$$

then

$$g' = \left|X^a_m\right|^2 g,$$

where g is the determinant of g_{ab}, $\left|X^a_m\right|$ is the determinant of X^a_m and, g' is the determinant of g_{mn}. A little consideration shows that this determinant is unity for a Cartesian coordinant system. This transformation law is commonly said to define a scalar density of weight 2. Corresponding to this there are entities known as tensor densities - but the general definition will not be given here. Important examples of tensor densities, however, are those known as the tensor densities of Levi Civita. These may be generally defined for N dimensional space but only 3 dimensional examples will be given here. In three dimensional space the tensor densities of Levi Civita are represented by the symbols (for the two representative coordinate systems):

$$\epsilon^{abc}, \quad \epsilon_{abc}, \quad \epsilon^{mnp}, \quad \text{and} \quad \epsilon_{mnp}.$$

They are defined, in each case, to be skew symmetric in any pair of indices, to take the value 0 if any two indices are equal, and to take the value ± 1 if all three indices are unequal, i.e., the value + 1 if the indices are an even permutation of the numbers 1, 2, 3 and the value - 1 if the indices are an odd permutation of the numbers 1, 2, 3. With these definitions it may be demonstrated that

$$\epsilon^{abc} X^m_a X^n_b X^p_c = \left| X^m_a \right| \epsilon^{mnp}$$

and

$$\epsilon_{abc} X^a_m X^b_n X^c_p = \left| X^a_m \right| \epsilon_{mnp}$$

Where the symbol $\left| X^m_a \right|$ means the determinant of X^m_a and $\left| X^a_m \right|$ means the determinant of X^a_m.

Multiplying the second expression by $\sqrt{g}$ gives

$$\sqrt{g}\, \epsilon_{abc} X^a_m X^b_n X^c_p = \sqrt{g} \left| X^a_m \right| \epsilon_{mnp}$$

$$= \sqrt{g'}\, \epsilon_{mnp}.$$

Hence the product $\sqrt{g}\ \epsilon_{abc}$ is a tensor with covariant order 3 (in 3 dimensional space). Similarly $(1/\sqrt{g})\, \epsilon^{abc}$ may be shown to be a tensor with contravariant order 3 (in 3 dimensional space).

1.2.12 Vector Product of Two Vectors

The tensor densities of Levi Civita are used to define the vector (cross) product of two vectors. Thus, for the vectors A^a and B^b:

Approved For Release 2003/01/28 : CIA-RDP78B05171A000600030001-0

$$\sqrt{g}\,\epsilon_{abc}\,A^b\,B^c = \sqrt{g}\,\epsilon_{abc}\,(X^b_m\,A^m)\quad(X^c_n\,B^n)$$

$$= \delta^d_a\,\sqrt{g}\,\epsilon_{dbc}\,X^b_m\,X^c_n\,A^m\,B^n$$

$$= X^d_p\,X^p_a\,\sqrt{g}\,\epsilon_{dbc}\,X^b_m\,X^c_n\,A^m\,B^n$$

$$= X^p_a\,\sqrt{g}\,\epsilon_{dbc}\,X^d_p\,X^b_m\,X^c_n\,A^m\,B^n$$

$$= X^p_a\,\sqrt{g}\,\left|X^a_m\right|\,\epsilon_{pmn}\,A^m\,B^n$$

$$= X^p_a\,\sqrt{g'}\,\epsilon_{pmn}\,A^m\,B^n$$

Hence the cross product of two contravariant vectors is a covariant vector. Similarly, for the vectors A_a and B_a:

$$\frac{1}{\sqrt{g}}\,\epsilon^{abc}\,A_b\,B_c = \frac{1}{\sqrt{g}}\,\epsilon^{abc}\,(X^n_b\,A_n)\quad(X^p_c\,B_p)$$

$$= \delta^a_d\,\frac{1}{\sqrt{g}}\,\epsilon^{dbc}\,X^n_b\,X^p_c\,A_n\,B_p$$

$$= X^a_m\,X^m_d\,\frac{1}{\sqrt{g}}\,\epsilon^{dbc}\,X^n_b\,X^p_c\,A_n\,B_p$$

$$= X^a_m\,\frac{1}{\sqrt{g}}\,\epsilon^{dbc}\,X^m_d\,X^n_b\,X^p_c\,A_n\,B_p$$

$$= X^a_m\,\frac{1}{\sqrt{g'}}\,\epsilon^{mnp}\,A_n\,B_p\,.$$

The cross product of two covariant vectors is thus seen to be a contravariant vector.

1.2.13 Two Types of Relations Which Can be Written for Tensors

From all of the foregoing it should be evident that two types of expressions may be written which involve tensors (i.e., tensor components). One type of expression shows the transformation of the tensor components relative to one (representative) coordinate system into corresponding components relative to some other (representative) coordinate system. This transformation formula shows, explicitly, the contravariant and covariant orders of the components. The other type of expression may be said to define new tensors in terms of existing tensors. Thus the sum and product of two existing tensors are each of them newly defined tensors. Evidently the transformation formula may always be used to test whether any particular combination of existing tensors does, in fact, define a new tensor.

Expressions which define new tensors in terms of existing tensors are said to be invariant (or covariant), since the same form may always be identified before and after the transformation. While several methods have been given for forming new tensors, there are other methods which have not been discussed. Some of these get quite involved and are better left to the standard texts on the subject. One might expect, for example, that a new tensor may be formed by differentiating an existing tensor. Hence starting with the transformation formula

$$T_p^{mn} = X_a^m X_b^n X_p^c T_c^{ab}$$

and differentiating both sides with respect to x^d, we obtain

$$\frac{\partial T_p^{mn}}{\partial x^d} = \frac{\partial T_p^{mn}}{\partial x^q} X_d^q$$

$$= X_a^m X_b^n X_p^c \frac{\partial T_c^{ab}}{\partial x^d} + \frac{\partial X_a^m}{\partial x^d} X_b^n X_p^c T_c^{ab} + X_a^m \frac{\partial X_b^n}{\partial x^d} X_p^c T_c^{ab} + X_a^m X_b^n \frac{\partial X_p^c}{\partial x^d} T_c^{ab} .$$

The result is seen to be a tensor only if the partial derivatives X_a^m and X_m^a are independent of x^d, which is not generally the case. The way in which this gets fixed up is shown in standard text books on tensor analysis, but will not be discussed here.

1.2.14 Translation of Tensor Expressions into FORTRAN

It is of interest to show how tensor relations may be translated into the computer programming language FORTRAN. Thus consider the tensor equation

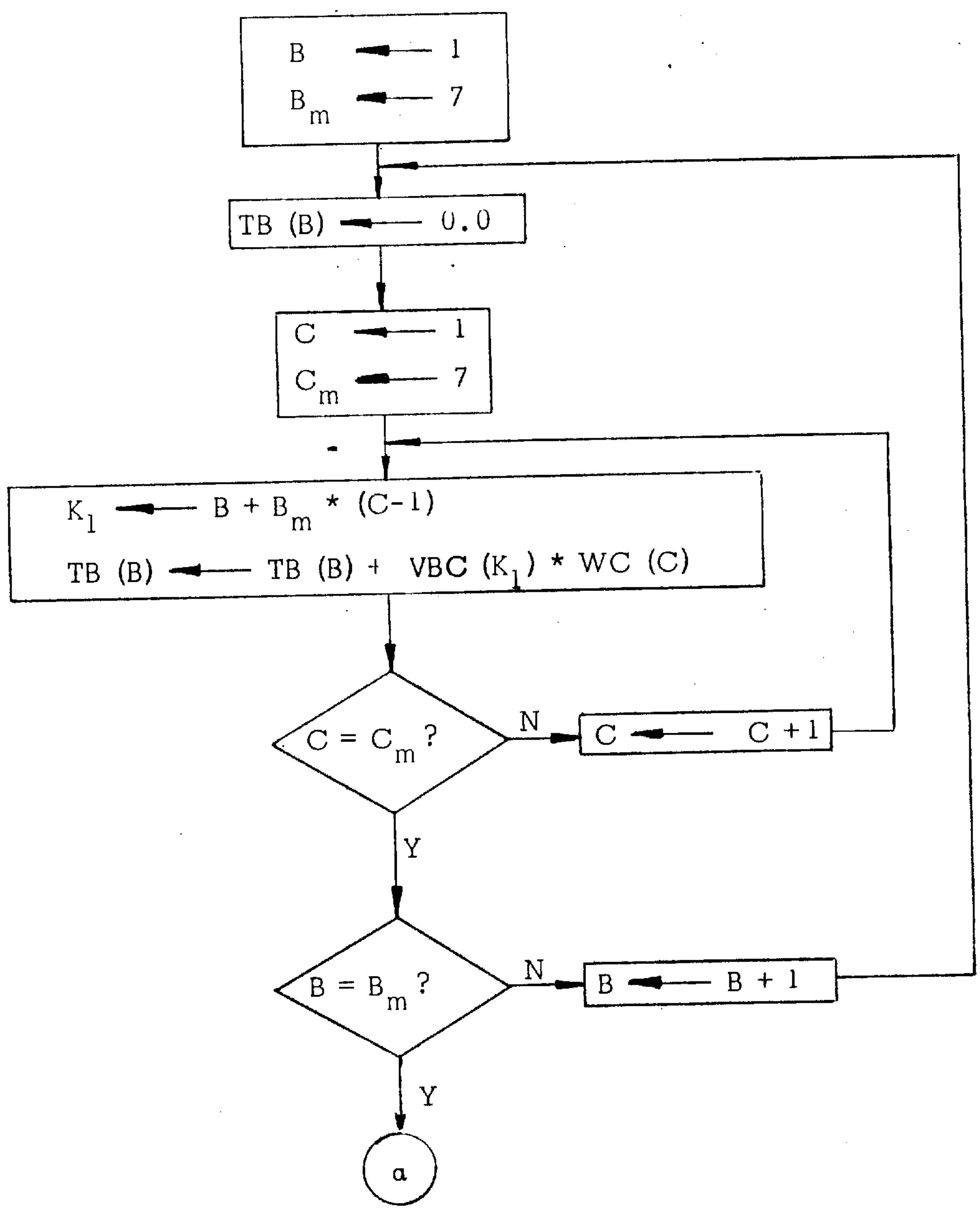
$$X_a^m = Y_a^m + U_{ab}^m V^{bc} W_c.$$

This may be translated into the following FORTRAN routine* (assuming that the space of interest has 7 dimensions):

```
      INTEGER M, A, B, C
      DIMENSION XMA (7,7), YMA (7,7), UMAB (7, 7, 7)
      DIMENSION VBC (7, 7), WC (7), TB (7)
      DO 10  B = 1, 7
      TB (B) = 0.0
      DO 10  C = 1, 7
10    TB (B) = TB (B) + VBC (B, C) * WC (C)
      DO 20  M = 1, 7
      DO 20  A = 1, 7
      XMA (M, A) = YMA (M, A)
      DO 20  B = 1,7
20    XMA (M, A) = XMA (M, A) + UMAB (M, A, B) * TB (B)
```

*Other FORTRAN routines are also possible; the one given here is thought to be more efficient than some others.

Fig. 1 shows a flow chart of this routine. Although this is only one example, and general rules have not been formulated, this example is thought to be sufficiently typical that readers who are familiar with FORTRAN programming will be able to thus program tensor expressions practically by inspection.



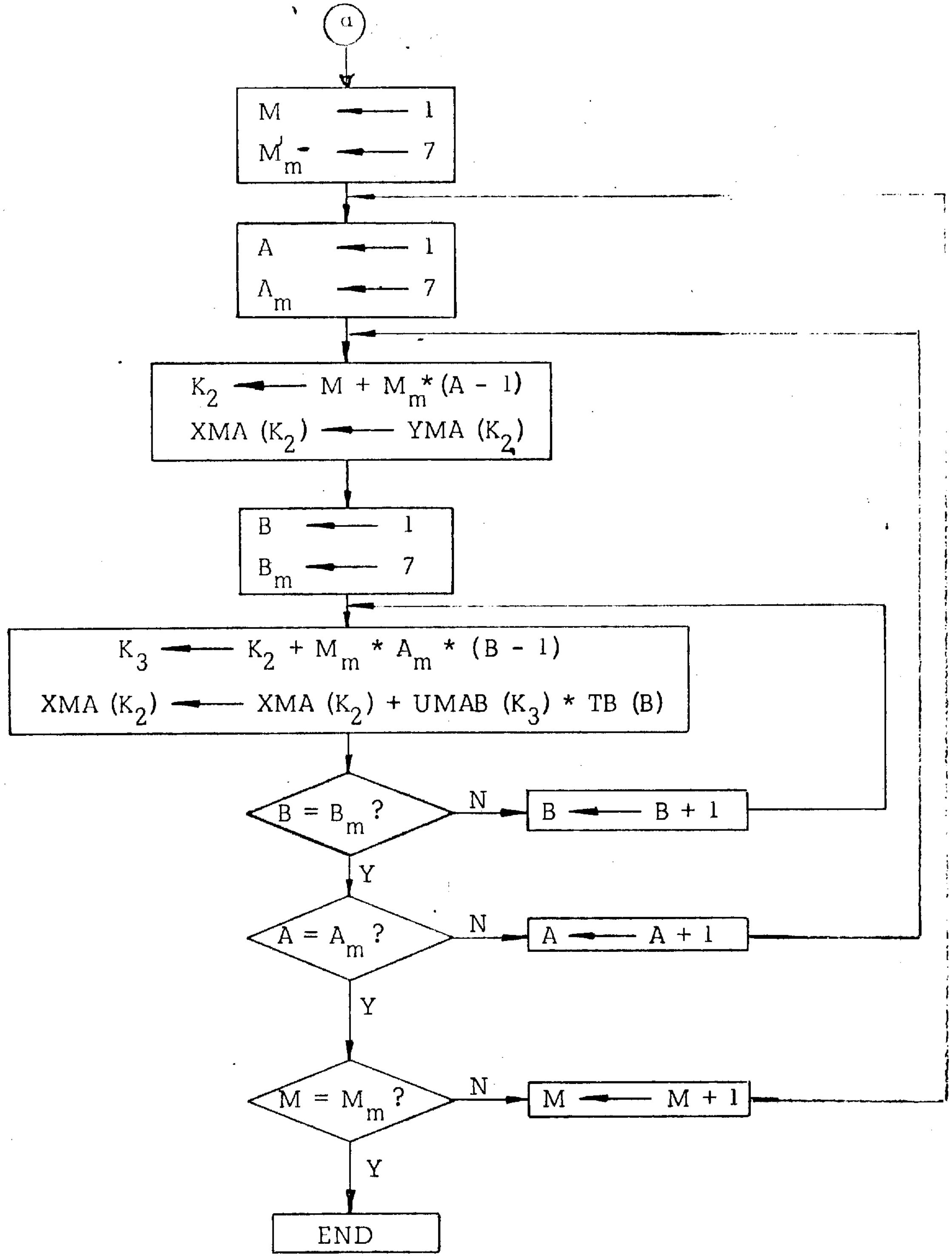


Figure 1. Flow Chart for example tensor equation.

1.3 Matrices

1.3.1 Use of Matrices to Exhibit Components of A Tensor

Although the availability of digital computers has greatly reduced the requirements for pencil and paper type computations, it is still sometimes desirable to be able to set down on paper the components of tensors relative to some coordinate system. Matrices provide an organized way of doing this. Corresponding to some of the tensor operations there are also rules for manipulating matrices. The latter are, in fact, sufficiently complete as to permit matrix solution of many problems without any use at all of tensor notation. For those who are proficient with tensor notation, however, the latter often brings to light useful relations (simplifications) which would probably not be discovered if the less explicit matrix notation were being used exclusively.

Although there are rules for dealing with third (and even higher) order matrices, practically all matrix computations are limited to operations with only first and second order matrices. Hence only these will be discussed here. First order matrices (representing components of vectors) are linear arrays (either rows or columns) of numbers. Second order matrices are rectangular (often square) arrays of numbers (which hence have both rows and columns).

The vector V^a may be represented by either a row or a column of components. Similarly the vector V_a may also be represented by either a row or a column. In other words, there is no fixed correspondence between contravariant and covariant vectors on the one hand and row vectors and column vectors on the other hand.

Similarly T^{ab}, T_{ab}, and T^a_b may all be represented by rectangular arrays of components. Thus, in general, a matrix does not show explicitly whether it represents covariant, contravariant, or mixed components. Likewise a matrix does not normally carry an explicit indication of what coordinate system its components are relative to. As a further source of ambiguity, either the first or second (upper or lower) index of the tensor may correspond to rows (or columns) of the matrix.

1.3.2 Linear Combination of Matrices

Matrix addition or subtraction means simply addition or subtraction of the elements (components) of the matrices. Likewise multiplication of a matrix by a scalar means multiplication of each element by the scalar. Thus a linear combination of matrices may be represented as follows:

$$\alpha \begin{bmatrix} A_{11} & A_{12} & \cdots A_{1n} \\ A_{21} & A_{22} & \cdots A_{2n} \\ \cdots & & \\ A_{m1} & A_{21} & \cdots A_{mn} \end{bmatrix} + \beta \begin{bmatrix} B_{11} & B_{12} & \cdots B_{1n} \\ B_{21} & B_{22} & \cdots B_{2n} \\ \cdots & & \\ B_{m1} & B_{m2} & \cdots B_{mn} \end{bmatrix}$$

$$= \begin{bmatrix} (\alpha A_{11} + \beta B_{11}) & (\alpha A_{12} + \beta B_{12}) & \cdots (\alpha A_{1n} + \beta B_{1n}) \\ (\alpha A_{21} + \beta B_{21}) & (\alpha A_{22} + \beta B_{22}) & \cdots (\alpha A_{2n} + \beta B_{2n}) \\ \cdots & & \\ (\alpha A_{m1} + \beta B_{m1}) & (\alpha A_{m2} + \beta B_{m2}) & \cdots (\alpha A_{mn} + \beta B_{mn}) \end{bmatrix}$$

where α and β are scalars and A_{11}, A_{12}, $\cdots B_{mn}$ are the elements of two m x n matrices; the result is, of course, also a m x n matrix.

Approved For Release 2003/01/28 : CIA-RDP78B05171A000600030001-0

The above matrix combination might, for example, represent any of the three tensor combinations:

$$\alpha A^{ab} + \beta B^{ab}$$

$$\alpha A_{ab} + \beta B_{ab}$$

$$\alpha A^{a}_{b} + \beta B^{a}_{b}.$$

1.3.3 Multiplication of Matrices

The so-called "product" of two matrices corresponds to the inner product (i.e., general product combined with contraction) of two second order tensors. Matrix multiplication is defined so as to be non-commutative; that is, (pre) multiplying the first matrix by the second is not the same as (pre) multiplying the second matrix by the first. The operation is defined so as to apply to rectangular matrices generally (i.e., non-square as well as square matrices). It requires, however, that the number of columns in the matrix which is to be the prefactor must be equal to the number of rows in the matrix which is to be the postfactor. The result then has the same number of rows as the prefactor and the same number of columns as the postfactor.

