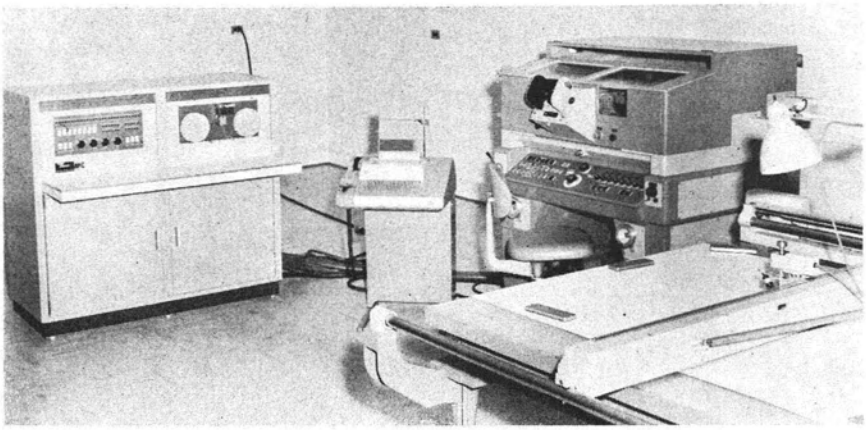

FIG. 1. AP-2C analytical plotter system. The components are, from left to right: control computer of the system, teletype unit, viewer, and plotting table.

DR. S. E. MASRY
DR. G. KONECNY
University of New Brunswick
Fredericton, N.B., Canada

New Programs for the Analytical Plotter

Programs for highway cross-sectioning, plotting from two-media photos and generation of digital terrain model, demonstrate the capabilities and limitations of the instrument.

INTRODUCTION

SEVERAL TYPES OF analytical plotters have been developed. A summary of these is given in Reference 1. Different models and their capabilities are described in several papers References 2 to 8. Very little, however, can be found in literature about photogrammetric problems actually dealt with, techniques used on analytical plotters, and a thorough demonstration of what can be done on models already developed.

This may be due to the fact that not many organizations have access to an analytical plotter as yet, or because the majority of computer models used with the analytical plotters use for their programming a machine language (or at best an assembly language) with which not many programmers and photogrammetrists are familiar or wish to become involved. The reasons for not utilizing a control computer with easier programming facilities are mainly economical.

Extended programming experience must therefore first be gained in order to treat new photogrammetric problems on the analytical plotters. In addition to that—and this is very important—flow charts and detailed documentation of basic programs developed with the instrument (such as the real-time program) must be available. Only in this way can many new photogrammetric problems be treated; by modifying the original programs. For example, a program developed for the treatment of conventional photography on the plotter can be modified for evaluation of two-media photography. If this documentation is not available, the user is left with one of two choices: first, to analyze the basic programs in order to obtain the necessary information, which is not an easy task for a machine language; or, second, to rewrite these programs from the beginning, which obviously is a duplication of effort and might take a longer time. The first course was adopted in the case of the AP-2C plotter at the University of New Brunswick.

Investigation of the capabilities of the different models of analytical plotters is quite important at this stage of their evolution. The rapid development in the field of electronic computers underlines this importance by raising several questions; for example, the question of time-sharing techniques which are gradually becoming established. In time-sharing, the photogrammetric operator has at his finger-tips the power of a large computer with banks of data files and all the programs developed in the field of photogrammetry as a resource. Would it then not be economical and efficient to have the analytical plotter controlled by a small computer in real-time and all other operations such as relative orientation, absolute orientation, strip and block computations, etc. carried out on the larger computer?

ABSTRACT: *Analytical plotters are developing rapidly. Their capabilities, limitations and future improvements can be determined only through their extensive use. Consequently, the evolution of the analytical plotters can be directed towards more economic and efficient systems. As a step in this direction, several programs have been developed for the AP-2C plotter. The programs demonstrate the capabilities of the instrument and the difficulties encountered. Some of the programs described allow semi-automatic scanning of the model, plotting from two-media photography, and semi-automatic highway cross-sectioning.*

In this instance, the capabilities of the system can be extended to practically all those of the time-shared computer and any disadvantages due to the control computer being small may be reduced by an efficient software which can be easily modified according to the problem. Or, considering the reduction in computer cost with time, would it be better to use a larger control computer than the ones usually in use at present? For analytical plotters with small control computers such as the AP-2C, several photogrammetric problems can be treated with only software changes. Therefore, it is only through extensive use of analytical plotters that their capabilities, limitations and future improvements can be determined. Consequently, the questions posed can be answered in a way which makes the evolution of the analytical plotters more economic and efficient.

