OPTIMUM SOFTWARE ARCHITECTURE FOR AN ANALYTICAL PHOTOGRAMMETRIC WORKSTATION AND ITS INTEGRATION INTO A SPATIAL INFORMATION ENVIRONMENT

J. OLALEYE

October 1992

GEODESY AND GEOMATICS ENGINEERING
UNB

TECHNICAL REPORT
NO. 162

PREFACE

OPTIMUM SOFTWARE ARCHITECTURE FOR AN ANALYTICAL PHOTOGRAMMETRIC WORKSTATION AND ITS INTEGRATION INTO A SPATIAL INFORMATION ENVIRONMENT

James Olaleye

Department of Surveying Engineering
University of New Brunswick
P.O. Box 4400
Fredericton, N.B.
Canada
E3B 5A3

October 1992

PREFACE

This technical report is a reproduction of a dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Surveying Engineering, March 1992. The research was supervised by Dr. Wolfgang Faig, and funding was provided partially by the Canadian Commonwealth Scholarship and Fellowship Committee and by the Natural Sciences and Engineering Research Council of Canada.

As with any copyrighted material, permission to reprint or quote extensively from this report must be received from the author. The citation to this work should appear as follows:

Olaleye, J. (1992). *Optimum Software Architecture for an Analytical Photogrammetric Workstation and its Integration into a Spatial Information Environment.* Ph.D. dissertation, Department of Surveying Engineering Technical Report No. 162, University of New Brunswick, Fredericton, New Brunswick, Canada, 228 pp.

ABSTRACT

This study has carried out a thorough investigation into the optimum software configuration for an analytical photogrammetric workstation. The options available for performing basic station's tasks such as mensuration, triangulation, stereomodel definition, stereomodel restoration and digitizing are explored. Using the **R**-space and the **C**-space concepts, a new methodology called the ARDOVS relations of vector spaces is developed. This enables the derivation of simplified vector-oriented algorithms for implementing the software tools for the workstation. Eight software designs are implemented on the DSR-11 through the integration of the basic processes into different operational systems. An empirical cost function and a set of constraint functions in terms of automation content, speed of operation, production cost, and achievable accuracy of the derived vector data are developed for a mathematical optimization scheme designed to select the optimum workstation design.

The investigation showed that there are differences in the performance of the software design possibilities for a photogrammetric workstation depending on the requirements for project time (speed of operation), operator involvement, and accuracy of the derived spatial information. However, the optimum software architecture for a workstation, in which minimum project time, minimum operator participation, and high spatial fidelity of the vector data are jointly critical is the BT_570 architecture, a comparator-based design which includes a bundle triangulation, supplying the object space control points (including minor control points) to a stereomodel definition unit which employs the automated relative and absolute orientation computations to recover the model restoration data. Nevertheless, when the accuracy requirement may be somewhat relaxed, then the software design (BT_000) which employs bundle triangulation supplying only the exterior orientation parameters of the camera (image space) to a definition unit which recovers the model restoration data from these parameters will be most economical.

Furthermore, the necessity to minimize data acquisition cost by eliminating expensive intermediate data transfer which often requires data conversion from one format to another, a common feature of most of the CAD/DBMS systems currently in use, requires a direct integration of the photogrammetric workstation to a GIS. In this work, the integration of the workstation to the CARIS GIS for real time operational communication is achieved through the CARIS server interface and the Mailbox technology in the VAX VMS operating environment which allows many programs to operate in parallel and to share data synchronously.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Table Page

ACKNOWLEDGEMENTS

The author gratefully acknowledges the encouragement and support of Dr. W. Faig, who supervised this study. His suggestions and comments on various aspects of the work have been helpful. My profound thanks are due to Dr. E. Derenyi for carefully reading the thesis and making valuable comments; to Dr. Y.C. Lee for his assistance in solving some system problems in the early stages of the work; to Dr. D.E. Wells, Dr. P. Vanicek, Dr. R.B. Langley, Dr. A. Kleusberg and Dr. A. Chrzanowski for their excellent lectures that have improved my knowledge in many areas of Surveying. My thanks are also due to Ms. Wilson for her help on the analytical plotter and in attending to my many requests for special privileges in the use of the laboratory facilities; Mr. Ed Light, for his help in solving system problems at the final stages of the work, and to the Universal Systems Limited (USL) for providing the literature on the server interface.

I owe a depth of gratitude to Professors F.A. Fajemirokun and O. Ayeni who have jointly influenced my career in academics by way of encouragement and support. I am grateful to all members of staff of the Department of Surveying of the University of Lagos for taking on extra teaching load for many years to enable me to undertake this study, and to the University of Lagos for granting me study leave. The financial support from the Canadian Commonwealth Scholarship and Fellowship Committee is gratefully acknowledged.