The rule for multiplying two matrices may be symbolized as follows:

1. Let A_{ij} represent the elements of the first matrix, with i corresponding to rows and j corresponding to columns.

2. Let B_{jk} represent the elements of the second matrix, with j corresponding to rows and k corresponding to columns.

3. The product C_{ik} is then given by the rule

$$C_{ik} = \sum_{j=1}^{N} A_{ij} B_{jk}.$$

Obviously, N is the number of columns in the first matrix and also the number of rows in the second matrix. The similarity to the following tensor inner products is obvious:

$$A^{ab} B_{bc}$$

$$A^{a}_{b} B^{b}_{c}$$

$$A_{ab} B^{bc}.$$

1.3.4 Transposition of Matrices

Matrices are also subject to the operation of transposing (i.e., interchanging) their rows and columns. If a particular matrix is represented by the symbol M (imagine bold face type) then the result of interchanging its rows and columns (called its transpose) is represented by M_t (again imagine bold face type). If the matrices A and B are both square and are equal in number of rows (and columns) then four (generally different) products can be formed by making use of their transposes:

$$AB, \; A_T B, \; AB_T, \; A_T B_T$$

In addition the following products may also be represented:

$$BA, \; B_T A, \; BA_T, \; B_T A_T.$$

It may be shown, however, that each of these is the transpose of one of the first four products. That is, it is generally true that if

$$C = AB$$

then

$$C_T = B_T A_T.$$

The four matrix products correspond to, for example, the following four tensor inner products:

$$A^{ae} B_{ed}, A^{eb} B_{ed}, A^{ae} B_{ce}, A^{eb} B_{ce}.$$

1.3.5 Inverse of A Matrix

If a particular matrix is square (i.e., it has an equal number of rows and columns) then one may compute the determinant of its elements. If this determinant is not equal to zero, the matrix is said to be non-singular, and a matrix may be found such that if multiplied by the original matrix (either pre or post multiplication) then the result is the unit matrix. (The unit matrix has all of its main diagonal elements equal to one and all other elements equal to zero.) In such a case the second matrix is called the inverse of the first matrix. Text books on matrix manipulations give rules for computing the inverse of a matrix, but these will not be stated here.

1.4 Cartesian Coordinate Systems

The preceding discussion has been written in terms of general coordinate systems in order to bring out the principle of invariance. According to this principle any proper combination of tensors has the same form irrespective of the coordinate system. Consequently it is not necessary to actually carry out a transformation of tensor forms when transforming from one coordinate system to another. One merely duplicates the form existing in one coordinate system - but substitutes the components of each tensor relative to the new coordinates.

In actual calculations, one uses Cartesian coordinates - almost always. This does not make the various tensor relations take a significantly simpler form, but it limits the numerical values that the components of the various tensors may have. For example, the partial derivatives X_a^m and X_m^a become constants (with respect to the coordinates - not necessarily with respect to other parameters, such as time) for any particular pair of Cartesian coordinate systems. In addition, as has been stated, the metric tensors become the Kronecker deltas with respect to any set of Cartesian coordinates. Hence the determinant g takes the value one when only Cartesian coordinate systems are used. Consequently this determinant is often omitted from the various tensor equations - but doing so makes the equations no longer invariant with respect to general coordinate transformations. The equations are then said to be invariant with respect to transformations among Cartesian coordinate systems (only). This will be done in much of what follows.

It can be demonstrated that all of the possible transformations between any two different sets of Cartesian coordinates are equivalent to combinations of parallel translations and rotations of the coordinate axes. For most purposes, the parallel translations are of trivial importance, and transformations among various Cartesian coordinate systems are commonly referred to as "coordinate rotations." The partial derivatives X_a^m and X_m^a (which then have unity determinants) are hence often referred to as "direction cosines" (of the various coordinate axes in one system with respect to those in the other). Accordingly the notations C_m^a and C_a^m (instead of X_m^a and X_a^m) will be used to represent transformations among Cartesian coordinate systems. As a result, in what follows it will generally be true that

$$\left|C_m^a\right| = \left|C_a^m\right| = 1,$$

and these determinants, also, will often be omitted. It is also a fact that when two reciprocal sets of direction cosines C_m^a and C_a^m are represented by their respective matrices (without indices) these two matrices are transposes of one another. This fact is used in the computer program, but is otherwise of little consequence.

When the permissible coordinate systems are limited to only those of Cartesian type, certain entities may be treated like tensors which are not tensors under more general transformations.

One example is the partial derivatives of the components of a tensor with respect to the coordinates. These derivatives were considered near the end of section 1.2 and found to define a tensor only when the partial derivatives are independent of the coordinates - exactly the situation when only Cartesian coordinates are considered. Another example is the incremental (more than just differential) displacement:

$$x^m - x_o^m = C_a^m (x^a - x_o^a).$$

These quantities may now be treated like components of a vector since the magnitude squared is

$$\delta_{mn} (x^m - x_o^m)(x^n - x_o^n)$$

$$= \delta_{mn} [C_a^m (x^a - x_o^a)] [C_b^n (x^b - x_o^b)]$$

$$= \delta_{mn} C_a^m C_b^n (x^a - x_o^a)(x^b - x_o^b)$$

$$= \delta_{ab} (x^a - x_o^a)(x^b - x_o^b),$$

i.e., invariant with respect to coordinate rotations. "Vectors" of this type will be used extensively in what follows.

II. OPTICAL IMAGING AND AERIAL PHOTOGRAPHS

2.1 Gaussian Optics

Most optical lenses consist of a series of interfaces between transparent media of different refractive index, which are all, as nearly as practical, spherical surfaces with their centers lying on a single straight line. The common line of centers is called the optical axis and the various spherical surfaces are incomplete spheres (usually less than hemi-spheres), usually with circular boundaries also centered on the optical axis. In somewhat rare instances lenses include one or more interfaces which are intentionally ground so as to depart from a spherical surface by a small, but definite, amount. These surfaces are, nevertheless, rotationally symmetrical (as nearly as practical) about the optical axis.

So-called "Gaussian optics" consists of a body of mathematical analyses of the refraction occurring at a series of such centered spherical interfaces between various optical media, which preserves only the degree of approximation obtained as the various "rays" considered, approach parallelism to the optical axis and also approach only an infinitesimal displacement from the optical axis. Thus Gaussian optics might be considered as a "zero order" approximation to the true analysis. This approximate analysis results, however, in what might be called a theory of "ideal" imaging, hence it is of prime importance. Real lenses are built so

Approved For Release 2003/01/28 : CIA-RDP78B05171A000600030001-0

as to have actual imaging properties substantially like the ideal imaging of Gaussian optics. Deviations from this ideal imaging are referred to as "aberations," and these aberations are made as small as is consistent with the intended price of any particular lens.

2.2 Projective Optics

Projective optics is a branch of geometrical optics that states a formal deductive theory of ideal imaging which corresponds closely to that of Gaussian optics. The basic assumptions of projective optics will here be used to derive the "general equation of optical imaging" which is the basis for the optical analysis used in subsequent chapters. This so-called "general" equation is, as stated above, an idealization of the real optical situation which neglects aberrations completely.

Projective optics considers that any optical system which has cylindrical symmetry about the optical axis may be treated as having the fundamental elements: (1) two principal planes, which are normal to the optical axis, (2) two focal points, which are on the optical axis, and (3) two nodal points, which are on the optical axis. These elements are illustrated in Figure 2, which also shows three rays diverging from a typical object point O and converging on the corresponding image point O'. These three rays are particular cases of three classes of rays. One class enters the first principal plane in a direction parallel to the optical axis; all such rays continue parallel to the optical axis until they intersect the second principal plane and then become "refracted" by just the right angle so they hence pass through the second focal point (F'). Rays of the second class pass through the first focal point (F) and continue until they intersect the first principal plane, at which point they become "refracted" so as to henceforth be parallel to the optical axis. The third class consists of rays which pass through the first nodal point (N) without having been previously refracted. All such rays emerge from the second nodal point (N') in a direction which is parallel to the direction in which they entered the first nodal point. Thus the ray $\overline{N'O'}$ is parallel to the ray $\overline{ON}$.

2.2.1 The General Equation of Optical Imaging

The above stated principles (of projective optics) and Figure 2 may be used to derive the relations between a typical object point (O) and the corresponding image point (O'). For this purpose assume an arbitrary Cartesian coordinate system (moving or stationary). Let X^a and X_1^a be the coordinates of O and N respectively. Likewise let x^a and x_1^a be the coordinates of O' and N' respectively. So long as only Cartesian type coordinate systems are considered, the displacements $X_1^a - X^a$ and $x^a - x_1^a$ may be treated as components of two vectors - which from the discussion above are known to be parallel to each other. (These two vectors are shown in Figure 2 as U and u respectively.) Parallel vectors have corresponding components which are respectively proportional. Hence: $x^a - x_1^a$ must be equal to some scalar quantity times $X_1^a - X^a$.

Figure 2 shows two pairs of similar triangles. One pair of similar triangles has a common junction at F and has a pair of corresponding legs lying along the optical axis. The other pair of similar triangles has a common junction at F' and also has a pair of corresponding legs lying along the optical axis. The lengths of the first pair of corresponding legs are seen to be (p.U - f') and f respectively,

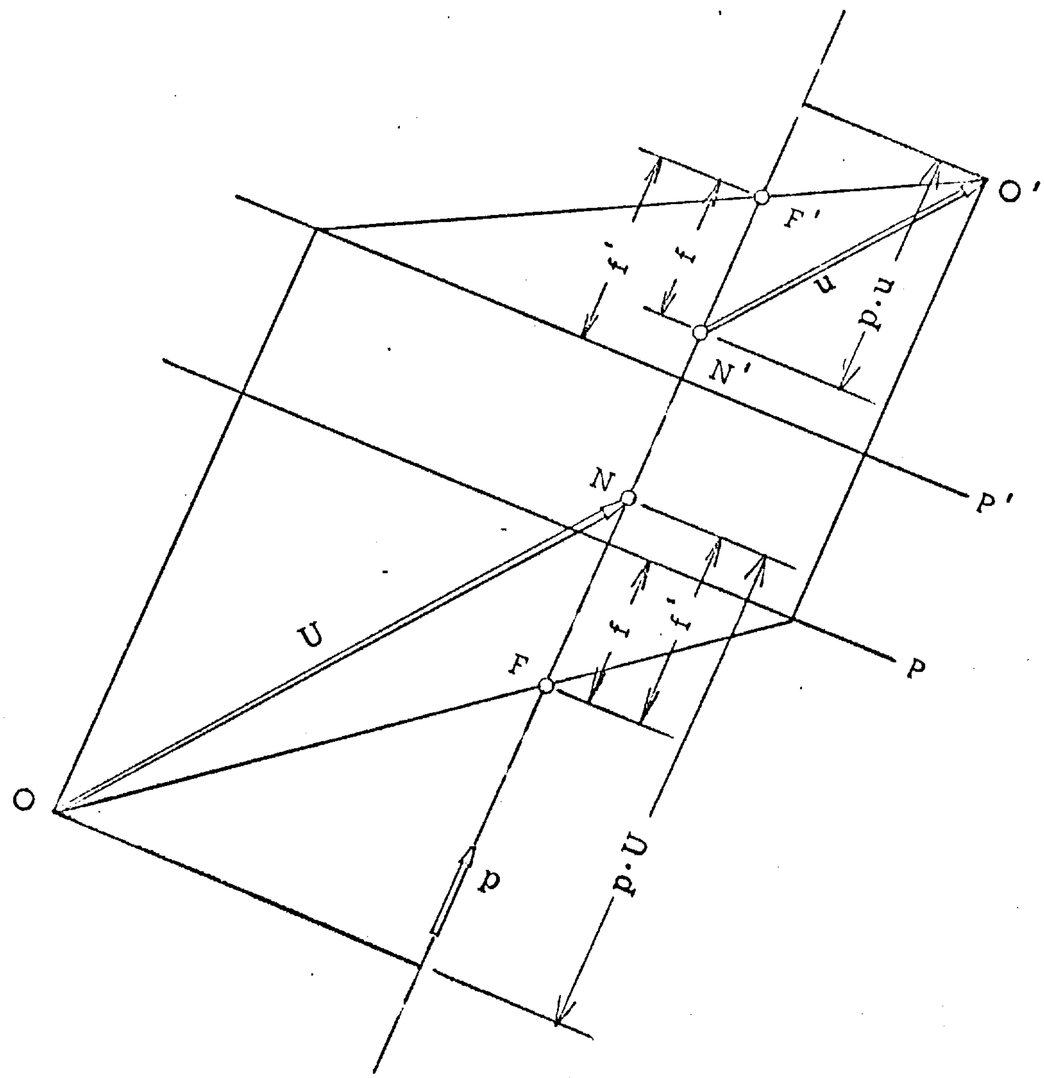


Fig. 2. Relations between an object point O and the image point O' formed by a lens with principal planes P and P'. Drawn for a positive lens, with f and f' both taken positive. For negative lenses f and f' are both negative.

where p is a unit vector parallel to the optical axis. The lengths of the second pair of corresponding legs are likewise seen to be f' and (p.u - f) respectively. Consequently the value of the scalar multiplier, mentioned at the end of the previous paragraph must be:

$$\frac{f}{p.U - f'} = \frac{p.u - f}{f'}.$$

Now let λ_a be the components of the unit vector p. Then

$$p.U = \lambda_a (X_1^a - X^a)$$

and

$$p.u = \lambda_a (x^a - x_1^a).$$

Using the ratio from the first pair of similar triangles, the desired relation between the two vectors U and u is thus

$$x^a - x_1^a = \frac{f (X_1^a - X^a)}{\lambda_b (X_1^b - X^b) - f'}$$

$$= \frac{f (X^a - X_1^a)}{f' + \lambda_b (X^b - X_1^b)}. \qquad (1)$$

This will be referred to as the optical imaging equation. Multiplying both sides of this equation by λ_a gives:

$$\lambda_a (x^a - x_1^a) = \frac{f \lambda_a (X^a - X_1^a)}{f' + \lambda_b (X^b - X_1^b)}.$$

Hence: $$p.u - f = \lambda_a (x^a - x_1^a) - f$$

$$= \frac{f \lambda_a (X^a - X_1^a)}{f' + \lambda_b (X^b - X_1^b)} - \frac{f [f' + \lambda_a (X^a - X_1^a)]}{f' + \lambda_b (X^b - X_1^b)}$$

$$= \frac{- f f'}{f' + \lambda_b (X^b - X_1^b)} = \frac{f f'}{p.U - f'}.$$

Thus the ratios derived separately from the two pairs of similar triangles are indeed equal to each other.

2.2.2 Invariance of the Optical Imaging Equation

That the optical imaging equation has been set up so as to be invariant to coordinate rotations, may be checked as follows:

(1.) Let C_a^m be the direction cosines of the (three dimensional) coordinate rotation, and let C_m^a be the reciprocal direction cosines.

(2.) The three vectors then have their components relative to the x^m coordinates given by

$$x^m - x_1^m = C_a^m (x^a - x_1^a),$$

$$X^m - X_1^m = C_a^m (X^a - X_1^a),$$

and

$$\lambda_m = C_m^a \lambda_a .$$

(3.) The above relations may be inverted to give:

$$x^a - x_1^a = C_m^a (x^m - x_1^m),$$

$$X^a - X_1^a = C_m^a (X^m - X_1^m),$$

and

$$\lambda_a = C_a^m \lambda_m$$

(4.) The imaging equation is therefore

$$C_n^a (x^n - x_1^n) = \frac{f\, C_n^a (X^n - X_1^n)}{f' + (C_b^m \lambda_m)\, C_p^b (X^p - X_1^p)}$$

$$= \frac{f\, C_n^a (X^n - X_1^n)}{f' + C_b^m C_p^b \lambda_m (X^p - X_1^p)} .$$

(5.) Multiplying both sides by C_a^m and using

$$C_a^m C_n^a = \delta_n^m \qquad (\text{and } C_b^m C_p^b = \delta_p^m)$$

then gives

$$x^m - x_1^m = \frac{f (X^m - X_1^m)}{f' + \lambda_n (X^n - X_1^n)} , \tag{2}$$

which has the desired form.

2.2.3 Use of Separate Coordinate Systems for Object Space and Image Space

The preceding discussion assumed that the same coordinate system was used for both the object space and the image space (these two spaces may be considered as separated by the principal planes). In practice, it is often desired to use one coordinate system for the object space and a different coordinate system for the image space. Evidently the imaging equation then takes the form

$$x^m - x_1^m = \frac{f\, C_a^m (X^a - X_1^a)}{f' + \lambda_b (X^b - X_1^b)} . \tag{3}$$

This equation is invariant to transformations among Cartesian coordinate systems both (separately) for the object space and for the image space. C_a^m is, of course, the set of direction cosines of the image space coordinate system with respect to the object space coordinate system.

2.3 Application to Aerial Photography

In an aerial camera the object point (O) is normally some point on the surface of the earth and the corresponding image point (O') is in the emulsion of a photographic film. Since the photographic film lies in a two dimensional surface (i.e., a plane surface for frame and strip cameras, a cylindrical surface for panoramic cameras) the object points which strictly satisfy the imaging equation also lie in a two dimensional surface. Aerial cameras, however, normally have sufficient depth of focus to satisfactorily image a substantial volume of space. Thus, in general, the imaging equation is only approximately satisfied with respect to camera focus. Specifically, the imaging equation correctly gives photograph coordinates as a function of ground coordinates but cannot be solved directly to give ground coordinates as a function of photograph coordinates. The most which can be inferred about ground coordinates from a single aerial photograph is their projective directions. Some additional information (or assumption) is necessary to determine the points at which rays projected from the photograph intersect the ground. Since the flying height is normally very much greater than the camera focal length, it is usual to neglect the rear focal length (f') which appears in the denominator and to write the imaging equation as:

$$x^m - x_1^m = \frac{f\, C_a^m \,(X^a - X_1^a)}{\lambda_b \,(X^b - X_1^b)} \, . \qquad (4)$$

This approximate form will be used from here on.