This paper describes first the AP-2C programming language and the basic programs provided by the manufacturer. Two main parts of the paper follow; the first part deals with utility programs and routines developed to facilitate the programming effort, and the second part deals with programs developed to solve new photogrammetric problems whereby emphasis is put on the method rather that on programming. Instruction lists of some of these programs are given in Reference 9.

AP-2C Programming and Basic Programs

The AP-2C analytical plotter system is shown in Figure 1. The viewer and plotting table were constructed by Ottico Meccanica Italiana (OMI) and the control computer was manufactured by the Bendix Corporation. Information about the organization of the components of the system can be found in References 1 and 7.

The computer is an AP/C model but it has an extended memory. The memory consists of 14 long delay lines (serial access) with 256 words each, one line with 16 one-word registers, and three one-word register lines. The computer word consists of 28 bits; 27 bits represent a binary number or an instruction; and one bit is a parity bit (even parity is used). For a number, the 27th bit represents the sign: a 0 represents a positive number and a 1 gives a negative number. For an instruction the bits are divided into four groups as shown in Figure 2.

The first six bits give the *operation code* (two digits in octal); the second group of five bits gives the *line code* (two digits in decimal); the next group of eight bits gives the *operand code* (three digits in decimal); and the last group of eight bits gives the *next instruction code* (three digits in decimal). A typical instruction in *decimal* format is:

70 01 150 151

If this instruction is executed, it transfers to one of the accumulator registers the content of word 150 in the long line 01.

The programmer writes his instructions in this decimal format which is referred to as *machine language* or *absolute language*. In order to store these instructions in memory in

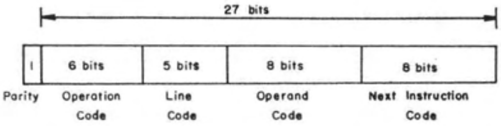

FIG. 2. Structure of the AP-2C computer word for an instruction. The word is divided into four groups giving the Operation Code, the Line Code, Operand Code, and Next Instruction Code.

binary form, the decimal instructions are punched on a paper tape using a teletype. The tape is then read into the computer, converted into binary, and the teletype characters representing each instruction in ASCII code are determined. A tape containing these characters is then punched on the teletype. The conversion of the decimal tape into an ASCII tape is accomplished by a program preparation routine provided with the instrument. The ASCII tape can then be loaded into memory using a tape input routine provided with the instrument. This procedure makes *debugging* of new programs tedious and can be avoided by loading the decimal instructions directly into memory using a *read* routine explained later.

The basic programs and routines provided with the instrument can be divided into two main groups: the real-time program, and programs which are not time-dependent. The latter are designated as service programs.[10] Examples of these are relative orientation, absolute orientation, input and output routines, etc. The real-time program and most of the service programs and routines, including the relative orientation program, are contained in one main tape. The absolute orientation program and some additional routines are supplied in an overlay tape. If the main tape is loaded into memory, relative orientation can be performed and computed. Loading the overlay tape allows one to perform absolute orientation, but the relative orientation program is not available after the overlay tape is loaded. In either mode of operation, only 10 long-memory lines are occupied. (These form the standard AP/C model memory).

The additional four long lines proved to be advantageous for developing new programs which utilize some of the basic programs. It was possible to write most of the new programs for these four lines without a need for overwriting some of the basic programs not utilized. This reduced greatly the effort of planning memory allocation, and following the minimum-access, serial-access computer rules which are particularly important for real-time programs. Because, in writing most or all of a new program in the form of an overlay, the programer's choice is limited to the overwritten memory locations which may not suit the planning or optimal timing of the new program.

UTILITY ROUTINES AND PROGRAMS DEVELOPED

Seven programs and routines were developed at the University of New Brunswick. Instruction lists for most of these are given in Reference 9. Two routines are similar to those provided by the manufacturer. They were, however, rewritten for more convenient use by the programer. The purpose of these programs and routines is as follows:

A *Loading Routine* loads decimal tapes directly into memory. No change of the program tapes to binary form using the program preparation routine is necessary. This routine loads instructions as well as positive and negative constants into any memory location. It reduces greatly the effort and time in developing new programs.

A *Dump Program* prints and punches on the teletype the contents of long memory lines. Because the program itself occupies one line, two versions which occupy two different lines were developed to dump the entire computer memory. The dump tapes are in the format used by the read routine and the program-preparation routine. The output tapes can, therefore, be easily incorporated in program tapes.

A *Change Constants Program* reads decimal constants and converts them into the decimal format of an instruction. The constants can thus be included with instructions on one decimal tape that is then converted to an ASCII tape using the program preparation routine. (This procedure is unnecessary if the tapes are read using the *Loading Routine*.)

A *Sine-cosine Routine* calculates the sine and cosine of an angle and returns back to the program.

A *Square-Root Routine* calculates the square root of a binary number and returns to the program.

A *Binary-to-Decimal Conversion-and-Punch Routine* converts the binary contents of any memory location to a seven-digit decimal number with its sign which are displayed on the viewer panel and punched on the teletype. The routine then returns back to the program. A new line on the teletype can be started after each output if the programmer so desires.

Photogrammetric Problems

Programs have been developed which can be incorporated into the basic programs provided by the manufacturer to treat three different photogrammetric problems on the analytical plotter.

Scanning Program

This program allows semi-automatic scanning of the model along lines parallel to the X-direction (Fig 3). the measuring mark is moved automatically in the X and Y directions. The height setting of the measuring mark is controlled using one of the handwheels of the viewer. The spacing between the scan lines in Y-direction and the area of the model to be scanned can be easily changed. The scanning speed can be varied to suit operation under different types of terrain. The time taken to scan one model obviously depends on the terrain and the experience of the operator. In some experiments, a hilly model was scanned in about 2.0 hours. Three versions of this program have been developed to serve different purposes:

- In the first version the scanning speed is controlled using the Veltropolo lever on the viewer panel allowing 13 different speeds. The X, Y and Z model coordinates are recorded automatically at equal intervals in the X-direction and can be stored via a link to an IBM-360/50 computer on a disk. A digital terrain model accessible to Fortran programs can be obtained in this way. The magnitude of the intervals can be changed by modifying one constant in the program. And the program can be used in the relative orientation or in the absolute orientation modes.
- The second version is similar to the first except that the model or ground coordinates can be printed and punched on a tape using the teletype unit.
- The third version is similar to the first version except that the scanning speed is controlled by a series of buttons on the viewer panel allowing four different speeds. For each button, the scanning speed is constant. This may be of advantage if the plotter is used in conjunction with an orthophoto printer.

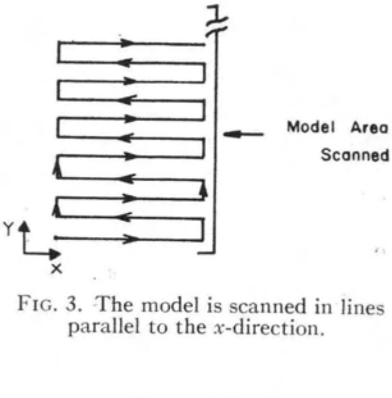

FIG. 3. The model is scanned in lines parallel to the x-direction.

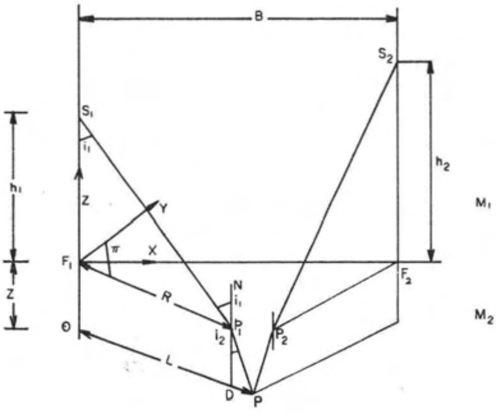

FIG. 4. Geometry of two-media photography.