2.3.1 Rotation of Image

From Figure 2 it may be seen that the image in a camera is rotated 180° about the optical axis as compared to the apparent scene viewed directly. If, however, the photographic negative is viewed through its film base there is no mirror type reversal unless the camera contains one or more (i.e., an odd number of) mirrors, in addition to its lens. Correspondingly, then, a dup positive would normally be viewed emulsion side up in order not to show a mirror type reversal. By rotating the photograph coordinate system 180° about the optical axis, with respect to the ground coordinate system, one may make photograph coordinates take substantially the same algebraic signs as corresponding ground coordinates.*

2.3.2 Application to Frame, Strip, and Pan Types of Photography

Equation (4) is to be interpreted for three different types of photography: frame, strip, and panoramic. For frame and strip type the photographic film is exposed while confined in a geometric plane at a fixed normal distance from the camera lens. Assume that the camera motion is such that the direction normal to the film does not appreciably change direction during the exposure time for the photograph. Then the lens normal λ_b must also be constant in direction during the exposure time. Let the photograph coordinate system be oriented with its $x^{3'}$ (z) axis normal to the film. Then $\lambda_b = C_b^{3'}$ and equation (4) becomes:

*The common practice of regarding a positive as a negative projected through the point of perspective and having a negative focal length introduces a negative unit matrix which is not formally correct but which gets cancelled out in the usual photogrammetric computations.

$$x^m - x_1^m = \frac{f\, C_a^m\, (X^a - X_1^a)}{C_b^{3'}\, (X^b - X_1^b)} \tag{5}$$

This is the basic equation for frame and strip type photographs. X_1^a and x_1^m are constant for frame type but time functions for strip type. C_a^m (including $C_b^{3'}$) are constant for frame type and, for small regions at least, of strip type photographs.

For panoramic photographs the film is maintained in the form of half of a circular cylinder during exposure. The lens (and a slit) are rotated to produce a sweeping exposure. Hence the lens normal λ_b must be expressed in terms of the camera sweep angle α_1 ($=\omega t$). Let the photograph coordinate system be oriented so the $x^{1'}$ axis is parallel to the cylinder axis and the $x^{3'}$ axis is normal to the tangent plane at the "top" of the half-cylinder. Let α_1 be zero at the $x^{3'}$ axis and increasing positively in the direction toward the $x^{2'}$ axis. Then

$$\lambda_m = C_m^b\, \lambda_b = (0,\ \sin \alpha_1,\ \cos \alpha_1)$$

Hence (using (4))

$$\lambda_m\, (x^m - x_1^m) = (x^{2'} - x_1^{2'}) \sin \alpha_1 + (x^{3'} - x_1^{3'}) \cos \alpha_1 = f \tag{6}$$

Evidently (6) will be true if

$$x^{2'} - x_1^{2'} = f \sin \alpha_1 \tag{7}$$

and

$$x^{3'} - x_1^{3'} = f \cos \alpha_1 \tag{8}$$

Thus $x^m - x_1^m$ are rectangular coordinates of the circular cylinder relative to a coordinate system in which the film is stationary.

Combining (7) and (8) with (4):

$$x^m - x_1^m = f \cos \alpha_1 \frac{C_a^m (X^a - X_1^a)}{C_b^{3'} (X^b - X_1^b)} \qquad (9)$$

with

$$\alpha_1 = \tan^{-1} \frac{C_a^{2'} (X^a - X_1^a)}{C_b^{3'} (X^b - X_1^b)} \qquad (10)$$

Equations (9) and (10) are basic for a panoramic type photograph. X_1^a, x_1^m, and α_1 are time functions depending on the camera motion, the [redacted] mechanism, and the lens sweep mechanism.

Equations (5) and (9), (10) are basically correct but are of little value until the various time functions are evaluated. Evaluation of these time functions is dependent on the particular cameras and on the flight pattern. The following examples are cursory and are based on descriptions of particular cameras which have been published in the literature and assume uniform flight velocity.

2.3.2.1 Frame type photography

The film is maintained in a plane and exposed simultaneously over the whole photograph. Image motion is neglected - since the exposure time is brief.

$$x^m - x_1^m = \frac{f_1 \, C_a^m \, (X^a - X_1^a)}{C_b^{3'} \, (X^b - X_1^b)}$$

$$x_1^m = x_{10}^m \quad \text{(constant)}$$

$$X_1^a = X_{10}^a \quad \text{(constant)}$$

$$\therefore \quad x^m - x_1^m = (x^{1'} - x_{10}^{1'}, \; x^{2'} - x_{10}^{2'}, \; f_1)$$

2.3.2.2 Strip type photography

The film is maintained in a flat plane but is exposed by moving it past a narrow slit which results in a sweeping exposure. The exposure time for any small area is brief but the time interval required to sweep the whole photograph is appreciable. There is a fixed angle β_1 associated with the slit-lens scanning operation. This angle is here taken positive for backward looking, or negative for forward looking, slits. Let the photograph coordinate system be oriented with its $x^{3'}$ axis normal to the film plane and its $x^{1'}$ axis parallel, but opposite, to the direction of lens-slit motion relative to the film [redacted] Assume that the time functions are linear in t. STAT

$$x^m - x_1^m = \frac{f_1 \, C_a^m \, (X^a - X_1^a)}{C_b^{3'} \, (X^b - X_1^b)}$$

$$X_1^a \cong X_{10}^a + V^a \, t$$

$$x_1^m \cong x_{10}^m + v \, \delta_{1'}^m \, t$$

$$x^m - x_1^m = (f_1 \tan\beta_1,\ x^{2'} - x_{10}^{2'},\ f_1)$$

$$\therefore\quad t \cong \frac{x^{1'} - x_{10}^{1'} - f_1 \tan\beta_1}{v}$$

$$\cong \frac{(C_a^{1'} - C_a^{3'} \tan\beta_1)\ (X^a - X_{10}^a)}{(C_b^{1'} - C_b^{3'} \tan\beta_1)\ V^b}$$

where V^b is the camera ground speed vector, and v is the velocity [redacted] (here taken as a <u>negative</u> number). STAT

2.3.2.3 Panoramic type photography

The film is maintained in half of a circular cylinder and exposed by a slit which scans around the cylinder. The exposure time for any small area is brief, but the total scanning time is appreciable. There is a scanning angle α_1 which increases at a <u>rate</u> approximately <u>uniform</u> in time, and is here taken positive in the direction from the $x^{3'}$ toward the $x^{2'}$ axes. Let the photograph coordinate system be oriented with its $x^{1'}$ axis parallel to the axis of the cylinder and with $\alpha_1 = 0$ at the $x^{3'}$ axis. Assume that the lens motion for [redacted] is in the negative $x^{1'}$ direction. STAT Assume that X_1^a is a linear function of time but that the [redacted] STAT velocity is proportional to $\cos\alpha_1$.

$$x^m - x_1^m = f_1 \cos \alpha_1 \frac{C_a^m}{C_b^{3'}} \frac{(X^a - X_1^a)}{(X^b - X_1^b)}$$

$$\alpha_1 = \tan^{-1} \frac{C_a^{2'} (X^a - X_1^a)}{C_b^{3'} (X^b - X_1^b)}$$

$$\alpha_1 \cong \omega t$$

$$X_1^a \cong X_{10}^a + V^a t$$

$$x_1^m \cong x_{10}^m + \frac{v_M}{\omega} \delta_1^{m\cdot\cdot} \sin \alpha_1$$

For measurement, the panoramic photograph is laid out flat. The linear coordinate in the scan direction is then

$$y - y_{10} = f_1 \alpha_1$$

but there is no change in the coordinate $x^{1'} - x_1^{1'}$ - parallel to the cylinder axis. Quantitative use of a panoramic photograph is usually for calculation of derived quantities (ground coordinates, or an equivalent image) as a function of measured photograph coordinates. Hence the following substitutions will usually be made:

$$x^m - x_1^m = (x^{1'} - x_{10}^{1'} - \frac{v_M}{\omega} \sin \frac{y - y_{10}}{f_1}, f_1 \sin \frac{y - y_{10}}{f_1}, f_1 \cos \frac{y - y_{10}}{f_1})$$

These follow quite easily from the preceding equations but they are discussed further in section 3.1.4.3.

STAT

2.4 [redacted] Series Expansion of the Projective Equations

Equations (5), (9), and (10) may be collected and written in the following generalized format:

$$x^m = x_1^m + f \cos \alpha \frac{C_a^m (X^a - X_1^a)}{C_b^{3'} (X^b - X_1^b)} \tag{11}$$

with

$$\alpha = \begin{cases} 0; & \text{frame photos} \\ 0; & \text{strip photos} \\ \tan^{-1} \dfrac{C_a^{2'} (X^a - X_1^a)}{C_b^{3'} (X^b - X_1^b)}; & \text{pan photos.} \end{cases} \tag{12}$$

These equations are not, however, entirely explicit since x_1^m, α, C_a^m, and X_1^a are, in general, functions of time (t). These functions of time are not usually available in closed form but can often be approximated by the first few terms of their [redacted] series expansions. Hence it is of interest to examine the [redacted] series expansion of equation (11). STAT STAT

To expand (11) as a [redacted] series it is necessary to find the various orders of partial derivatives of equations (11) and (12) with respect to X^a. In doing so it will be convenient to represent the partial derivatives taken two ways. The expressions STAT

$$\frac{\partial x^m}{\partial X^a}, \; \frac{\partial x^m}{\partial x_1^n}, \quad \ldots$$

$$\frac{\partial X_a^m}{\partial X^b}, \; \frac{\partial X_a^m}{\partial X_1^b}, \quad \ldots$$

will represent partial derivatives taken by treating x_1^m, α, C_a^m, and X_1^a like independent variables. On the other hand the expressions

$$X_a^m, \; X_{ab}^m, \; X_{abc}^m, \; \cdots$$

will represent partial derivatives taken by treating x_1^m, α, C_a^m, and X_1^a as functions of t (time) which is in turn treated as a function of X^a.

Hence:

$$X_a^m = \frac{\partial x^m}{\partial X^a} + \left(\frac{\partial x^m}{\partial x_1^n} \frac{dx_1^n}{dt} + \frac{\partial x^m}{\partial \alpha} \frac{d\alpha}{dt} + \frac{\partial x^m}{\partial C_c^n} \frac{dC_c^n}{dt} + \frac{\partial x^m}{\partial X_1^c} \frac{dX_1^c}{dt} \right) \frac{\partial t}{\partial X^a} \tag{13}$$

$$X_{ab}^m = \frac{\partial X_a^m}{\partial X^b} + \left(\frac{\partial X_a^m}{\partial x_1^n} \frac{dx_1^n}{dt} + \frac{\partial X_a^m}{\partial \alpha} \frac{d\alpha}{dt} + \frac{\partial X_a^m}{\partial C_c^n} \frac{dC_c^n}{dt} + \frac{\partial X_a^m}{\partial X_1^c} \frac{dX_1^c}{dt} \right.$$
$$\left. + \frac{\partial X_a^m}{\partial \frac{dx_1^n}{dt}} \frac{d^2 x_1^n}{dt^2} + \frac{\partial X_a^m}{\partial \frac{d\alpha}{dt}} \frac{d^2 \alpha}{dt^2} + \frac{\partial X_a^m}{\partial \frac{dC_c^n}{dt}} \frac{d^2 C_c^n}{dt^2} + \frac{\partial X_a^m}{\partial \frac{dX_1^c}{dt}} \frac{d^2 X_1^c}{dt^2} \right) \frac{\partial t}{\partial X^b} \tag{14}$$

$$X_{abc}^m = \frac{\partial X_{ab}^m}{\partial X^c} + \cdots \text{ etc.}$$

STAT

The [redacted] series is then

$$x^m = x_o^m + X_a^m \, \Delta X^a + \frac{1}{2} X_{ab}^m \, \Delta X^a \, \Delta X^b + \frac{1}{3!} X_{abc}^m \, \Delta X^a \, \Delta X^b \, \Delta X^c + \cdots \tag{15}$$

where x_o^m are the photograph coordinates of the "center" of the region over which (15) is useful. If the expansion is desired for a certain ground region then x_1^m, α, C_a^m, X^a, and X_1^a must all be determined for a point near the center of this ground region and substituted into (11) and (12), to give x_o^m, and into (15). Thus for any particular ground region, over which (15) is considered valid, x_1^m, α, C_a^m, X^a, and X_1^a are all treated as constants. Hence over this ground region, x_o^m are constants and ΔX^a are the independent variables which correspond to motion over the region.

2.4.1 Evaluation of the Partial Derivatives

The various partial derivatives are obtained as follows. From (11)

$$\frac{\partial x^m}{\partial X^a} = f \cos \alpha \left\{ \frac{C_a^m}{C_b^{3'} (X^b - X_1^b)} - \frac{C_b^m (X^b - X_1^b) C_a^{3'}}{[C_c^{3'} (X^c - X_1^c)]^2} \right\}$$

$$= f \cos \alpha \frac{(C_a^m C_b^{3'} - C_a^{3'} C_b^m)(X^b - X_1^b)}{[C_c^{3'} (X^c - X_1^c)]^2} \qquad (16)$$

$$\frac{\partial x^m}{\partial x_1^n} = \delta_n^m \qquad (17)$$

$$\frac{\partial x^m}{\partial \alpha} = -f \sin \alpha \frac{C_a^m (X^a - X_1^a)}{C_b^{3'} (X^b - X_1^b)} \qquad (18)$$

$$\frac{\partial x^m}{\partial C^n_c} = f \cos \alpha \left\{ \frac{\delta^m_n (X^c - X^c_1)}{[C^{3'}_d (X^d - X^d_1)]} - \frac{C^m_a (X^a - X^a_1)\, \delta^{3'}_n (X^c - X^c_1)}{[C^{3'}_d (X^d - X^d_1)]^2} \right\}$$

$$= f \cos \alpha \frac{(\delta^m_n C^{3'}_a - C^m_a \delta^{3'}_n)(X^a - X^a_1)(X^c - X^c_1)}{[C^{3'}_d (X^d - X^d_1)]^2} \tag{19}$$

$$\frac{\partial x^m}{\partial X^c_1} = - \frac{\partial x^m}{\partial X^c} \tag{20}$$

The total derivatives $\frac{dx^n_1}{dt}$, $\frac{d\alpha}{dt}$, $\frac{dC^n_c}{dt}$, and $\frac{dX^c_1}{dt}$ are taken from the camera motion along the flight path and [redacted] sweep motions in the camera, and are often treated as constants though, in general, their values vary from one region to another. STAT

2.4.2 Time (t) Treated as A Function of the Ground Coordinates

The partial derivatives $\frac{\partial t}{\partial X^a}$ are different for each different type of photography and are as follows:

2.4.2.1 Frame Photography

All points in the photograph are exposed simultaneously.

Hence:

$$\frac{\partial t}{\partial X^a} = 0$$

2.4.2.2 Strip Photography

Since $x^{1'} - x^{1'}_1 = f \tan \beta$ where β is a constant angle, characteristic of each particular strip camera (zero for systems currently of interest), equation (11) for $m = 1'$ becomes

$$\tan \beta = \frac{C^{1'}_a (X^a - X^a_1)}{C^{3'}_b (X^b - X^b_1)} \tag{21}$$

Multiplying through by the denominator of the right side and differentiating with respect to X^a gives

$$C_b^{1'} \left[\delta_a^b - \frac{dX_1^b}{dt} \frac{\partial t}{\partial X^a}\right] + \frac{dC_b^{1'}}{dt} \frac{\partial t}{\partial X^a} (X^b - X_1^b)$$

$$= \tan\beta \left\{C_b^{3'} \left[\delta_a^b - \frac{dX_1^b}{dt} \frac{\partial t}{\partial X^a}\right] + \frac{dC_b^{3'}}{dt} \frac{\partial t}{\partial X^a} (X^b - X_1^b)\right\} \qquad (22)$$

Solving (22) for $\frac{\partial t}{\partial X^a}$ then gives

$$\frac{\partial t}{\partial X^a} = \frac{C_a^{1'} - C_a^{3'} \tan\beta}{(C_b^{1'} - C_b^{3'} \tan\beta) \frac{dX_1^b}{dt} - \left(\frac{dC_b^{1'}}{dt} - \frac{dC_b^{3'}}{dt} \tan\beta\right)(X^b - X_1^b)} \qquad (23)$$

2.4.2.3 Panoramic Photography

The last part of (12) may be written in the form

$$C_a^{2'} (X^a - X_1^a) = C_a^{3'} (X^a - X_1^a) \tan\alpha. \qquad (24)$$

Differentiating (24) with respect to X^a gives

$$C_b^{2'} \left(\delta_a^b - \frac{dX_1^b}{dt} \frac{\partial t}{\partial X^a}\right) + \frac{dC_b^{2'}}{dt} \frac{\partial t}{\partial X^a} (X^b - X_1^b)$$

$$= \left[C_b^{3'} \left(\delta_a^b - \frac{dX_1^b}{dt} \frac{\partial t}{\partial X^a}\right) + \frac{dC_b^{3'}}{dt} \frac{\partial t}{\partial X^a} (X^b - X_1^b)\right] \tan\alpha$$

$$+ C_b^{3'} (X^b - X_1^b) \frac{d\alpha}{dt} \frac{\partial t}{\partial X^a} \sec^2\alpha \qquad (25)$$

Equation (25) may be solved for $\frac{\partial t}{\partial X^a}$. Equation (24) is then used to substitute for tan α and for $\sec^2 \alpha = 1 + \tan^2 \alpha$. The result is

$$\frac{\partial t}{\partial x^a} = \frac{(C_a^{2'} C_b^{3'} - C_a^{3'} C_b^{2'})(X^b - X_1^b)}{(C_c^{2'} C_d^{3'} - C_c^{3'} C_d^{2'})(X^d - X_1^d)\frac{dX_1^c}{dt} - (\frac{dC_c^{2'}}{dt} C_d^{3'} - \frac{dC_c^{3'}}{dt} C_d^{2'})(X^c - X_1^c)(X^d - X_1^d) + \{[C_c^{2'}(X^c - x_1^c)]^2 + [C_c^{3'}(X^c - X_1^c)]^2\}\frac{d\alpha}{dt}} \quad (26)$$

Evidently (16) through (20) and (23) or (26) must be substituted in (13) before applying (14), and similarly for higher order derivatives. Fortunately, however, only terms of (15) through the first order will be needed in what follows. Note that (15) gives the rectangular coordinates, hence for panoramic photographs it must be supplemented by the following formula for the photograph coordinate in the scan direction:

$$y - y_{10} = f \tan^{-1} \frac{x^{2'} - x_{10}^{2'}}{x^{3'} - x_{10}^{3'}}. \quad (27)$$

2.4.3 Generalized Formula for the Partial Derivatives

Equations (13), (14), (22), and (25) are written for coordinate systems which are stationary with respect to the ground (i.e., the object photographed). For such coordinate systems

$$\frac{dX^a}{dt} = 0 \qquad (28)$$

and

$$\frac{d^2X^a}{dt^2} = 0. \qquad (29)$$

Using (28), equation (13) can have the term

$$\frac{\partial x^m}{\partial X^c} \frac{dX^c}{dt} \frac{\partial t}{\partial X^a}$$

added to its right side without upsetting the equality. If this is done, then (13) becomes

$$X_a^m = \frac{\partial x^m}{\partial X^a} + \frac{dx^m}{dt} \frac{\partial t}{\partial X^a} \qquad (30)$$

Similarly (14) can have

$$\left(\frac{\partial X_a^m}{\partial X^c} \frac{dX^c}{dt} + \frac{\partial X_a^m}{\partial \frac{dX^c}{dt}} \frac{d^2X^c}{dt^2}\right) \frac{\partial t}{\partial X^b}$$

added to its right side, and it then becomes

$$X_{ab}^m = \frac{\partial X_a^m}{\partial X^b} + \frac{dX_a^m}{dt} \frac{\partial t}{\partial X^b}. \qquad (31)$$

Similarly (22) and (25) are respectively equivalent to

$$\frac{\partial}{\partial X^a} [(C_c^{1'} - C_c^{3'} \tan \beta) (X^c - X_1^c)]$$

$$+ \frac{d}{dt} [(C_c^{1'} - C_c^{3'} \tan \beta) (X^c - X_1^c)] \frac{\partial t}{\partial X^a} = 0 \qquad (32)$$

and

$$\frac{\partial}{\partial X^a}\left[(C_c^{2'} - C_c^{3'} \tan \alpha)(X^c - X_1^c)\right] + \frac{d}{dt}\left[(C_c^{2'} - C_c^{3'} \tan \alpha)(X^c - X_1^c)\right] \frac{\partial t}{\partial X^a} = 0. \quad (33)$$

Hence (23) may be written

$$\frac{\partial t}{\partial X^a} = - \frac{C_a^{1'} - C_a^{3'} \tan \beta}{\frac{d}{dt}\left[(C_b^{1'} - C_b^{3'} \tan \beta)(X^b - X_1^b)\right]} \quad (34)$$

and (26) may be replaced by

$$\frac{\partial t}{\partial X^a} = - \frac{C_a^{2'} - C_a^{3'} \tan \alpha}{\frac{d}{dt}\left[(C_b^{2'} - C_b^{3'} \tan \alpha)(X^b - X_1^b)\right]} \quad (35)$$

Note that (24) cannot be substituted into (35) until after the differentiation in the denominator has been performed.