Another version of the program can be written so that model contours are plotted simultaneously as the model is being scanned.

Evaluation of Two-Media Photography Program*

The two media-photography program allows plotting from photography taken with the camera in one medium M_1 and the object or part of it is in a different medium M_2. Such a case is given if the camera is in air and the object is partly under water.†

To evaluate a stereo-pair of photographs of this type, the effect of refraction should be taken into consideration for object points in the medium M_2 (Figure 4), whereas object points in the medium M_1 can be dealt with in the conventional manner. The effect of refraction for points in the medium M_2 can be derived from Figure 4. In this Figure, the boundary between M_1 and M_2 is assumed to be a plane π; S_1 and S_2 are the two camera stations; S_1F_1 and S_2F_2 are perpendicular to π and intersect it in F_1 and F_2; XYZ is a coordinate system taken with the X-axis along F_1F_2; point P is an object point in the medium M_2; point 0 is along S_1F_1 and has the same Z-coordinate as P; i_1 and i_2 are the angles of refraction and incidence, of the principal ray‡ Pp_1S_1 from P to S_1; ND is the nor-

* Programming of the evaluation of two-media photography commenced on the initiative of Prof. K. Schwidefsky of the University Karlsruhe, Germany. The original formulas used were those of Mr. J. Höhle, Professor Schwidefsky's collaborator. These formulas have been modified and analysed for treatment of the problem on the analytical plotter. A test of this program was based on photography and data supplied by Mr. Höhle.

† Additional work in this field is given in references 11 to 14

‡ It may be interesting to note that the rays $Pp_1\,S_1$ and $Pp_2\,S_2$ are not, except in special cases, coplanar, and lie in two planes perpendicular to π.

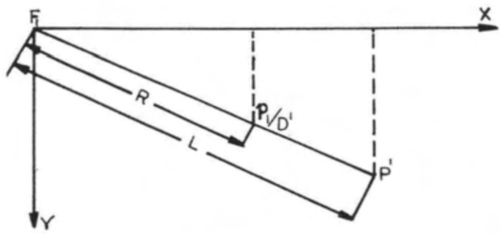

FIG. 5. Projection of points p_1 and P of Figure 4 onto the XY-plane.

mal to π at p_1. The notation for distances are as shown in the figure. From the laws of refraction we have:

$$\sin i_1/\sin i_2 = n \tag{1}$$

where n is the refractive index between M_1 and M_2. From the two triangles $S_1F_1p_1$ and p_1PD, *sin* i_1 and *sin* i_2 can be expressed as:

$$\sin i_1 = R/\sqrt{(h_1^2 + R^2)} \tag{2}$$

and

$$\sin i_2 = (L - R)/\sqrt{[Z^2 + (L - R)^2]} \tag{3}$$

where L and R can be expressed in terms of X and Y-coordinates of points P and p_1 respectively. From Equations 1, 2 and 3 it follows that

$$\frac{R\sqrt{[Z^2 + (L - R)^2]}}{(L - R)\sqrt{(h_1^2 + R^2)}} = n$$

from which

$$R^2[(n^2-1)(L-R)^2-Z^2]+n^2h_1^2(L-R)^2=0. \tag{4}$$

If the X, Y and Z coordinates of point P, and h_1 are known, R can be determined from Equation 4 using Newton's method of approximating to the roots of an equation*. As a first approximation, R can be taken as $Lh_1/(h_1+Z)$. Knowing R, the X and Y coordinates of point p_1 can then be determined using:

$$X_{p_1} = X_P \cdot R/L$$
$$Y_{p_1} = Y_P \cdot R/L \tag{5}$$

where the Z-coordinate is equal to zero. Equations 5 can be obtained from Figure 5 which represents the normal projection of points p_1 and P onto the XY-plane.

A similar procedure can be followed to determine the coordinates of point p_2.