2.4.4 Invariance of the Generalized Formula

Examination of (30) through (35) shows that all but (31) have forms which are invariant with respect to translation and/or rotation of the ground coordinate system. Equation (31) would also be invariant if the derivatives were covariant derivatives; the latter are not discussed in this paper, however. Thus these equations are valid for all Cartesian coordinate systems, whether moving or stationary with respect to the ground. The fact that these equations have invariant forms allows X_a^m, X_{ab}^m, $\cdots$ to be treated as tensors. Hence if ΔX^a is treated as a tensor, then (15) also is valid for moving, as well as for stationary, coordinate systems.*

*Note that the photograph coordinate system remains unchanged for the transformations of the ground coordinate system considered here.

2.4.5 Importance of Moving Coordinate Systems

Moving coordinate systems are of, at least theoretical, interest since the photographs are taken by moving cameras. The usual object in reducing the data contained in aerial photographs is to obtain a series of ground coordinates with respect to a coordinate system fixed on the ground. Nevertheless the data available are those in the photographs, and the equations of motion for the camera. Thus, at least in principle, the data reduction is equivalent to transforming ground coordinates from a system fixed on the camera to a system fixed on the ground.

Evidently, then, one may distinguish three types of coordinate systems which may be used in computing the various terms in (30), (31), and similar equations for the higher order derivatives. One type is that which is fixed on the ground. For this type, it is usually the case that

$$\frac{dC_a^m}{dt} \neq 0, \quad \frac{dX^a}{dt} = 0, \text{ and } \frac{dX_1^a}{dt} \neq 0.$$

A second type is that which is fixed on the camera. For this type, usually

$$\frac{dC_a^m}{dt} = 0, \quad \frac{dX^a}{dt} \neq 0, \text{ and } \frac{dX_1^a}{dt} = 0.$$

Finally, there is the general coordinate system type, which is moving with respect to the ground but yet not fixed on the camera. In this case, generally

$$\frac{dC_a^m}{dt} \neq 0, \quad \frac{dX^a}{dt} \neq 0, \text{ and } \frac{dX_1^a}{dt} \neq 0.$$

Similarly for the higher order derivatives of C_a^m, X^a, and X_1^a.

Furthermore, in all these ground coordinate system types

$$\frac{dx_1^m}{dt} \neq 0 \text{ except for frame type photographs,}$$

and

$$\frac{d\alpha}{dt} \neq 0 \text{ for panoramic type photographs.}$$

As an example of a somewhat similar situation one may note that the vector net effective velocity of the camera with respect to a particular ground point is given by the formula

$$V_R^a = - C_m^a \frac{d}{dt} [C_b^m (X^b - X_1^b)]. \tag{36}$$

That this formula is reasonable may be seen by noting first that it is invariant to transformations among Cartesian coordinate systems, and second that for a coordinate system fixed on the camera (36) gives

$$V_R^a = - \frac{dX^a}{dt}.$$

Thus (36) may be used to compute the camera to ground relative effective velocity in any of the three coordinate system types listed above.

III. COMPUTATION OF GROUND COORDINATES

The optical imaging equation (3) can be solved for X^a, giving:

$$X^a = X_1^a + \frac{f' \; C_m^a \; (x^m - x_1^m)}{f - \lambda_n \; (x^n - x_1^n)} . \tag{37}$$

If, however, (37) is applied to an aerial photograph, in an attempt to compute ground coordinates (X^a), then it is usually found that the magnitude of the denominator $[f - \lambda_n \; (x^n - x_1^n)]$ is so much less than the focal length f that the errors in the determination of f cause an unacceptable degree of error in the values computed for X^a. Hence (37) is not a practical formula for computing ground coordinates from the coordinates in an aerial photograph. This situation may be likened to the reverse side of the fact that an aerial photograph has a practical depth of focus which is somewhat larger than the range of relief in the terrain photographed. Therefore it is not practically meaningful to solve for the (ground) surface which is theoretically in "perfect" focus, as would be done by computing with equation (37).

It is for a somewhat similar reason that the projective equations ((4), or (5), or (11) and (12)) cannot be completely solved for X^a. In a practical sense the most which can be strictly inferred about ground points from a single aerial photograph (and the corresponding camera parameters) is their projective directions. Some additional assumption, or other information, is necessary to determine where the projected directions intersect the earth's surface. A common way out of this is to assume some approximate geometrical shape (a plane or a sphere or a spheroid) for the earth's surface. Such an assumption allows one to write a constraint equation which may be used in conjunction with the projective equations to solve for approximate ground coordinates. Another possibility, of particular interest here, is to use two overlapping aerial photographs taken with different camera locations. In this case rays may be projected from corresponding coordinates in both photos so as to triangulate ground coordinates. Both these schemes will be discussed in the following sections.

3.1 Approximate Ground Computations Based on Only One Photograph

3.1.1 Use of A Tangent Plane

Let it be assumed that the ground coordinates over some local region may be adequately approximated as lying in a geometrical plane, whose parameters are known (or may be assumed with sufficient accuracy). The necessary parameters are the direction cosines μ_a of a normal to the plane and the normal distance D_l from the plane to the camera station. It is assumed that the camera parameters C_a^m and X_l^a are known (i.e., known functions of time). Hence the quantities

$$\mu_m = C_m^a \mu_a$$

may be computed. Multiplying these quantities by both sides of equation (11) gives

$$\mu_m x^m = \mu_m x_l^m + f \cos \alpha \frac{\mu_m C_a^m (X^a - X_l^a)}{C_b^{3'} (X^b - X_l^b)}. \tag{38}$$

In (38), the quantity

$$\mu_m C_a^m (X^a - X_1^a) = \mu_a (X^a - X_1^a)$$

may be identified as the negative of the normal distance from the plane to the camera station. That is,

$$\mu_a (X^a - X_1^a) = - D_1 . \tag{39}$$

Substituting this in (38) and rearranging somewhat, then gives

$$\frac{f \cos a}{C_b^{3'} (X^b - X_1^b)} = - \frac{\mu_m (x^m - x_1^m)}{D_1} . \tag{40}$$

(40) may be substituted in (11) and the result solved for X^a:

$$X^a = X_1^a - \frac{D_1}{\mu_n (x^n - x_1^n)} C_m^a (x^m - x_1^m) . \tag{41}$$

Thus equation (41) may be used to compute coordinates of points lying in a plane surface whose normal has the direction cosines μ_a and whose normal distance from the camera location (coordinates X_1^a) is $- D_1$. Obviously the extent to which the points in this plane surface approximate actual ground coordinates depends on the accuracy with which μ_a and D_1 represent real conditions of the local ground region. If three or more ground control points (i.e., ground points with known coordinates whose images can be recognized in the aerial photograph being examined) are available, then the "best average" plane may be passed through these points. Otherwise it's usually necessary to compute a level surface tangent to some point of the geometrical model (plane, or sphere, or spheroid) assumed for the earth. The latter procedure will be discussed here - using a spherical model of the earth.

3.1.2 Tangent Plane to A Spherical Earth

In equation (41) it is assumed that, relative to some established ground coordinate system, the camera parameters C_m^a and X_1^a are known (as functions of time) and that, relative to the photograph coordinate system, the lens coordinates x_1^m are known (as functions of time - determined by the [redacted] pan sweep STAT
mechanism). Thus the coordinates x^m of any particular photo point determine a projective ray from the known camera location (X_1^a). The problem to be solved in this section is to determine μ_a and D_1 for a plane surface which is tangent to the earth (assumed spherical) at the point at which this projective ray intersects the plane. Thus the tangent plane is made a level datum plane at the point of the earth's model which is intersected by the particular projective ray, and the coordinates of the intersection point are actually approximate ground coordinates.

The solution given here for the problem stated above is one by successive approximations - suitable for iterative computation by a digital computer. Figure 3 illustrates the first two steps of the method. In Figure 3 the camera is represented as having ground coordinates X_1^a and flying height H over a spherical earth with its center at coordinates X_0^a. P_1 represents a plane tangent to the earth at the ground nadir, and X^a (1) are the coordinates of the point

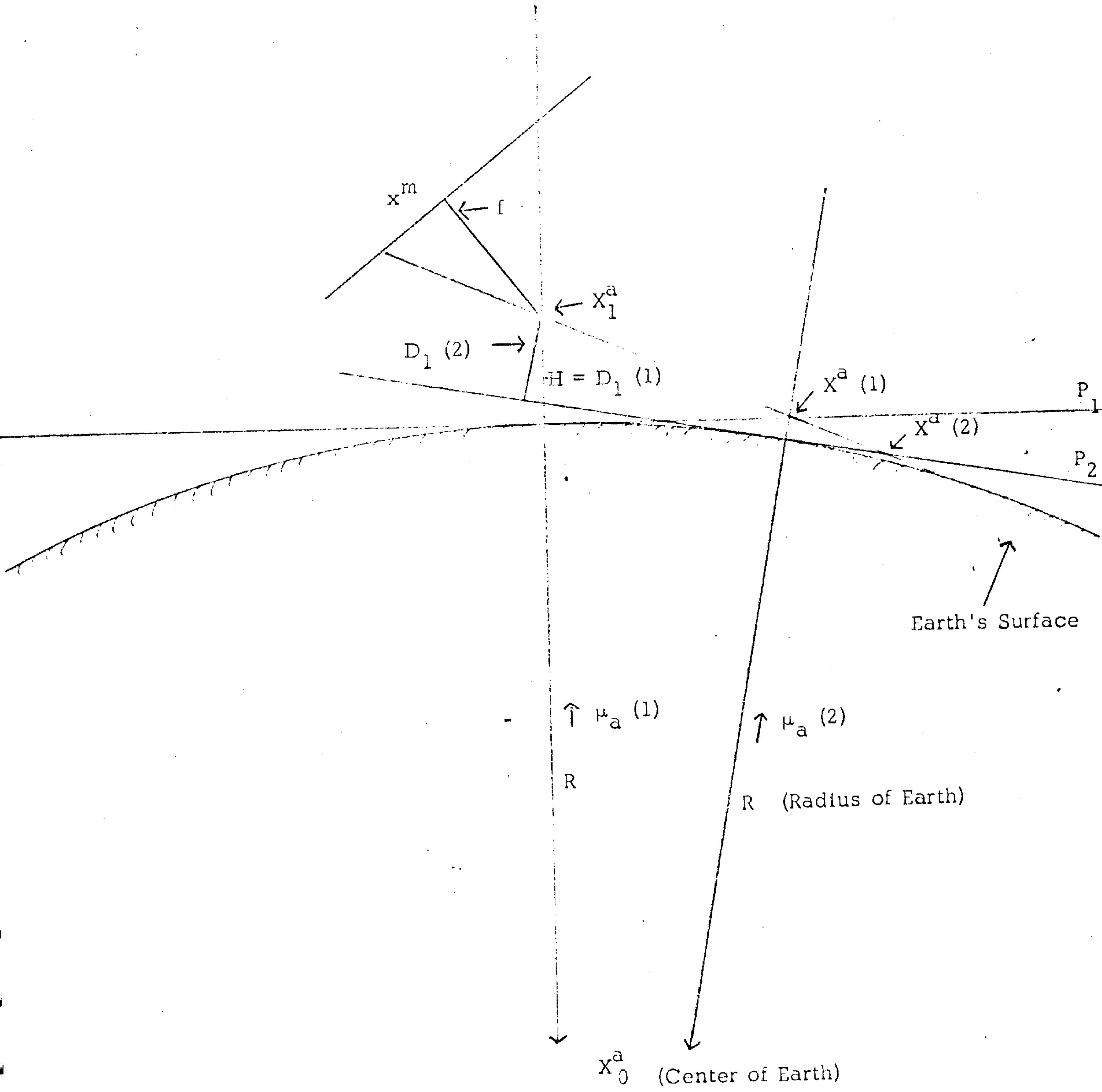


Figure 3

First two steps of iterative solution for level datum plane tangent at point intersected by a protective ray.

at which this plane is intersected by the projective ray from the photo point with coordinates x^m. Now an earth's radius vector is passed through the point X^a (1); μ_a (2) are the direction cosines of this radius vector which has, in fact, the vertical direction through the point X^a (1). P_2 is a plane tangent to the earth at the point intersected by the vertical through X^a (1). Thus μ_a (2) are the direction cosines of the normal to the level tangent plane P_2. D_1 (2) is then the normal distance from this plane to the camera station, and X^a (2) are the coordinates of the point at which this plane is intersected by the projective ray from x^m.

The third step in the solution (not shown in Figure 3) consists of passing a plane P_3 tangent to the earth at the point vertically below X^a (2) and finding the intersection X^a (3) of P_3 by the projective ray from x^m. Evidently the process may then be iterated until a plane P_N is tangent to the earth at a point as close as desired to the intersection X^a (N) of this plane by the projected ray from x^m. Thus X^a (N) is a satisfactory approximation of the intersection of the projective ray from x^m with the surface of the spherical model of the earth.

The actual computations are as follows:

1. $x^a - x^a_1 = C^a_m \, (x^m - x^m_1)$

2. $R + H = \sqrt{\delta_{ab} \, (X^a_1 - X^a_0) \, (X^b_1 - X^b_0)}$

3. $\mu_a \, (1) = \delta_{ab} \, (X^b_1 - X^b_0)/(R + H)$

4. $i = 1$

5. $D_1 (i) = [\mu_a (i)] [X_1^a - X_0^a] - R$

6. $X^a (i) = X_1^a - \dfrac{D_1 (i) (x^a - x_1^a)}{[\mu_b (i)] [x^b - x_1^b]}$ ... (e.g., (41))

7. $R_1 (i) = \sqrt{\delta_{ab} [X^a (i) - X_0^a] [X^b (i) - X_0^b]}$

8. $\mu_a (i + 1) = \delta_{ab}^{\cdot} [X^b (i) - X_0^b]/R_1 (i)$

9. If $\left| \dfrac{R_1 (i) - R}{R} \right| >$ the allowable error threshold:

increase i by 1 and repeat steps 5 through 9; otherwise the computation is complete.

The following FORTRAN subroutine was written to test the computation scheme given above (BETA is the pitch angle; roll and yaw are each taken as zero):

```
      SUBROUTINE GNDCUR
      INTEGER A, B, M
      REAL XA0(3), XA1(3), MUA(3), MUM(3), XA(3), CAM(3,3)
      REAL DLTAAB(3,3), XM(3), X1MX0(3), MUDOTX
 10   FORMAT (3F15.5)
 11   FORMAT (/3F15.5)
      READ (1,10) H, BETA, R
      READ (1,10) XM
      DO 12 A=1,3
      DO 12 B=1,3
      IF (A.EQ.B) DLTAAB(A,B) = 1.0
      IF (A.NE.B) DLTAAB (A,B) = 0.0
 12   CONTINUE
      C=COS(BETA/57.2958)
      S=SIN(BETA/57.2958)
      CAM(1,1) = -C
      CAM(1,2) = 0.0
      CAM(1,3) = -S
```

```
      CAM(2, 1) = 0.0
      CAM(2, 2) = -1.0
      CAM(2, 3) = 0.0
      CAM(3, 1) = -S
      CAM(3, 2) = 0.0
      CAM(3, 3) = C
      DO 13 A = 1,2
      XA0(A) = 0.0
13    XA1(A) = 0.0
      XA0(3) = -R
      XA1(3) = H
      RPLSH = 0.0
      DO 15  A = 1, 3
      X1MX0(A) = XA1(A)  -  XA0(A)
15    MUA(A) = 0.0
      DO 20  A = 1, 3
      DO 16 B = 1, 3
16    MUA(A) = MUA(A) + DLTAAB(A, B)*X1MX0(B)
20    RPLSH = RPLSH + MUA(A)*X1MX0(A)
      RPLSH = SQRT(RPLSH)
      DO 21 A = 1, 3
21    MUA(A) = MUA(A)/RPLSH
30    D1 = -R
      DO 31  A = 1, 3
31    D1 = D1 + MUA(A)*X1MX0(A)
      DO 35  M = 1, 3
      MUM(M) = 0.0
      DO 35  A = 1, 3
35    MUM(M) = MUM(M) + CAM(A, M) * MUA(A)
      DO 41  A = 1, 3
      XA(A) = 0.0
      MUDOTX = 0.0
      DO 40  M = 1, 3
      XA(A) = XA(A) - CAM(A, M)*XM(M)
40    MUDOTX = MUDOTX+MUM(M)*XM(M)
41    XA(A) = XA(A)*D1/MUDOTX+X1 MX0(A)
      R1 = 0.0
      DO 46 A = 1, 3
      MUA(A) = 0.0
      DO 45 B = 1, 3
45    MUA(A) = MUA(A)+DLTAAB(A, B)*XA(B)
46    R1 = R1+MUA(A)*XA(A)
      R1 = SQRT(R1)
      DO 50 A = 1, 3
      MUA(A) = MUA(A)/R1
50    XA(A) = XA(A)+XA0(A)
      WRITE (1, 11) MUA
      WRITE (1, 11) XA
      X = R1/R
      Y = ABS(1.0-X)
      IF (Y-10.0**(-6)) 55, 30, 30
```

```
     55   RETURN
          END
$0        END OF JOB
```

STAT

It may be seen that this FORTRAN program does not exactly follow all of the steps of the computational scheme given above, but is equivalent in all respects. The program was compiled and run on a computer with the allowable error threshold set at 10^{-6} (see the IF statement just before statement 55). Convergence occurred in from one to three passes (only one case required three passes - all others were less) over the iterative loop. Table I shows some of the results obtained

Table I

Some Results Obtained with Subroutine GNDCUR

	H = 15240.0	β = 15.0	R = 6378388.0
$(x^m - x_1^m)/f$ =	0.1	0.0	1.0
$\mu_a(2)$ =	0.00090	0.00000	1.00000
$X^a(1)$ =	5761.92969	0.00000	0.00000

	H = 15240.0	β = 15.0	R = 6378388.0
$x^m - x_1^m/f$ =	0.2	0.0	1.0
$\mu_a(2)$ =	0.00118	0.00000	1.00000
$X^a(1)$ =	7535.35742	0.00000	0.00000

H = 15240.0	β = 15.0	R = 6378388.0
$(x^m - x_1^m)/f$ = 0.5	0.0	1.0
$\mu_a(2)$ = 0.00212	0.00000	1.00000
$X^a(1)$ = 13514.08203	0.00000	0.00000
$\mu_a(3)$ = 0.00212	0.00000	1.00000
$X^a(2)$ = 13527.11133	0.00000	-15.00000

H = 15240.0	β = 15.0	R = 6378388.0
$(x^m - x_1^m)/f$ = 1.0	0.0	1.0
$\mu_a(2)$ = 0.00414	0.00000	0.99999
$X^a(1)$ = 26396.42578	0.00000	0.00000
$\mu_a(3)$ = 0.00415	0.00000	0.99999
$X^a(2)$ = 26491.28125	0.00000	-55.00000

H = 15190.0	β = 15.0	R = 6378438.0
$(x^m - x_1^m)/f$ = 0.1	0.0	1.0
$\mu_a(2)$ = 0.00090	0.00000	1.00000
$X^a(1)$ = 5743.02539	0.00000	0.00000

H = 15190.0	β = 15.0	R = 6378438.0
$(x^m - x_1^m)/f$ = 0.2	0.0	1.0
$\mu_a(2)$ = 0.00118	0.00000	1.00000
$X^a(1)$ = 7510.63477	0.00000	0.00000

	H = 15190.0	β = 15.0	R = 6378438.0
$(x^m - x_1^m)/f$ =	0.5	0.0	1.0
$\mu_a(2)$ =	0.00211	0.00000	1.00000
$X^a(1)$ =	13469.74609	0.00000	0.00000
$\mu_a(3)$ =	0.00211	0.00000	1.00000
$X^a(2)$	13482.60352	0.00000	-15.00000

	H = 15190.0	β = 15.0	R = 6378438.0
$(x^m x_1^m)/f$ =	1.0	0.0	1.0
$\mu_a(2)$ =	0.00412	0.00000	0.99999
$X^a(1)$ =	26309.82422	0.00000	0.00000
$\mu_a(3)$ =	0.00414	0.00000	0.99999
$X^a(2)$ =	26406.91406	0.00000	-57.00000

	H = 15240.0	β = 20.0	R = 6378388.0
$(x^m - x_1^m)/f$ =	0.1	0.0	1.0
$\mu_a(2)$ =	0.00115	0.00000	1.00000
$X^a(1)$ =	7337.97852	0.00000	0.00000

	H = 15240.0	β = 20.0	R = 6378388.0
$(x^m - x_1^m)/f$ =	1.0	0.0	1.0
$\mu_a(2)$ =	0.00512	0.00000	0.99999
$X^a(1)$ =	32682.25391	0.00000	0.00000
$\mu_a(3)$ =	0.00515	0.00000	0.99999
$X^a(2)$ =	32863.65625	0.00000	-85.00000

	H = 30480.0	β = 15.0	R = 6378388.0
$(x^m - x_1^m)/f$ =	0.1	0.0	1.0
$\mu_a(2)$ =	0.00181	0.00000	1.00000
$X^a(1)$ =	11523.48437	0.00000	1.00000
$\mu_a(3)$ =	0.00181	0.00000	1.00000
$X^a(2)$ =	11527.97070	0.00000	-11.00000

	H = 30480.0	β = 15.0	R = 6378388.0
$(x^m - x_1^m)/f$ =	1.0	0.0	1.0
$\mu_a(2)$ =	0.00828	0.00000	0.99997
$X^a(1)$ =	52791.14062	0.00000	1.00000
$\mu_a(3)$ =	0.00834	0.00000	0.99997
$X^a(2)$ =	53174.14844	0.00000	-221.00000

3.1.3 Plane Tangent at the Nadir

From Figure 3 and from the computational scheme given above, it may be seen that the first pass over the iterative loop corresponds to the plane P_1 which is tangent at the ground nadir.* The FORTRAN subroutine may be seen to be based on a ground coordinate system with its origin at the nadir and its Z axis vertical. Thus in Table I the $X^1 = X$ and $X^2 = Y$ results are in the plane P_1 and the $X^3 = Z$

---

*Since the camera is moving, the term ground nadir is possibly ambiguous; here it is used to mean a fixed ground point which is vertically below the camera lens at some defined instant of time.

Approved For Release 2003/01/28 : CIA-RDP78B05171A000600030001-0

results are perpendicular to P_1. If only the coordinates in P_1 are considered then Table I shows that the values obtained on the first pass are within a fraction of a percent of the final values - for the cases computed. Thus it is sometimes sufficiently accurate to compute approximate (horizontal) ground coordinates by treating the earth's surface as a level plane through the nadir.

For a level plane through the nadir and with the ground coordinate system having its X^3 axis vertical equation (41) may be put in a more familiar form. In this case $D_1 = X_1^3 - X^3$ and $\mu_a = (0, 0, 1)$. Hence

$$\mu_m = C_m^a \, \delta_a^3 = C_m^3,$$

and

$$X^a = X_1^a + \frac{C_m^a \, (x^m - x_1^m)}{C_n^3 \, (x^n - x_1^n)} \, (X^3 - X_1^3). \qquad (42)$$

It is common practice to write (42) with $X^3 - X_1^3$ set equal to $-(H-h)$ where H is the flying height and h is the ground elevation at the nadir - both with respect to some datum (as mean sea level). Table I includes cases which illustrate the effect of varying the ground elevation by 50 meters.

3.1.4 Treatment of the Photograph Coordinates for the Different Types of Photography

Equations (41) and (42) give the ground coordinates as functions of the rectangular components of the displacement of the photograph point from the instantaneous position of the camera lens $(x^m - x_1^m)$. The relations of this displacement vector to actual photograph measurements are different for the three different types of photography.

3.1.4.1 Frame Type Photography

In analyzing frame photographs it is usual practice to neglect the finite time required for exposure, and to consider the entire photograph to be exposed in the same instant of time (thus also neglecting the scanning time for the focal plane shutter - if one is used). Hence both the position of the camera with respect to the ground and the position of the lens with respect to the film are the same for all points in the photograph. In other words x_1^m, C_m^a, and X_1^a are treated like constants (i.e., constant at the particular values they have at the instant of exposure). Thus the displacement vector $(x^m - x_1^m)$ is related to the actual photograph measurements simply by translation and/or rotation of the measurement coordinate system into the photograph coordinate system. Both of these coordinate systems are usually taken with the $x^{3'}$ (z) axis normal to the photograph - hence $(x^{3'} - x_1^{3'}) = f$ (the camera focal length), and the coordinate transformation (if any) is effectively a two dimensional transformation. Thus the displacement vector may be stated as:

$$x^m - x_1^m = C_\alpha^m (x^\alpha - x_{10}^\alpha) \qquad (43)$$

where C_α^m, and x_{10}^α are constants with $x^{3'} - x_{10}^{3'} = f$, and x^α are the photograph measurements. The lens coordinates x_{10}^α are usually determined from fiducial marks on the photograph.

In discussions from this point on the distinction between the measurement (i.e., comparator stage) coordinate system and the photograph coordinate system will be neglected unless there is particular reason to discuss it, and photograph coordinates will be treated as though they were measured directly. Hence (43) will be stated simply as:

$$x^m - x_1^m = x^m - x_{10}^m = (x^{1'} - x_{10'}^{1'},\ x^{2'} - x_{10'}^{2'},\ f) \tag{44}$$

3.1.4.2 Strip Type Photography

Strip type photographs are exposed by having a narrow slit, with its long dimension extending across the width of the film and perpendicular to the edges, scan in a direction parallel to the edges of the film which are approximately parallel to the direction of flight. The scanning speed is coordinated with the flight speed so the image on the film is as nearly stationary as is practical. Thus the projected ray from a strip camera lies in a plane which bears a fixed angular relation to the camera and which includes the scanning slit. The angular relation is stated as the tilt of the plane with respect to the normal to the film, which is here called β. This tilt angle (β) is normally fixed, for any particular strip camera, by the camera design and is zero for systems currently of interest.

Throughout this and following discussions the following conventions will be assumed for examining (i.e., measuring) processed photographs:

a. For cameras which do not produce a mirror type reversal: negatives viewed emulsion side down (away from observer) and positives viewed emulsion side up (toward observer).

b. For cameras which do produce a mirror type reversal: negatives viewed emulsion side up and positives viewed emulsion side down.

In all cases the photograph coordinate system (right handed Cartesian) for strip type photos will be assumed to have the x axis approximately in the direction of flight (i.e., the direction of the camera motion with respect to the ground), the y axis parallel to the long dimension of the scanning slit, and the z axis upward normal to the film plane. Analysis shows that, under these assumptions, the displacement vector for strip photos has the following components (to be substituted in (41) or (42):

$$x^m - x_1^m = (f \tan \beta,\ x^{2'} - x_{10}^{2'},\ f). \qquad (45)$$

Since (45) does not include the x photograph coordinate, it is not sufficient to merely substitute (45) in formulae for ground coordinates. The effect of the x photo coordinate is to enable determination of the time at which a particular point of the photo was exposed. This value of time must then be used to determine the corresponding instantaneous values of X_1^a and C_m^a (which are time functions). Strip photos usually have time tics along the edge to facilitate determination of the actual time at which any particular point was exposed. Thus (41) and (42) are applied in a slightly different fashion for strip photos than for frame type photos.

3.1.4.3 Panoramic Type Photography

Panoramic cameras expose the film by a slit which scans around half of a circular cylinder to which the film is clamped. The axis of the cylinder is approximately parallel to the flight direction, and the scan is from high oblique on one side to vertical to high oblique on the other side. The displacement vector, with <u>rectangular</u> components $(x^m - x_1^m)$ is from the lens (on the cylinder axis) to the cylindrical surface of the film emulsion. The photo coordinate system (right hand Cartesian) will here be taken with the $x^{1'}$ axis parallel to the axis of the cylinder and approximately in the flight direction, the $x^{2'}$ axis approximately horizontal and to the left, and the $x^{3'}$ axis upward normal to an imaginary plane which is tangent to the "top" of the cylinder. The scanning angle α $(= \omega t)$ is positive left handed about the $x^{1'}$ axis, and zero at the $x^{3'}$ axis. The STAT velocity is assumed proportional to cos α.

When laid out flat, for measurement, a panoramic photograph exhibits its $x^{1'}$ axis unchanged but has the original $x^{2'}$ and $x^{3'}$ axes combined into one - which is here called $y - y_{10}$. Thus (see section 2.3.2):

$$y - y_{10} = f\,\alpha = f \tan^{-1} \left(\frac{x^{2'} - x_1^{2'}}{x^{3'} - x_1^{3'}}\right). \tag{46}$$

Approved For Release 2003/01/28 : CIA-RDP78B05171A000600030001-0

Putting the above stated definitions together and solving for the rectangular components then leads to

$$x^m - x_1^m = \left(x^{1'} - x_{10}^{1'} - \frac{V_M}{\omega}\sin\frac{y-y_{10}}{f},\ f\sin\frac{y-y_{10}}{f},\ f\cos\frac{y-y_{10}}{f}\right), \quad (47)$$

where V_M is the maximum rate [redacted] (a negative number in the conventions used here).

STAT

Equation (47) is to be substituted in formulae for ground coordinates, such as (41) and (42), when these are applied to panoramic photographs. As with strip photography it is also necessary to evaluate the time functions C_m^a and X_1^a for the particular time at which any given photo point was exposed. For pan photos this time is given (at least approximately) by

$$t = \frac{\alpha}{\omega} = \frac{y - y_{10}}{f\,\omega}. \quad (48)$$

(48) is therefore to be substituted in the time functions assumed for C_m^a and X_1^a whenever (47) is used in a formula for ground coordinates.

3.2 Stereoscopic Triangulation of Ground Points

In any aerial photograph, at the instant of exposure, each photo point lies on a ray from some particular ground point through the camera lens ("point of perspective"). (This ray is here assumed straight - since atmospheric refraction is being neglected.) If two photographs are taken with different camera locations but including coverage of the same ground region then corresponding points (as they are called here) in the two photos, together with their common ground point, define a plane which includes the instantaneous positions of the two points of perspective. The line segment joining the instantaneous points of perspective will henceforth be referred to as the "airbase" (i.e., the particular airbase associated with a particular pair of corresponding points). Hence two overlapping photographs, together with data stating the positions and orientations of their respective cameras, may be used to compute ground coordinates by triangulation.

Equation (42) was written as giving ground coordinates when certain assumptions are justified. Under somewhat more general conditions, however, (42) defines a projective ray from the instantaneous camera position.* Thus, if two photographs are located and oriented as they were originally exposed then projective rays from corresponding points must intersect at the common ground point. An equation equivalent to (42) but stated for the other photograph (with photo coordinate system axes designated by x^r, x^s, $\cdots$) is

$$X^a = X_2^a + \frac{C_r^a\,(x^r - x_2^r)}{C_s^3\,(x^s - x_2^s)}\,(X^3 - X_2^3). \tag{49}$$

Equating (42) and (49) gives

$$X_1^a + \frac{C_m^a\,(x^m - x_1^m)}{C_n^3\,(x^n - x_1^n)}\,(X^3 - X_1^3) = X_2^a + \frac{C_r^a\,(x^r - x_2^r)}{C_s^3\,(x^s - x_2^s)}\,(X^3 - X_2^3). \tag{50}$$

---

*Note that in this sense (42) is valid even if the X^3 axis is not restricted to have the vertical direction.

Multiplying by the product of the two denominators then gives

$$(C^a_m C^3_r - C^3_m C^a_r)(x^m - x^m_1)(x^r - x^r_2) X^3 = C^a_m C^3_r (x^m - x^m_1)(x^r - x^r_2) X^3_1 - C^3_m C^a_r (x^m - x^m_1)(x^r - x^r_2) X^3_2 + C^3_m C^3_r (x^m - x^m_1)(x^r - x^r_2)(X^a_2 - X^a_1). \tag{51}$$

Now set the index a = 1 and solve (51) for X^3:

$$X^3 = \frac{[C^3_m C^3_r (X^1_2 - X^1_1) - (C^3_m C^1_r X^3_2 - C^1_m C^3_r X^3_1)](x^m - x^m_1)(x^r - x^r_2)}{(C^1_n C^3_s - C^3_n C^1_s)(x^n - x^n_1)(x^s - x^s_2)} \tag{52}$$

Hence:

$$X^3 - X^3_1 = \frac{C^3_m [C^3_r (X^1_2 - X^1_1) - C^1_r (X^3_2 - X^3_1)](x^m - x^m_1)(x^r - x^r_2)}{(C^1_n C^3_s - C^3_n C^1_s)(x^n - x^n_1)(x^s - x^s_2)} \tag{53}$$

Combining (42) and (53) results in

$$X^a = X^a_1 + \frac{C^a_m [C^3_r (X^1_2 - X^1_1) - C^1_r (X^3_2 - X^3_1)](x^m - x^m_1)(x^r - x^r_2)}{(C^1_n C^3_s - C^3_n C^1_s)(x^n - x^n_1)(x^s - x^s_2)} \tag{54}$$

Similarly, using (52) to find $X^3 - X^3_2$ and combining the result with (49) results in

$$X^a = X^a_2 + \frac{C^a_r [C^3_m (X^1_2 - X^1_1) - C^1_m (X^3_2 - X^3_1)](x^m - x^m_1)(x^r - x^r_2)}{(C^1_n C^3_s - C^3_n C^1_s)(x^n - x^n_1)(x^r - x^r_2)}. \tag{55}$$

Evidently (54) gives ground coordinates based on X^a_1, the coordinates of the first camera station, and (55) gives ground coordinates based on X^a_2, the coordinates of the second camera station. Both (54) and (55) involve the orientations of both camera stations, the flight base, and the photo coordinates of corresponding points in the two photographs. If there were no errors, in measurement or in computation, and if there were no approximations involved, then (54) and (55) would be expected to give the same values for ground coordinates. Since there generally are errors (54) and (55) may be expected to give slightly different values and presumably the averages of the two are the "best" values for ground coordinates.

Equations (54) and (55) involve the displacement vectors $(x^m - x_1^m)$ and $(x^r - x_2^r)$. Substitutions for these vectors are to be made according to the method stated in section 3.1.4. Section 3.1.4 specifically states the method for the first photograph but can easily be modified to state it also for the second photograph. All that's necessary is to replace the various indices for the first photo with corresponding indices for the second photo. The details will not be given here.

IV. AUTOMATIC STAGE TRACKING IN THE OPERATION OF THE STEREOCOMPARATOR

The preceding chapter brings out the fact that for accurate computation of ground coordinates from aerial photographs it's at least desirable and perhaps actually necessary to use two overlapping photographs taken with different camera locations. Reduction of the data in these overlapping photographs involves measuring the photo coordinates of pairs of points in the two photos which correspond, in each case, to the same ground point. To facilitate this operation the Stereocomparator is designed to automatically adjust its measuring stages and its various optical elements so as to enable viewing selected regions of the two photos in stereo. The method by which the two stages are maintained on approximately corresponding points will be discussed in this chapter. The method for controlling the optical system will be given later.

Before details of the stage tracking are given, two other matters will be stated. One is that the primary functions of the Stereocomparator involve measurement of the coordinates of corresponding points and output of the resulting digital information, but not further reduction of this information. Computations (for example, ground coordinates) based on the digital output of the Stereocomparator are performed by a computer which is external to the Stereocomparator (not by the control computer which is part of the Stereocomparator). The other matter has to do with the precision of automatic stage tracking. This is intended to be sufficient for comfortable stereo viewing but not necessarily sufficient for automatic digitization of corresponding points. That is, it is expected that the operator will, in general, perform the final setting of the stages to precisely corresponding points - after the automatic tracking has set them approximately on the desired points.