A point of practical consideration may be mentioned here. For small values of L relative to h_1 and h_2, the effect of refraction is small. Consequently, the terms (L-R) of Equation 4 become small relative to other terms in the equation. This may cause computational errors and difficulties in the scaling of the quantities involved. In this event, an approximate formula may be used directly to calculate the ratio R/L of Equations 5. From Figure 4, we have:

$$\sin i_1 \simeq \tan i_1 = R/h_1 \tag{6}$$

and

$$\sin i_2 \simeq \tan i_2 = (L - R)/Z. \tag{7}$$

From 6 and 7,

$$n = ZR/[h_1(L - R)]$$

and by rearranging,

$$(R/L) = h_1/[(Z/n) + h_1].$$

In treating this problem on the analytical plotter, the program determines from the sign of the Z-coordinate of point P the medium in which it lies. Model points in medium M_1 are treated in the same way as in conventional photography: the real-time program calculates the positions of image points corresponding to a model point using the collinearity equations, and outputs to the photocarriages, if any, are determined. On the other hand, for a point P in M_2, two points p_1 and p_2 (Figure 4) must be determined; p_1 shall be considered the model point when dealing with the image point of station S_1 and similarly, p_2 corresponds to S_2. (This way of treatment is made possible through the *mathematical projection* principle which the analytical plotter employs.)

In the above discussion, it was considered that the coordinates of point P and the orientation parameters of the cameras are known. As far as the coordinates are concerned, these can be taken to be equal to the model coordinates due to the movements of the handwheels and foot-plate, and can be used directly in the calculations provided that two conditions are satisfied: (a) the system of model coordinates is similar to the XYZ-system of Figure 4, and (b) the coordinates of the camera stations, which are used by the program in calculating the effect of refraction and image positions, should represent h_1, h_2 and B of actual photography with the same scale s (say), i.e., should be of the form (0, 0, sh_1) and (B, 0, sh_2) where s is a suitable scale factor.† The model scale will in this case be equal to s.

* This equation has four roots, two of which are imaginary. One of the real roots has a value larger than L and has, therefore, no physical meaning.

† This condition is necessary due to the presence of refraction. If h_1, h_2 and B are not known, approximate values can first be entered which results

Consider now the orientation of the cameras. If the orientation parameters are known, their values can be entered using the viewer data entry switches. The values entered should, however, be referred to a sytsem of coordinates similar to that of the plotter. In the event where the parameters are unknown, relative orientation can be calculated using the relative orientation program provided with the instrument. In this instance, two points must be noted. First, only points in the medium M_1 must be selected for relative orientation. Second, any changes in the coordinates of the camera stations due to relative orientation, and which upset the scale relationship explained above, should be substituted for by common rotations of the cameras. The relative orientation program can be modified so that points in the medium M_2 can also be used in relative orientation.

If h_1, h_2 and B are known, absolute orientation is established with respect to the X, Y and Z system of Figure 4 by applying a common ω rotation. One control point in height is required for the determination of ω.

This program was tested and used in plotting from photography taken by a stereometric camera under laboratory conditions. Accuracy comparable to that of conventional photography was obtained.

Highway Cross-sectioning Program.

The highway cross-sectioning program allows semi-automatic determination of cross-sections along the center line of a highway.

The operator slews to the first station A (Figure 6) along the center-line, sets the number of this station on the data entry switches of the viewer panel, and enters the coordinates and the station number of this point into the program by pressing one of the viewer buttons. The operator then drives to the second point B and enters the coordinates of this point in the program in a similar manner. He also defines for the program the boundaries of the stero model, and three constants related to the magnitude of the intervals. The plotter is then ready for cross-sectioning.

Upon energizing one button, the plotter will drive automatically to point A for the determination of the cross-section at this point. The plotter will drive automatically to points of equal spacing, p_1 to p_n along the

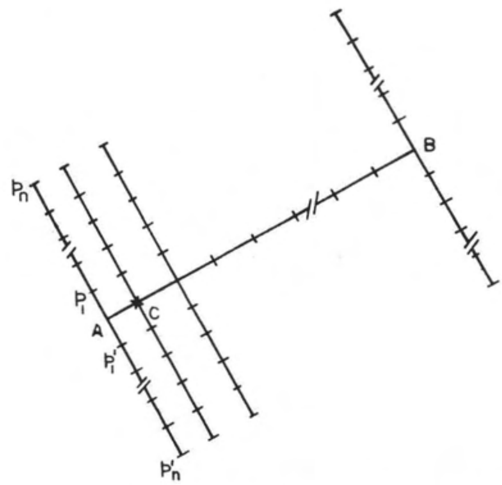

Fig. 6. The highway cross-sectioning program allows an automatic determination of the positions of points along the cross-sections and stations along the center line.