A joystick and two trackballs are provided for use by the operator in directing the two stages to desired points. Pushbuttons are available which permit selection of various modes wherein the joystick or either trackball controls either stage or both stages together. Probably the most frequently selected mode will be one in which the joystick controls both stages together, and each trackball controls one stage independently. In this mode the operator directs both stages to various selected areas with the joystick and performs final setting on precisely corresponding points with the two trackballs. Under these conditions the Stereocomparator, unless inhibited by the operator having selected a non-tracking mode, performs automatic stage tracking when directed by the joystick and temporarily discontinues automatic stage tracking when directed by either trackball. No alteration of the pushbutton setting is required to produce switching in or out of automatic tracking, since such switching is automatically performed, as required, by deflection of the joystick or either trackball. If the joystick is standing at its neutral position and neither trackball is being rotated then both stages remain stationary as last directed by the operator.

Approved For Release 2003/01/28 : CIA-RDP78B05171A000600030001-0

The Stereocomparator automatic tracking system can be described as a master-slave type of control system. That is, the operator, in effect, directs one stage (the master) to a desired point, and the Stereocomparator electronics direct the other (i.e., slave) stage to the corresponding point (approximately). To the operator, however, it appears that both stages move simultaneously to corresponding points - since the time delay in servoing the slave stage to the master is so small as to be unnoticeable. Somewhat similarly, the optics control is by a master-slave type of system, but this will be discussed later. Two tracking modes are available for selection by the operator, which function - one with and one without - an opto-electronic correlation system. These will be described separately.

4.1 Automatic Without Correlator Tracking Mode

4.1.1 Operations Performed

Automatic tracking in this mode is initiated by the operator directing the stages successively to three different points and manually establishing a stereo model at each of these three points. In each case the operator depresses the REORIENT button after having established a stereo model.

Depressing the REORIENT button causes the internal control computer to read the stage coordinates STAT for both stages, transform these into the respective photo coordinate systems and substitute in equation (54), thus obtaining X, Y, Z coordinates for one model point. When three model points have been thus obtained the control computer, in effect, passes a geometric plane through these three points. That is, it computes the direction cosines for the normal to such a plane and then computes the normal distance from the plane to the first camera station. A plane thus determined will hereafter be referred to as a "tracking plane."

Once a tracking plane has been established, then tracking motion (i.e., stage motion such that the floating dot appears to remain in the tracking plane) may be directed by the joystick. Motion of the floating dot out of the tracking plane (for example, to the top of a building if the tracking plane corresponds to the ground level) may be directed by the trackballs.

4.1.2 Computation of Slave Stage Coordinates

The [redacted] computer controls stage tracking by periodically reading the coordinates of the master stage and computing corresponding coordinates for the slave stage. For this purpose equation (41) is used to compute model coordinates in the tracking plane. Equations (11) and (12) are then used to compute the slave stage coordinates. Equation (11) for the second photograph is written as:

$$x^r = x_2^r + f_2 \cos \alpha_2 \frac{C_a^r (X^a - X_2^a)}{C_b^{3''} (X^b - X_2^b)} \qquad (56)$$

Equation (41) is used to obtain the following equation for $X^a - X_2^a$:

$$X^a - X_2^a = - \frac{D_1}{\mu_n (x^n - x_1^n)} C_m^a (x^m - x_1^m) - (X_2^a - X_1^a)$$

$$= - \frac{D_1}{\mu_n (x^n - x_1^n)} [C_m^a + \frac{\mu_m}{D_1} (X_2^a - X_1^a)] (x^m - x_1^m)$$

Hence (56) becomes

$$x^r = x_2^r + f_2 \cos \alpha_2 \frac{C_a^r Y_m^a (x^m - x_1^m)}{C_b^{3''} Y_n^b (x^n - x_1^n)}; \tag{57}$$

wherein

$$Y_m^a = C_m^a + \frac{\mu_m}{D_1} (X_2^a - X_1^a). \tag{58}$$

The angle α_2 is given by

$$\alpha_2 = \begin{cases} 0 & ; \quad \text{frame type slave photo} \\ 0 & ; \quad \text{strip type slave photo} \\ \tan^{-1} \dfrac{C_a^{2''} Y_m^a (x^m - x_1^m)}{C_b^{3''} Y_n^b (x^n - x_1^n)} & ; \quad \text{pan type slave photo} \end{cases} \tag{59}$$

Equations (57), (58), and (59) give slave photo coordinates as functions of master photo coordinates, utilizing the parameters μ_m and D_1 of the tracking plane and the camera station parameters for both cameras. Since the latter are functions of t_1 and t_2, the times at which the corresponding master and slave points were exposed, it is necessary to determine these times. The time t_1 for the master photo is determined from the master photo coordinates as described in section 3.1.4. This method cannot be used for the slave photo, however, since the object at this point is to compute slave photo coordinates - hence the latter must be treated as unknown until after t_2 has been determined.

4.1.3 Computation of Slave Photo Point Time of Exposure

The method of solving for the time of exposure of the unknown slave photo point is somewhat similar to that given in section 2.4.2. Each camera type imposes a constraint on its scanning and/or exposure which is peculiar to its design. This constraint may be combined with the flight equations to write a function of time which can be equated to zero. The resulting equation may then be solved for the particular value of time for which it is satisfied.

An iterative scheme, known as the Newton-Raphson method, is used to solve the equation. This method assumes that a first approximation is known for the solution and enables computing a correction to this - thus yielding a second approximation. The method is then successively repeated to give third, fourth, etc., approximations as far as is desired. Thus the equation

$$f(t) = 0$$

may have its left side approximated by the first two terms of its series expansion:

$$f(t_0) + f'(t_0)(t - t_0) = 0.$$

Solving this for t:

$$t = t_0 - \frac{f\ (t_0)}{f'\ (t_0)}\ . \tag{60}$$

Hence if f(t) and its first derivative f'(t) are stated explicitly and t_0 is an approximate solution of some order then t, as given by (60), is an approximate solution of order one greater. Thus (60) may be used iteratively to determine a value of t which makes f(t) as close as desired to zero. Application of this method to the different types of photography is described below.

4.1.3.1 Frame Type Photos

The idealized frame camera constrains all of its points to be exposed simultaneously and instantaneously. The time of exposure must be stated as part of the auxiliary input data, and no additional computation is necessary.

4.1.3.2 Strip Type Photos

The strip camera scanning slit causes a constraint which was stated as equation (21). Re-expressing (21) for the slave photo and arranging the result in a slightly different form gives

$$[C_a^{1''} - C_a^{3''}\ \tan \beta_2]\ (X^a - X_2^a) = 0 \tag{61}$$

Equation (61) is a function of time t_2 since C_a^r and X_2^a are functions of time - stated by the equations of flight. Hence (61) may be solved by the method of equation (60). That is

$$t = t_0 - \frac{(C_a^{1''} - C_a^{3''} \tan \beta)\,(X^a - X_2^a)}{\frac{d}{dt}\,[(C_b^{1''} - C_b^{3''} \tan \beta_2)\,(X^b - X_2^b)]} \tag{62}$$

wherein the second term on the right is evaluated at a particular approximation t_0 for the solution of (61), and t is hence the next (better) approximation. Thus (62) is used iteratively until successive values of t cease to show significant change from one to the next. This can only occur when the numerator of the second term is sufficiently near zero. Hence (61) is adequately satisfied by the final value of t.

4.1.3.3 Panoramic Type Photos

The last statement of (59) may be used as a constraint to solve for the slave photo time of exposure. Time t_2 enters (59) by C_a^r (i.e., $C_a^{2''}$ and $C_a^{3''}$) and Y_m^a (X_2^a). In this case the formal scheme of (60) can be somewhat simplified, since it is assumed that

$$\alpha_2 = \omega_2\,(t_2 - t_{20})$$

where ω_2 is a constant (stated in the input data) and t_{20} is the value of time at which the scan angle α_2 is zero. Thus if t is an approximation of some order for the value of time which satisfies (59) then the next better approximation t_2 is given by

$$t_2 = t_{20} + \frac{1}{\omega_2} \tan^{-1} \frac{C_a^{2''}\ Y_m^a\ (x^m - x_1^m)}{C_b^{3''}\ Y_n^b\ (x^n - x_1^n)}\ . \tag{63}$$

wherein C_a^r and Y_m^a are evaluated at the time t. The value t_2 is then used to re-evaluate C^r_a and Y^a_m, and (63) is used again - to obtain the next better approximation. Thus (63) may be used iteratively until successive values of t_2 cease to show significant change from one to the next. The final value of t_2 is hence the time of exposure of a point of the slave photo corresponding to the master photo point whose coordinates are x^m (as substituted in (63)). This value of time is then substituted in (57) and in (27)* to give the predicted slave photo coordinates.

4.1.4 Modified Computations for Real-Time Stage Control

The Stereocomparator design specifications require that, for smooth stage motion, the control computer must output stage drive commands at the rate of 120 per second. Each such drive command consists of incremental values of x and y (either of which may be zero) for each stage. The two stages attempt to follow each incremental drive command, but the frequency response of the stage servo systems is much below 120 hertz. Hence the incremental (step) nature of the computer outputs is very much smoothed out in the actual stage motion.

In order to satisfy the requirement for 120 cycles of input/output transfer per second the control computer program is separated into two parts - called the real time and the non-real time programs respectively. The real time program includes the I/O transfers and performs only the minimum amount of computing which is necessary -

*That is (27) rewritten for the slave photograph.

according to the scheme which is described below. The non-real time program does the bulk of the computing and feeds results to the real time program. Since the non-real time program requires considerably more than (1/120) second for complete execution, it is interrupted 120 times a second and transfers control to the real time program. The latter requires much less than (1/120) second for one complete cycle and, after executing one cycle, returns control to the non-real time program at the location from which the latter was interrupted. The non-real time program is executed repetitively - requiring about 1 or 2 seconds for each cycle of execution.

As a means of separating the computations into those which must be performed in real time and those which may be done in non-real time a number of the formulae are expanded as [redacted] STAT series. This enables computing the coefficients (parameters) in non-real time and evaluating the series each time the real time program cycles. The frequency of cyling the real time program ensures that the variations between any two successive cycles are small; hence only the zero and first order terms are retained for each series.

The aerial photographs which are measured by the Stereocomparator, generally speaking, have appreciable amounts of distortion due to tilt and motion* of the camera during exposure. To enable viewing two distorted pictures in stereo the Stereocomparator optical system introduces automatic compensation for the distortion

*The [redacted] mechanism minimizes multiple exposure but still permits geometric distortion. STAT within any particular field of view. The means for doing this will be described later, but the point here is that the distortion correcting features of the optical system also affect the way the images in the eyepieces appear to move when the stages are moved. Hence the control computer is programmed so that when the operator deflects the joystick or one of the trackballs in a certain direction and by a certain amount then the stages are commanded to move in such a direction and at such a rate that <u>the images in the eyepieces</u> move in the directions and by the amounts directed by the control which was deflected.

Motion of the images may be represented by the two dimensional vector Δx^j, which will apply to either image or to both images simultaneously as is appropriate in any particular discussion. Motion of the master and slave stages will be represented by the two dimensional vectors Δx^m and Δx^r respectively. Thus the operator deflects the joystick or one of the trackballs in a direction and by an amount corresponding to the vector Δx^j (with x and y but not z components). The computer must determine the vectors Δx^m and Δx^r and output corresponding commands to the two stages.

Evidently for master-slave stage tracking Δx^j corresponds to deflection of the joystick and the above mentioned vectors must be related by the expressions

$$\Delta x^m = X^m_j \, \Delta x^j \tag{64}$$

and

$$\Delta x^r = X^r_m \, \Delta x^m \tag{65}$$

where X_j^m and X_m^r are 2 x 2 matrices to be determined by the computer. The matrix X_j^m is the reciprocal of the master optics transformation matrix which will be described later. The matrix X_m^r is called the tracking matrix and is derived, basically, from equation (57). Thus the computer operates to satisfy (64) so as to move the master stage as required to produce motion of the master image as directed by the operator. Simultaneously the computer operates to satisfy (65) so as to move the slave stage as required to maintain the two photos with corresponding points at (or at least close to) the respective optical axes. Hence the floating dot appears to move as directed by the operator and to remain in the tracking plane.

For non-tracking motion of the stages (64) still applies to the master stage but control of the slave stage must be according to the expression

$$\Delta x^r = X_j^r \Delta x^j \qquad (66)$$

where X_j^r is the reciprocal of the slave optics transformation matrix. In this case Δx^j for one image corresponds to deflection of one trackball and Δx^j for the other image corresponds to deflection of the other trackball. Discussion of (64) and (66) will be deferred until later. At this point computation of X_m^r, which appears in (65), will be discussed.

The [redacted] series expansion of equation (57) STAT
may be derived by a method similar to that given in section 2.4. The

result is

$$x^r = x_0^r + X_m^r \, \Delta x^m + \frac{1}{2} X_{mn}^r \, \Delta x^m \, \Delta x^n + \cdots \tag{67}$$

with

$$X_m^r = \frac{\partial x^r}{\partial x^m} + \frac{dx^r}{dt} \frac{\partial t}{\partial x^m}, \tag{68}$$

$$X_{mn}^r = \frac{\partial X_m^r}{\partial x^n} + \frac{dX_m^r}{dt} \frac{\partial t}{\partial x^n}, \text{ etc.} \tag{69}$$

and $\frac{\partial t}{\partial x^m}$ determined from the appropriate constraint equation. Only the first two terms on the right side of (67) will be retained in the computations, however. Equation (57) - hence also (67) - is basically a <u>three</u> dimensional relation between two photographs both of which are <u>two</u> dimensional. Hence if both photograph coordinate systems are oriented with their Z axes normal to the respective photographs then the Z axis components in (67) may be ignored. Thus, neglecting higher order terms, (67) may be written as

$$\Delta x^r = x^r - x_0^r = X_m^r \, \Delta x^m$$

and the 2 x 2 (x, y) portion of X_m^r is seen to be the same as the tracking matrix in (65).

From (57), (58), and (59) it may be seen that the explicit formula for the derivatives $\frac{dx^r}{dt}$ is quite complex. Consequently the non-real time program does not use (68) to compute the tracking matrix (as would be the theoretically correct procedure).

Approved For Release 2003/01/28 : CIA-RDP78B05171A000600030001-0

Instead it uses an approximate method based directly on (57). Equation (57) may be written symbolically:

$$x^r = f^r\ (x^m). \tag{70}$$

For any increment Δx^m, (70) may be written in the form

$$x^r + \Delta x^r = f^r\ (x^m + \Delta x^m),$$

i.e.,

$$\Delta x^r = f^r\ (x^m + \Delta x^m) - f^r\ (x^m). \tag{71}$$

Thus (57), in the form of (71), may be used to compute two values of Δx^r corresponding to two arbitrarily chosen values of Δx^m. These may then be substituted in (65) and the resulting equations solved for X^r_m. If the values chosen for Δx^m are smaller than the diameter of the largest portion of a photograph over which X^r_m may be considered constant and not so small as to cause excessive truncation errors in computing (71) then the resulting values for X^r_m should be approximately correct. As tracking proceeds the values of x^m and x^r will change and X^r_m will have to be recomputed - hence the non-real time program runs repetitively. For accuracy and convenience the non-real time program computes with floating point arithmetic.

Thus the computing which is left for the real time program - with regard to stage tracking - is to evaluate (64) and (65) or (66), using values of X^m_j, X^r_m, and X^r_j supplied by the non-real time program. Only a few multiplications and additions are needed, and, for maximum speed, these are performed in fixed point arithmetic. The values of Δx^j are obtained by reading the latest position of the joystick and the trackballs. The computed values of Δx^m and Δx^r are transferred to the master and slave stage servo systems, and the stages move accordingly.

In general, it may be expected that stage tracking, as described above, will have a limited range over which it may proceed without introducing appreciable non-correspondence. Thus when operating in the automatic without correlator mode it may be expected that new tracking planes will have to be established from time to time. The three point procedure described earlier may, of course, be repeated whenever desired. However another procedure is also available if a tracking plane has been previously established. This consists of manually establishing a stereo model on some one point and depressing the REORIENT button, and then immediately using the joystick to re-establish the tracking mode. Under these conditions the computer will compute a new tracking plane through the desired point and parallel to the previously established tracking plane.

4.2 Automatic With Correlator Tracking Mode

The opto-electronic correlator consists of two image disector type TV camera tubes and electronics for controlling the beam deflections, correlating the video signals, and computing 6 analog output signals. The image disector tubes are mounted so as to "see" essentially the same views as the operator's two eyes.

Approved For Release 2003/01/28 : CIA-RDP78B05171A000600030001-0

The 6 output signals are derived from the video correlation in such a way as to represent approximate measures of 6 respects in which the two images differ from one another. These 6 respects are called: x displacement, y displacement, x scale factor, y scale factor, x skew factor, and y skew factor. Thus two of the output signals (x and y displacement) may be used to aid stage tracking. The other four output signals are used to aid in setting the optical system - as will be described later.

Supplementary to the prime functions described above, the opto-electronic correlator (called the "Image Analysis System" or IAS) also outputs a digital signal showing whether it can or cannot satisfactorily correlate the video from the images which it is seeing at any particular time. Some minimum degree of detail, contrast, and brightness are needed for satisfactory correlation, but also it is necessary that the two images be within the "pull-in" range in each of the 6 respects. Whenever the IAS is called for by the operator's mode selection but is nevertheless unable to satisfactorily correlate then it outputs a digital signal which operates an indicator light on the Stereocomparator control console and tells the control computer to disregard the IAS output lines.

In initiating operation in the Automatic With Correlator mode the operator manually establishes a stereo model at some one pair of corresponding points. This may be done somewhat approximately since the IAS will automatically take hold as soon as the two images are brought within the pull-in range in all 6 respects. Once the IAS has taken hold then the operator may direct tracking or non-tracking operation by way of the joystick or trackballs as was described for the Automatic Without Correlator mode.

Computer control with the aid of the IAS differs in two respects from the scheme which was described in section 4.1. In the first place the tracking plane is established by the computer with the aid of the correlator, without any need for manual setting on three points. For this purpose a plane tangent to the "spherical" earth is established by the method of section 3.1.2 and parallel shifted by an automatic one point reorient so as to permit best correlation. In the second place when the slave stage vector is computed by the equivalent of equation (65) then the IAS x, y displacement outputs are used to compute a correction term. Equations (64) and (66) are, however, used just as before.

The IAS images are essentially the same as those seen by the operator - hence the IAS coordinate system is logically the same as that used for the images. In other words the IAS x, y displacement signals may be treated as components of the two dimensional vector Δx^j_c. Consequently for stage tracking with the aid of the IAS, equation (65) is replaced by

$$\Delta x^r = X^r_m \, \Delta x^m + X^r_j \, \Delta x^j_c \tag{72}$$

with X^r_m and X^r_j the same as defined in section 4.1.4. Thus the correlator signals are applied as corrections to the slave stage only, rather than being divided between both stages. This is done because it results in simpler stability conditions for the servo systems.

4.3 Digital Integration of the Control Commands

Although equations (64), (65) or (72), and (66) are used, as has been described, to compute the required increments of stage motion, the actual outputs to the stages sometimes differ from these increments. This is done since there may be time lags in the actual stage motion as compared to that which is commanded.

When operation is initiated in either of the automatic modes, the computer reads and stores the x, y coordinates of both stages. Henceforth each time the computer obtains values for Δx^m and Δx^r (i.e., 120 times a second) it adds these increments to the stored values which it is maintaining for commanded stage coordinates. This is equivalent to digital integration of the computed increments, and the initially read stage coordinates are the initial values of the integrals thus obtained. In each cycle of the real time program the computer compares these integrated increments with actual stage coordinates at the time of that particular cycle. The vector differences are the values actually output to the stage servo system. Thus the stages are continually being commanded to move by the amounts which their actual positions differ from where the computer has determined they ought to be. Consequently time lags occurring during periods when the stage commands call for acceleration tend to be offset by overshoots during periods of deceleration.

Servo theory tells us that for zero steady state positional error a servo open loop transfer function should have a pole at the complex frequency origin. This is the effect which is achieved by the digital integration described above. Zero positional error is particularly relevant for the correction signals derived from the IAS since it's desired that the two stages approach precisely corresponding points. Zero positional error is also relevant for trackball control since it's desirable that the operator be able to direct precise stage positioning. In both these cases the temporary time lags are inconsequential, since they do not result in net position errors. All this is accomplished by having the computer perform digital integration of the computed position increments.

V. COMPENSATION OF DISTORTIONS FOR STEREO VIEWING

Figure 4 illustrates the importance of the points of perspective for stereo viewing. P_1 and P_3 represent two frame photographs both having their point of perspective at coordinates X_1^a. Similarly P_2 and P_4 represent two frame photographs with their point of perspective at X_2^a. A and B are two ground points and a_1, a_2, a_3, a_4 are the respective photograph points corresponding to A. Likewise b_1, b_2, b_3, b_4 are photograph points corresponding to B. From the figure it's evident that photographs P_1 and P_3 contain the same information (though possibly in different geometric form). Likewise photographs P_2 and P_4 contain the same information. In general, however, the photographs P_2 and P_4 do not have the same information content as P_1 and P_3. In what follows, P_1 and P_3 will be referred to as "equivalent" photographs, and likewise for P_2 and P_4. Evidently, then, a pair of photographs which are equivalent to one another cannot be viewed in stereo. In other words, a necessary (but not sufficient) condition that two photographs be suitable for stereo viewing is that they be taken with different points of perspective.

5.1 Conditions for Stereo Viewing

Figure 4 is drawn to suggest that P_1 and P_2 are vertical photographs whereas P_3 and P_4 are tilted photographs. If vertical frame photographs are regarded as undistorted standards, then tilted photographs are regarded as undistorted standards, then tilted photographs may be said to contain tilt distortion, and strip and pan photographs also contain motion distortion. Generally speaking, these

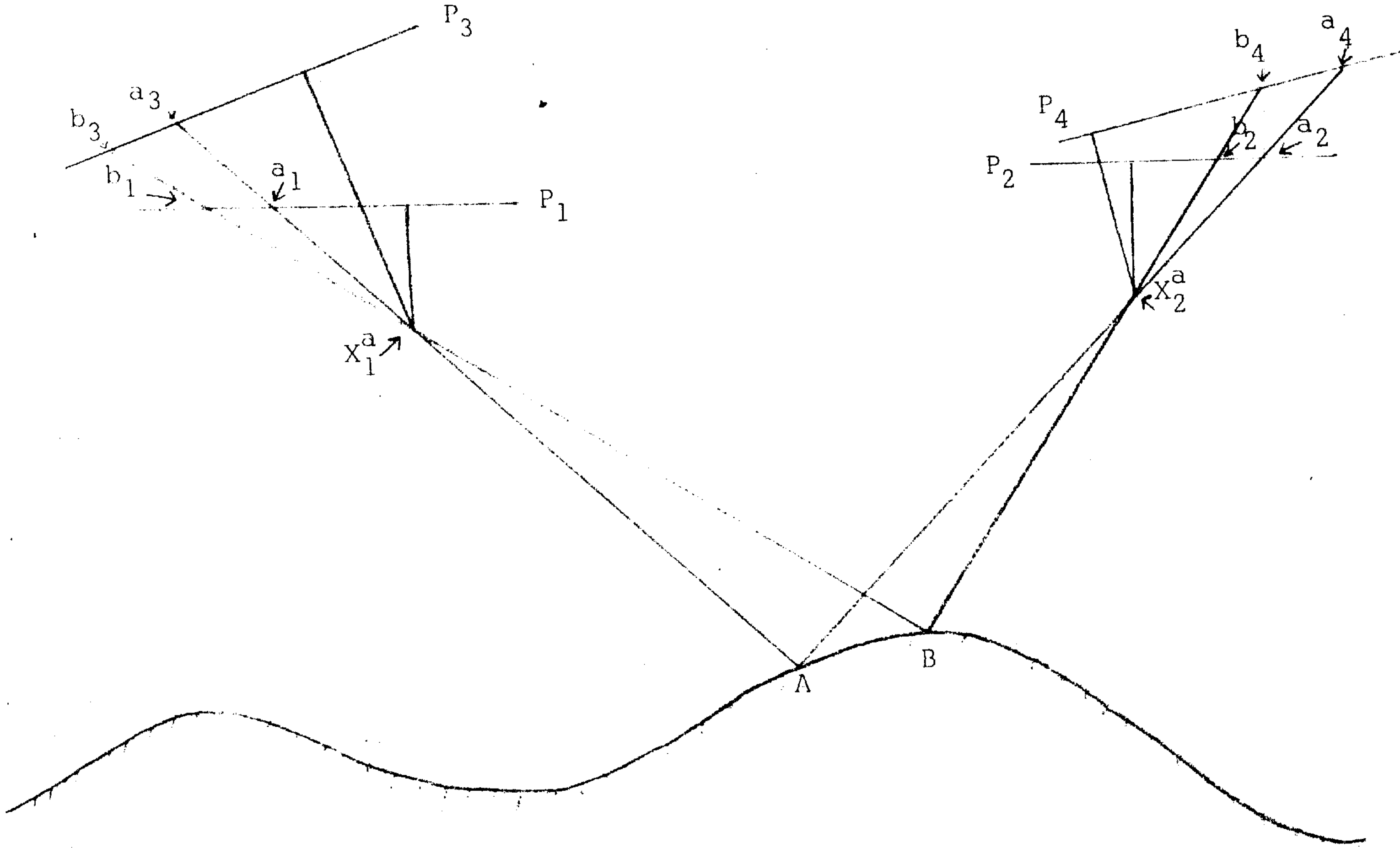


Figure 4.

Photographs P_1 and P_2 illustrate stereo viewing. Photographs P_3 and P_4 are represented as "equivalent" to P_1 and P_2 respectively.

distortions tend to interfere with stereo viewing, but some degree of distortion can be tolerated without undue discomfort.

Experience seems to demonstrate that frame type photograph pairs with oblique tilt (not convergence) can be viewed about as comfortably as vertical photographs. Convergence type tilt, on the other hand, seems to make stereo viewing quite uncomfortable. Hence the criterion which will be used here for correcting distortions will be to produce images as nearly as practical like oblique or vertical (not convergent) frame photographs. Furthermore, the obliquity angles for the two images of a stereo pair will be made equal. Discussion of the details of this scheme will, however, be deferred until after the functioning of the optical system has been outlined.

5.2 Elements of the Stereocomparator Optical System

Figure 5 shows the main projection elements of one-half of the Stereocomparator optical system (the other half is analogous to a mirror image of Figure 5). The system is designed to provide continuously variable magnification over two ranges: 10X to 100X and 20X to 200X (by way of the zoom lens and two interchangeable objective lenses). It also provides continuously adjustable anamorphic stretch over the range 1/1 to 2/1 - with a continuously variable angle for the stretch axis. Continuously variable rotation of the magnified and anamorphosed image is also provided. In other words, each half of the main projection optical system is continuously adjustable through 4 degrees of freedom: magnification, anamorphic stretch ratio, anamorphic

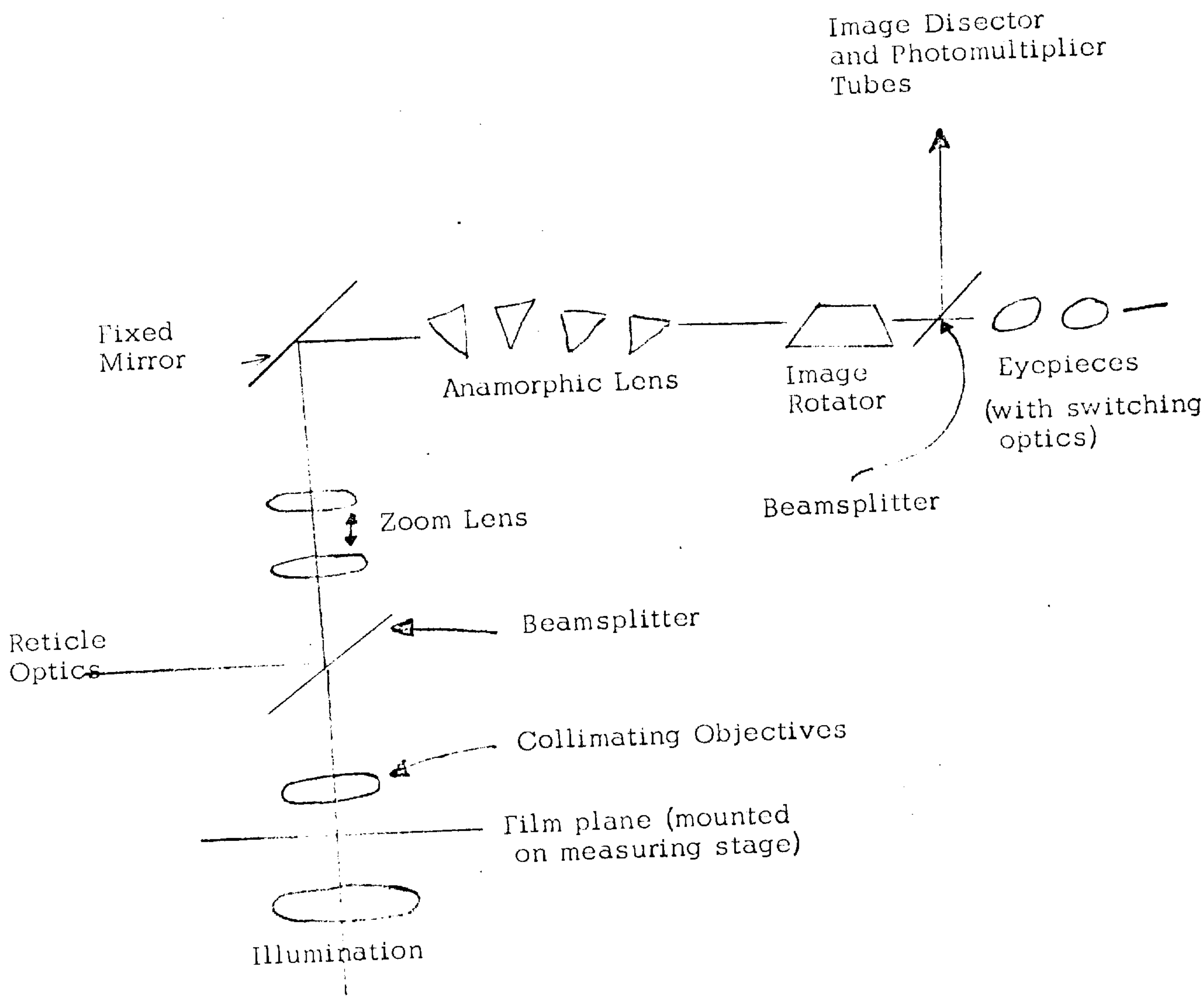


Figure 5

Partial diagram of one-half of Stereocomparator Optical System stretch angle, and image angle. In addition the eyepieces can be switched 4 ways: (1) normal stereo, (2) crossed stereo, (3) binocular viewing of the left photograph, and (4) binocular viewing of the right photograph.

Figure 5 shows a beamsplitter just above the objective lens - to allow superposing a reticle image on the image of the photograph (i.e., appearing to lie in the plane of the photograph). The reticle optics include a zoom lens and an anamorphic lens which are servoed to the corresponding elements in the main projection path so as to minimize changes in size or shape of the reticle when the photographic image is zoomed and/or anamorphosed. The reticle optics also include yet another zoom lens which allows the operator to adjust the reticle size over a 4 to 1 range. Thus the reticle serves as a fixed pointer (marking the optical axis), to which various points of the photograph may be brought as directed by the operator - in order to measure the coordinates of the selected points.

Considering both halves of the optical system (which are like mirror images of one another), it's evident that two aerial photographs taken of the same ground region, but with different points of perspective, may be viewed (and measured) simultaneously. The 4 degrees of freedom available in adjusting each half of the optical system are generally sufficient to compensate geometric distortions in the photographs so the operator is able to form a stereo perception of the two images. Simultaneously with forming a stereo model of the

Approved For Release 2003/01/28 : CIA-RDP78B05171A000600030001-0

ground region covered by the two photographs, the operator tends to perceive the two reticles (one in each eyepiece) fused into a single dot which appears to lie in, or float over, the stereo model. By adjusting the two trackballs the operator may set the floating dot so it appears to lie precisely on the surface of the stereo model. Under these conditions the two photographs have precisely corresponding points brought to the respective optical axes. Depressing one of the record buttons then produces measurement and digital output (either to a card punch or to a link to an external computer) of the two sets of x-y coordinates for the corresponding points. Such digitization of corresponding points is the prime function of the Stereocomparator. Automatic adjustment of the optical system so as to enable stereo viewing is a secondary function which greatly aids the operator in selecting the points he wishes to have digitized.

5.3 Matrix Representation of the Functions of the Optical Elements

Photographs placed on the Stereocomparator's two measuring stages are, of course, constrained to planes perpendicular to the respective optical axes. The two images of these photographs formed by the respective optical systems may also be thought of as lying in planes perpendicular to the optical axes - at the top or eyepiece ends. Hence it is convenient to orient the photograph and image coordinate systems with their respective z axes parallel to the optical axes. As in chapter IV the master and slave photograph coordinate axes are designated by x^m and x^r respectively.* The coordinate axes for both images are designated x^j, and the x axes are taken in the plane of, and normal to, the two optical axes - i.e., in the direction of the operator's eye base. That it is reasonable to use parallel axes for the two images follows from the criteria for stereo viewing - as will be shown eventually.

Thus the master image may be represented as a (two dimensional) function of the master photograph

$$x^j = f_1^j \, (x^m). \tag{73}$$

Similarly the slave image may be represented as a (two dimensional) function of the slave photograph

$$x^j = f_2^j \, (x^r). \tag{74}$$

STAT

The [redacted] series expansions of these equations may then be written:

$$\Delta x^j = X_m^j \, \Delta x^m + \frac{1}{2} X_{mn}^j \, \Delta x^m \, \Delta x^n + \ldots \tag{75}$$

for the master image, and

$$\Delta x^j = X_r^j \, \Delta x^r + \frac{1}{2} X_{rs}^j \, \Delta x^r \, \Delta x^s + \ldots \tag{76}$$

for the slave image.

Evidently (75) and (76) must represent the optical transformations (from photograph to image in each case) produced respectively

---

*Although there are, in general, two dimensional coordinate transformations between photograph coordinates and measuring stage coordinates, these transformations are ignored as too trivial to require discussion.

by the master and slave optical systems. In section 5.2 the optical systems were stated to each have four functional parameters (magnification, anamorphic stretch ratio, anamorphic stretch angle, and image rotation angle). These parameters may be used to compute the components of the 2 x 2 matrices X^j_m and X^j_r which appear in (75) and (76) - as will be shown. From this it follows that the 2 x 2 matrices X^j_m and X^j_r must be substantially independent of the x, y coordinates about the optical axes, and hence, that the higher order derivatives $X^j_{mn} \cdots$ and $X^j_{rs} \cdots$ may be taken as zero. The 2 x 2 matrices X^j_m and X^j_r will be referred to as the master and slave optics transformation matrices.

Figure 5 shows that each optical system has three elements in tandem - the zoom lens, the anamorphic lens, and the image rotator. These elements all operate in collimated light. Hence they may be thought of as separated by planes corresponding to intermediate images. The input (plane) to the zoom lens is the photograph itself. The output (plane) from the zoom lens is also the input (plane) to the anamorphic lens, whose output, in turn, is input to the image rotator. Finally the output from the image rotator becomes the actual optical image (by way of a decollimating lens and the eyepieces). Thus it is proper to represent the function of each optical element by a 2 x 2 matrix, and the matrix product of all the constituent matrices is the optical transformation matrix for the whole system. The matrices for the individual elements are as follows (for the master optical system, which is typical of both optical systems):

5.3.1 Zoom Lens

Since the output from the zoom lens is a magnified replica of the input, the transformation matrix must be the identity matrix multiplied by a scalar whose value is the magnification.

$$X_m^\alpha = M\delta_m^\alpha \tag{77}$$

5.3.2 Anamorphic Lens

The anamorphic lens may be regarded as having a major axis and a perpendicular minor axis - of magnification. The magnification in the direction of the minor axis will be taken as unity - hence the magnification in the direction of the major axis is equal to the anamorphic stretch ratio. The anamorphic angle will be defined as the angle of the major axis with respect to the x axis of the optical system. The transformation matrix is hence the product of three matrices:

$$X_\alpha^\lambda = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 \\ \sin\theta_2 & \cos\theta_2 \end{bmatrix} \begin{bmatrix} a & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta_2 & \sin\theta_2 \\ -\sin\theta_2 & \cos\theta_2 \end{bmatrix} \tag{78}$$

wherein a is the anamorphic stretch ratio and θ_2 is the anamorphic angle. Evidently the third matrix in (78) rotates the image plane axes so the rotated x axis is parallel to the major axis of the anamorph. The second matrix in (78) then gives the magnifications along the rotated x and y axes. Finally the first matrix on the right side of (78) rotates the axes back to the original orientation. Performing the matrix multiplications in (78) gives

$$X_a^\lambda = \begin{bmatrix} (a\cos^2\theta_2 + \sin^2\theta_2) & (a-1)\sin\theta_2\cos\theta_2 \\ (a-1)\sin\theta_2\cos\theta_2 & (a\sin^2\theta_2 + \cos^2\theta_2) \end{bmatrix} \tag{79}$$

Thus X_a^λ is a symmetric matrix which is the identity matrix if the anamorphic ratio is unity.