left-hand half of the cross-section. At each point, there will be a pause for a few seconds for the operator to set the measuring mark in height at this point using one of the handwheels. The following data are then printed and punched on the teletype: station number, height at point p, difference between the heights of A and p, and horizontal distance between A and p. The plotter then proceeds to the next point. The program is arranged in such a way that output takes place a few seconds after the height setting is completed or the movement of the handwheel is too small to have an appreciable effect on the measurements.

After point p_n is reached the plotter drives to A. Points p_1' to p_n' on the other half of the cross-section are then similarly determined. The program then calculates the next station C along the centre-line, the plotter drives to C and similar procedure is followed, and so on until point B is reached. Table 1 gives an example of the output obtained.

The number of points observed along the cross-sections, the magnitude of intervals along the cross-sections, and the length of delay between the height setting and output can be easily changed to suit a particular problem.

The program checks for each cross-section whether it extends outside the boundaries of the stereo model. If so, the number of points which will be observed along that cross-section is reduced by the program until all the points lie within the model.

The program allows different speeds of

in approximate refraction calculations. Relative and absolute orientations may have to be repeated until correct coordinates of control points are obtained.

TABLE 1. EXAMPLE OF OUTPUT FROM HIGHWAY CROSS-SECTIONING PROGRAM

(1)	(2)	(3)	(4)	
1023+315	0002645	0000000	0000000	Cross-section at Station A 1023+31.5
1023+315	0002696	0000051	0000999	
. .	. .	. .	. .	
−1023+315	0002754	0000109	0003995	
−1023+315	0002758	0000113	0004995	
1024+000	0002684	0000000	0000000	Cross-sections at Intermediate Stations
1024+000	0002690	−0000006	0000999	
. .	. .	. .	. .	
. .	. .	. .	. .	
−1033+000	0002861	0000002	0003995	
−1033+000	0002878	0000018	0004995	
1034+220	0002872	0000000	0000000	Cross-section at Station B 1034+22.0
1034+220	0002826	−0000046	0000999	
. .	. .	. .	. .	
. .	. .	. .	. .	
−1034+220	0002898	0000026	0003996	
−1034+220	0002915	0000043	0004995	

Column (1): Station (negative sign indicates right-hand half of cross-section).
Column (2): Ground height at cross-section points in 0.1 ft.
Column (3): Difference between heights at cross-section point and at station in 0.1 ft.
Column (4): Distance between cross-section point and station in 0.1 ft.

travelling between the cross-section points. At the maximum speed it was possible to observe 15 cross-sections with 12 points along each cross-section in less than half an hour.

REMARKS ON REAL-TIME REQUIREMENTS FOR THE PROGRAMS DEVELOPED

In the main part of the real-time program for conventional photography, increments are added to the model coordinates for any motion of the handwheels and the footplate. Using the newly formed model coordinates, calculations are made for the positions of the corresponding two photo-points, the outputs to the two photo-carriages, and the outputs to the plotting table. These calculations must be performed about 30 times per second for a smooth operation of the photocarriages. The real-time program calculates also corrections due to lens distortion, earth curvature, etc. As the change in these corrections is usually always at least one order of magnitude smaller than changes in model coordinates, they need not be calculated with the same rate of repetition. For example, if there are five types of corrections to be applied, they can be arranged in the form of five branch routines so that each time the model coordinates are processed one of these routines is operated on in succession. In this way, all routines are executed once for every five calculations of the main part. This arrangement is quite advantageous because all the correction routines cannot be included at the same time in the main part of the real-time program without reducing the repetition rate so that the performance of the carriages is affected. (The problem may be dealt with by using a computer with two computation sections which perform simultaneously: one for the real-time requirements with high repetition rate, and the other for lower rate requirements. This arrangement is used in the *AS-11A* system.[10])

In treating new photogrammetric problems on the analytical plotter system, usually real-time requirements additional to those of conventional photography are created. The new requirements contribute to an increase in the execution time of the real-time program and consequently reduce its repetition rate. The reduction must not exceed the limit at which the performance of the photo-carriages is affected. This consequently limits the length of the new programs that may be executed in real-time. The length of the scan program, for example, nearly reaches this limit and it is not possible to include into the program complicated functions to be performed during scanning, e.g., lengthy manipulation of model coordinates.