5.3.3 Image Rotator

The transformation matrix for the image rotator is essentially a standard two dimensional rotation matrix. However, it represents rotation of vectors (in the image) rather than rotation of axes. Consequently it is the transpose of the corresponding axes rotation matrix.

$$X_\lambda^j = \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 \\ \sin\theta_1 & \cos\theta_1 \end{bmatrix} \tag{80}$$

wherein θ_1 is called the image rotation angle. Note, however, that the image rotator acts on an image which is, in general, modified by the anamorphic lens. The combined effect is such that this name for θ_1 should not be taken too literally.

Multiplying together the three component matrices then gives the overall master optics transformation matrix:

$$X_m^j = X_\lambda^j X_a^\lambda X_m^a. \tag{81}$$

Similarly the slave optics transformation matrix is the product of three component matrices:

$$X_r^j = X_\mu^j X_\beta^\mu X_r^\beta. \tag{82}$$

Approved For Release 2003/01/28 : CIA-RDP78B05171A000600030001-0

The forms of these component matrices are, of course, the same as those given for the corresponding master component matrices - but with values for the slave optical elements substituted.

The right sides of equations (81) and (82) are each functions of four variables (magnification, etc.). If the latter are regarded as unknowns then (81) and (82) may be solved for them as functions of the components of the matrices on the left sides of (81) and (82). These solutions will be indicated later, but first a method will be shown for determining values for the components of the matrices X_m^j and X_r^j, so that these components become the known variables in (81) and (82).

5.4 Equivalent Frame Images

The concept of an equivalent frame image will be used to determine a method for computing the optics transformation matrices X_m^j and X_r^j. Referring back to Figure 4, P_3 is considered to be equivalent to the frame photograph P_1 since it has the same point of perspective. Strip and panoramic photographs do not, however, have the same point of perspective for all of their points. Hence for these types of photographs there is no strictly equivalent frame type. Nevertheless, the concept of equivalence will be used, possibly loosely, to include frame images of small extent which are considered approximately equivalent to small regions of real photographs - by reason of having nearly the same point of perspective.

By reasoning similar to that leading to equation (57) the following equation may be written for a frame type image related to the master photograph, using the parameters of a tracking plane.

$$x^j = x^j_{10} + f \frac{C^j_a \, Y^a_m \, (x^m - x^m_1)}{C^3_b \, Y^b_n \, (x^n - x^n_1)} \tag{83}$$

with

$$Y^a_m = C^a_m + \frac{\mu_m}{D_1} (X^a_{10} - X^a_1). \tag{84}$$

Equation (83) is expanded as a [redacted] series which applies over the region of the photograph which is, at any particular time, within the field of view of the master optical system. This series may be written: STAT

$$\Delta x^j = X^j_m \, \Delta x^m + \frac{1}{2} X^j_{mn} \, \Delta x^m \, \Delta x^n + \cdots \tag{85}$$

with

$$X^j_m = \frac{\partial x^j}{\partial x^m} + \frac{dx^j}{dt} \frac{\partial t}{\partial x^m}, \tag{86}$$

$$X^j_{mn} = \frac{\partial X^j_m}{\partial x^n} + \frac{dX^j_m}{dt} \frac{\partial t}{\partial x^m}, \text{ etc.} \tag{87}$$

Thus (85) has the same form as (75), and X^j_m, as computed by (86), is seen to be the master optics transformation.

The following considerations apply in taking the derivatives called for in (86): (1) To make the image as nearly like a frame type as possible, x^j_{10}, C^j_a, and X^a_{10} are treated like constants. (2) To make the image as nearly equivalent as possible X^a_{10} is set equal to the instantaneous value of X^a_1 which corresponds to the photograph point which is at the optical axis. (3) The tracking plane parameters μ_m and D_1 are treated as constants. (4) C^a_m, x^m_1, and X^a_1 are, as usual, treated as functions of time which is itself a function of the coordinates of the point under observation. Wherever the displacement vector $(x^m - x^m_1)$ appears, it is treated as in section 3.1.4 - after differentiation has been performed. Equations (44), (45) and (48) may be used to obtain $\frac{\partial t}{\partial x^m}$.

The formulas for the image equivalent to the slave photograph have the same forms as those given above but with appropriate changes in indices. Thus:

$$x^j = x^j_{20} + f\,\frac{C^j_a\, Y^a_r\, (x^r - x^r_2)}{C^3_b\, Y^b_s\, (x^s - x^s_2)} \tag{88}$$

with

$$Y^a_r = C^a_r + \frac{\mu_r}{D_2}\,(X^a_{20} - X^a_2), \tag{89}$$

wherein

$$\mu_r = C^a_r\, \mu_a = C^a_r\, C^m_a\, \mu_m$$

and

$$D_2 = D_1 + \mu_a\,(X^a_2 - X^a_1).$$

$$\Delta x^j = X^j_r\, \Delta x^r + \cdots \tag{90}$$

$$X^j_r = \frac{\partial x^j}{\partial x^r} + \frac{dx^j}{dt}\,\frac{\partial t}{\partial x^r}. \tag{91}$$

The symbol x^j is used for the coordinates in both the master equivalent image and the slave equivalent image. This means, not that these two different sets of coordinates can be equated to each other, but that they are computed for parallel sets of axes. Parallel sets of axes are necessary in order that the images produced by the two optical systems will appear to have their normals parallel to each other and perpendicular to the eyebase (which is taken as the x axis direction). These conditions on the respective normals are implied in the criteria for stereo viewing, given in section 5.1.

5.5 Solution of Equations

In section 5.3 it was shown that the basic performance of the Stereocomparator optical system could be expressed by the optics transformation matrices X_m^j and X_r^j. In section 5.4 a method was given for computing the components of these matrices in terms of the photograph geometry and the criteria for stereo viewing. In this section equations (81) and (82) will be solved for the unknowns in their right sides, i.e., magnification, anamorphic stretch ratio, etc.

It will be convenient to use the following short hand notation:

$$s_1 = \sin \theta_1$$

$$c_1 = \cos \theta_1$$

$$s_2 = \sin \theta_2$$

$$c_2 = \cos \theta_2$$

Then (79) may be written

$$X_a^\lambda = \begin{bmatrix} a\,c_2^{\,2} + s_2^{\,2} & (a-1)\, s_2\, c_2 \\ (a-1)\, s_2\, c_2 & a\, s_2^{\,2} + c_2^{\,2} \end{bmatrix}$$

$$= \begin{bmatrix} \left[\frac{a+1}{2}(c_2^{\,2} + s_2^{\,2}) + \frac{a-1}{2}(c_2^{\,2} - s_2^{\,2})\right] & \left[\frac{a-1}{2}\, 2\, s_2\, c_2\right] \\ \left[\frac{a-1}{2}\, 2\, s_2\, c_2\right] & \left[\frac{a+1}{2}(c_2^{\,2} + s_2^{\,2}) - \frac{a-1}{2}(c_2^{\,2} - s_2^{\,2})\right] \end{bmatrix}$$

$$= \begin{bmatrix} \left(\frac{a+1}{2} + \frac{a-1}{2}\cos 2\,\theta_2\right) & \frac{a-1}{2}\sin 2\,\theta_2 \\ \frac{a-1}{2}\sin 2\,\theta_2 & \left(\frac{a+1}{2} - \frac{a-1}{2}\cos 2\,\theta_2\right) \end{bmatrix} \qquad (92)$$

Hence (81) may be written

$$X_m^j = M \begin{bmatrix} \frac{a+1}{2}\, c_1 + \frac{a-1}{2}\cos(\theta_1 + 2\,\theta_2) & -\frac{a+1}{2}\, s_1 + \frac{a-1}{2}\sin(\theta_1 + 2\,\theta_2) \\ \frac{a+1}{2}\, s_1 + \frac{a-1}{2}\sin(\theta_1 + 2\,\theta_2) & \frac{a+1}{2}\, c_1 - \frac{a-1}{2}\cos(\theta_1 + 2\,\theta_2) \end{bmatrix} \qquad (93)$$

Now represent the components of X_m^j as follows:

$$X_m^j = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \qquad (94)$$

Thus (93) and (94) are 4 equations with A, B, C, D known (by section 5.4) and M, a, θ_1 and θ_2 treated as the unknowns. From them may be derived the following 4 equations:

$$\begin{aligned} A + D &= M\,(a+1)\, c_1 \\ C - B &= M\,(a+1)\, s_1 \\ A - D &= M\,(a-1)\cos(\theta_1 + 2\,\theta_2) \\ C + B &= M\,(a-1)\sin(\theta_1 + 2\,\theta_2) \end{aligned} \qquad (95)$$

Equations (95) may be solved, and the results are:

$$a = \frac{A^2 + D^2 + C^2 + B^2 + \sqrt{[(A-D)^2 + (C+B)^2][(A+D)^2 + (C-B)^2]}}{2\ (AD - CB)} \qquad (96)$$

$$M = \sqrt{\frac{AD - CB}{a}} \qquad (97)$$

$$0_1 = \tan^{-1} \frac{C - B}{A + D} \qquad (98)$$

$$0_1 + 2\ 0_2 = \tan^{-1} \frac{C + B}{A - D} \qquad (99)$$

wherein the positive sign (only) has been chosen in front of the radical; corresponding to the physical fact that $a \geq 1$. Equations (96)-(99) apply (separately) both to the master and to the slave optical systems. For convenience they may have the subscript 1 appended to all variables when applied to the master optics, and the subscript 2 appended to all variables when applied to the slave optics. It may be shown that the determinent (AD - CB), when its elements are computed by equation (86) for an aerial photograph, does not vanish for any point in the photograph which corresponds to a ground point lying below the horizon.

5.6 Automatic Without Correlator Control of Optics

As was stated in section 4.1.4 the control computer program may be separated into a real time part and a non-real time part. Evidently, to control the optics, the real time part must perform operations equivalent to evaluating equations (96) - (99) for both the master and the slave sides. In order that this may be done rapidly (96) - (99) are therefore expanded as [redacted] series, with terms of order higher than first being neglected. Hence the real time program must evaluate the following expressions:

STAT

$$\Delta a = \left(\frac{\partial a}{\partial A}\frac{\partial A}{\partial x} + \frac{\partial a}{\partial B}\frac{\partial B}{\partial x} + \frac{\partial a}{\partial C}\frac{\partial C}{\partial x} + \frac{\partial a}{\partial D}\frac{\partial D}{\partial x}\right)\Delta x + \left(\frac{\partial a}{\partial A}\frac{\partial A}{\partial y} + \cdots\right)\Delta y \qquad (100)$$

$$\Delta M = \left(\frac{\partial M}{\partial A}\frac{\partial A}{\partial x} + \cdots\right)\Delta x + \left(\frac{\partial M}{\partial A}\frac{\partial A}{\partial y} + \cdots\right)\Delta y \qquad (101)$$

$$\Delta\theta_1 = \cdots \qquad (102)$$

$$\Delta\theta_2 = \cdots \qquad (103)$$

In (100) - (103) the various parenthized expressions, which may be labeled

$$\frac{\partial a}{\partial x},\ \frac{\partial a}{\partial y},\ \frac{\partial M}{\partial x},\ \frac{\partial M}{\partial y},\ \frac{\partial \theta_1}{\partial x},\ \cdots,$$

are computed in the non real time program, and the values are whence supplied to the real time program. Thus the real time program performs only a few computations (in fixed point arithmetic) to determine the required increments of adjustment for the optical elements. As in stage control (see section 4.3), these incremental adjustments are digitally integrated before being output to the optics. The reasons for integrating the optical control signals are substantially the same as those given for the stage servos.

The optical servos differ from the stage servos, however, in that the position feed-back signals are analog rather than

Approved For Release 2003/01/28 : CIA-RDP78B05171A000600030001-0

digital (i.e., they are generated by analog potentiometers rather than by digital counters). These analog position signals are converted into digital form so that the computer can keep track of the optical settings. The A/D converters, however, turn out to have a resolution which is too low for the requirements of the optical servo systems. Consequently the analog position signals are used directly (rather than the digital conversion of them) in the servo position feed-back loops. Correspondingly the computer output signals are the actual integrated values rather than the differences between these and the optical position signals. These digital outputs are converted to analog and the differences between them and the position feed-back signals are taken by analog circuitry as part of the optical servo systems. In other words, by subtracting the position feed-back signals from the computer position commands with analog circuitry rather than digitally in the computer, the position feed-back loops bypass having them, in effect, converted to digital and then back to analog form.

The non real time program uses (94), (86) and the indicated partial derivatives of (96) through (99) to evaluate the parenthesized expressions in (100) - (103). Although it would be proper to use (87) for the partial derivatives $\frac{\partial A}{\partial x}$, etc., an approximate method is used instead. This consists of evaluating (86) for several nearby points and dividing out the incremental differences in much the same way as was described for (70) and (71).

A study of the partial derivatives of (96) and (97) shows that they include, among others, the expressions

$$\frac{A - D}{\sqrt{(A - D)^2 + (C + B)^2}} \tag{104}$$

and

$$\frac{C + B}{\sqrt{(A-D)^2 + (C + B)^2}} \tag{105}$$

which become indeterminent when

$$A - D = C + B = 0.^{*} \tag{106}$$

Furthermore, (99) itself, becomes indeterminent under these conditions. It may be seen, however, that under these conditions the right side of (96) becomes equal to one. Hence the value of θ_2 becomes inconsequential. The computer program contains a test for the condition

$$\frac{(A - D)^2 + (C + B)^2}{AD - CB} < \epsilon \tag{107}$$

where ϵ is some sufficiently small positive number (say about 10^{-6}). This test is applied repetitively and whenever it suddently becomes true a is set to 1 and the value of θ_2 is frozen at the last value which was assigned to it. The freeze continues until (107) is found to be no longer true, at which point use of (96) and (99) is resumed for evaluating a and θ_2. In other words, the indetermency condition corresponds to there being no need for anamorphic correction, and a logical switch causes the anamorph to be set to unity when this is true.

*The corresponding conditions $A + D = C - B = 0$ cannot occur since $AD - CB > 0$.

The preceding paragraphs describe the method by which the computer adjusts the two optical systems so as to compensate for changing distortions as the stages move to various parts of the two photographs. During this automatic process, the operator has no manual control of the optics except that he may direct the scale factor to which the computer is to bring the two equivalent images. He does this by turning either magnification control in the direction for either increasing or decreasing scale factor as he desires. The two magnification controls operate interchangeably (when in either AUTOMATIC mode) to drive both zoom lenses at rates proportional to their respective magnification settings. The computer program then resets the otherwise arbitrary scale factor (f) in equations (83) and (88) to correspond to the new setting of the two zoom lenses.

The OPTICS INDEPENDENT button provides the operator with means for directing the computer to discontinue automatic tracking of the optics. When this button is selected the operator may manually set the various optical elements as he desires. Having done so, the operator may either reset the OPTICS INDEPENDENT button, or he may select the REORIENT button. In the former case the computer will cancel the manual adjustments and will resume automatic tracking from the settings which existed prior to the OPTICS INDEPENDENT operation. If the REORIENT button is selected (without having reset OPTICS INDEPENDENT) then the computer will resume automatic tracking from the newly established settings. In other words, in this case the computer takes the new settings as "initial" conditions for the various digital integrations which it is performing.

5.7 Automatic With Correlator Control of Optics

Equations (64), (65) and (66) may be combined to give the slave image as a function of the master image:

$$\Delta x^j = X^j_r \, X^r_m \, X^m_k \, \Delta x^k \qquad (108)$$

Equation (108) may be interpreted as follows: Δx^k represents a (two dimensional) vector in the master image, X^m_k transforms this vector into a corresponding vector in the master photograph, X^r_m further transforms it into a vector in the slave photograph, and X^j_r finally transforms it into a vector in the slave image - all in two dimensions. The matrix X^r_m has been labeled the tracking matrix and is used as the basis for keeping the two photographs on corresponding points. In fact, however, it is a two dimensional scaling matrix of the slave photograph relative to the master photograph, for corresponding small regions. Hence it may also be used as is done in equation (108). In previous sections of this manual, methods have been described for computing values for the three matrices in (108) such that, in general, the two images should match as required for stereo viewing. The IAS (Image Analysis System) makes an emperical comparison of the two images and provides feed back signals representing differences (to the first order) between them. These differences are used to modify the settings of the <u>slave</u> optical system so as to improve the first order match between the two images, i.e., to reduce the differences themselves.

It may be seen that the conditions for optimum stereo viewing require that the two images be identical with one another except for differences higher than first order (in other words, the x-parallax required for perception of stereo is a difference of higher order than first). This means that under optimum conditions the matrix product in equation (108) should equal the identity matrix. As was stated in Section 4.2, the IAS outputs 6 analog signals - 2 for stage tracking and 4 for optics tracking. The latter are equivalent to the components of the matrix which is reciprocal to the product matrix in (108). Hence the matrix of the 4 IAS analog output signals - x and y scale factors and x and y skew factors - should approach the identity matrix as the optics adjustments approach those required for optimum stereo viewing.

Let the matrix of the IAS optical output signals be represented by

$$X_j^k = \begin{bmatrix} A_c & B_c \\ C_c & D_c \end{bmatrix}. \qquad (109)$$

Then the matrix

$$X_j^k - \delta_j^k = \begin{bmatrix} (A_c - 1) & B_c \\ C_c & (D_c - 1) \end{bmatrix} \qquad (110)$$

may be interpreted as an error matrix representing the extent to which the optics adjustments fail to be optimum for stereo viewing. Hence (110) may be applied to the optics servo systems as a correction matrix which will drive the optics as required to make (109) approach the identity matrix.

As are other control signals, the IAS error signals are treated incrementally and digitally integrated. The corrections are applied to the slave optical system only. Hence the slave optics transformation matrix is multiplied by (110) to produce the modified slave optics matrix:

$$(X_j^k - \delta_j^k)\, X_r^j = X_j^k X_r^j - X_r^k \tag{111}$$

The four elements in (111) are next treated like a 4 dimensional vector

$$\Delta A^{\alpha} = (\Delta A,\ \Delta B,\ \Delta C,\ \Delta D) \tag{112}$$

This 4-vector is multiplied by the four 4-vectors

$$\frac{\partial a_2}{\partial A^{\alpha}},\ \frac{\partial M_2}{\partial A^{\alpha}},\ \frac{\partial \theta_{12}}{\partial A^{\alpha}},\ \text{and}\ \frac{\partial \theta_{22}}{\partial A^{\alpha}}$$

to produce the increments

$$\Delta a_2,\ \Delta M_2,\ \Delta \theta_{12},\ \Delta \theta_{22}$$

which are added to the results of (100) - (103) for the slave side. In this way the optics adjustments computed by (100) - (103) are corrected for the signals from the IAS. Otherwise optics control is as was described under 5.6.