It may be possible in some instances to

overcome this problem partly by arranging part of the new real-time requirements as an additional branch routine (calculated every sixth time). This technique was followed for two-media photography. Even then, the smooth performance limit was reached.

It can, therefore, be concluded that for more sophisticated programs hardware modifications are necessary.

Acknowledgement

The research for this paper was possible through grants from the National Research Council of Canada, Canadian Defence Research Board and the Canadian Donner Foundation.

References

1. Konecny, G., "The Analytical Plotter and its Interfacing with an IBM 360–50 System", *Bildmessung und Luftbildwesen*, Vol. 36, 1968, No. 2, pp. 107–116.
2. Edmond, J. J. and Lux, F. B., "Automatic Analog Analytical Plotter," Bendix Technical Journal, Vol. 1, No. 2, 1968.
3. Forrest, R. B., Moore, W. H. and Scarano, F. A., "Automatic Comparator," Bendix Technical Journal, Vol. 1, No. 2, 1968.
4. Helava, U. V., "Analytical Plotter," Canadian Surveyor, Vol. XVII, No. 2, June 1963.
5. Helava, U. V., "A Family of Photogrammetric Systems," *Photogrammetric Engineering*, Vol. XXXIII, No. 1, January 1967.
6. Helava, U. V., Whiteside, A. E. and Matherly, C. W., "New Automatic Analytical Stereoplotter," Bendix Technical Journal, Vol. 1, No. 2, 1968.
7. Johnson, E. C. and Kamm, V. C., "Computer and Programs for the AP-C Analytical Stereoplotter," Canadian Surveyor, Vol. XVII, No. 2, June 1963.
8. Parenti, G., "Orthophoto Printing with the Analytical Plotter," *Photogrammetric Engineering*, Vol. XXXIII, No. 4, April 1967.
9. Konecny, G., and Masry, S. E., "Machine Language Programs and Routines for the Computer of the O.M.I.-Bendix Analytical Plotter AP-2C, Part 1, Research Report, August 1969.
10. Forrest, R. B., Whiteside, A. E., and Hornbuckle, J. A., "Programming Computer-Controlled Photogrammetric Instruments," Bendix Technical Journal, Vol. 1, No, 2, 1968.
11. Tewinkel, G. C., "Water Depths from Aerial Photographs," *Photogrammetric Engineering*, Vol. XXIV, No. 6, Nov. 1963, pp. 1037–1042.
12. Van Wijk, M. C., Discussion on "Water Depths from Aerial Photographs" by G. C. Tewinkel, *Photogrammetric Engineering*, Vol. 30, No. 5, 1964, p. 647.
13. Rinner, K., "Problems of Two-Medium Photogrammetry," *Photogrammetric Engineering*, Vol. XXXV, No. 3, March 1969, pp. 275–282.
14. Shmutter, B., Bonfiglioli, L., "Orientation Problems in Two-Medium Photogrammetry," *Photogrammetric Engineering*, Vol. 33, 1967, pp. 1421–1428.

Symposium Proceedings Available

	Price to Members	*Price to Nonmembers*
Annual March Conventions		
1968—428 pages, 1969—415 pages, 1970—769 pages; 5.8×8.5 inches	$2.50	$ 5.00
Fall Conventions		
Portland, Oregon, 1969. 363 pages, 39 papers Denver, Colorado, 1970. 542 pages, 33 papers 5.8×8.5 inches	2.50	5.00
Remote Sensing		
21 selected papers, 1966, 290 pages, 8½×11 inches	2.00	3.00
Computational Symposium, 1970		
32 papers, 247 pages, 5.5×8.5 inches	5.00	10.00

American Society of Photogrammetry, 105 N. Virginia Ave., Falls Church, Virginia 22046.