Finally, I like to offer my greatest appreciation to my children, Seyi and Folake for enduring my absence at such early stage of their life. The dedication of the work to them is meant to express my sincere love for them. I also like to thank my wife Cecilia, whose company and helpful assistance have seen me through the difficult times of the study.

DEDICATION

This thesis is dedicated to my Children - Seyi and Folake.

CHAPTER 1

Introduction

1.1 Initial Remarks

Recent advances in computer technology and subsequent developments in both photogrammetric and cartographic methodologies, have brought us into an age (an information age), where the integration of the various processes of mapping is becoming increasingly important; not only to use new technological devices to improve the procedures, but also to expedite the rapid and economic production of maps in order to meet the growing demand for geo-information products. The rapid rate of development in all aspects of human life has necessitated the need for frequent, adequate, and up-to-date information about the resources and ecological changes of the earth for prudent management and control to the benefit of mankind. Population growth, urban expansion, shortages of resources such as food and clean water, especially in developing countries, frequent occurrence of natural disasters such as flooding, subsidence, and earthquakes all underscore the need for complete and accurate supply of basic knowledge of the earth's surface and its current state for proper planning. Old systems and methods of mapping are no longer able to meet the pressure and to achieve the high standards of modern geo-information production.

In all aspects of mapping today, as in all areas of human endeavour, the inventions of science are being used in one way or another either to improve the traditional procedures in order to make maps more conveniently or cheaply, or to evolve new procedures for doing things that could not be done in the past in order to keep pace with product demand, and variety. For example, several decades ago, triangulation baselines were measured by days or months of catenary taping; today, a goedetic engineer involved in such tedious work would be surprised by the convenience, speed, and precision of the global positioning system (GPS); a plane surveyor at that time who used plane tabling for detail surveys would marvel at the magic of modern electronic tacheometers, called total stations. Not only that, digital technology has produced orbiting satellites that have expanded land cover mapping to the entire globe, and improved remote-sensing systems that extend the photo-interpreter's vision well beyond the small part of the spectrum visible to him, and also provided tremendous assistance in interpreting such images. In a similar manner, photogrammetric mapping procedures have undergone remarkable innovations due to the emergence of new technology. The invention of the analytical plotter about three decades ago was received with pessimism [Blachut 1963]; however, advancements in electronic technology have made this programmable hardware instrument an economical tool for mapping, thereby rapidly liquidating the reliable and popular work horses of mapping, the analogue plotters [Blachut 1979, Yzerman 1979, Niedzwiadek 1980, Petrie 1990a,c]. Nowadays, almost all photogrammetric problems are being solved analytically. The high computer content of the system has made some of the operations easier and enabled the mathematically correct treatment to be used at critical stages of the mapping process. Moreover, developments in computer technology and digital image processing have allowed a complete change in some of the procedures, for example, it is now possible to use digital images as input data instead of the conventional photograph. This has subsequently opened up a new horizon in the mapping industry. "From Analytical to Digital"; was the title of the proceedings of the Symposium of the International Society for Photogrammetry and Remote Sensing (ISPRS) held in Finland in 1986, and it emphasises the fact that interest is shifting from analytical to digital image photogrammetric systems. Interestingly, because scientific development is not static, the popularity of the analytical plotter is now being challenged by the invention of so called "digital systems", which are generally believed to have great potential for total automation of the mapping process [Dowman 1977, Case 1981, Helava 1988]. However, because of their limited accuracy and high cost, their use in mapping is not yet widespread [Allam 1982, Helava 1988, Petrie 1990c].

Equally noteworthy are the improvements made in the presentation of map information; experiencing the impact of digital technology, cartographers have abandoned the pen and paper technology and have adopted digital methods, making use of electronic pulses and files [Goodchild 1988, Tomlinson 1988, Petrie 1990b, Parker 1990]. The tedium of map construction by scribing is over; now, symbolization and editing of graphic elements are done electronically; map elements can be selectively displayed on graphic screens with great visual appeal and clarity. Moreover, the realization of the ability of the computer to store, retrieve, display, and even to use maps, has lead to the evolution of the modern digital technique of processing map information [Zarzycki 1978, Greve 1980, Monmonier 1982, Lambourne and Sutherland 1990, Campbell and Egbert 1990], thus, creating the all-pervading "Geographic Information Systems" (GIS) technology.

From the consumers" viewpoint, the use of map information has witnessed significant improvement due to the development of digital information processing which has immensely increased the value of the map as a decision-making tool. The ready availability of the computer, its reduced cost, increased power and ease of use, coupled with its ability to extract geographic information from digital maps with speed and precision, and even to analyse this information to propose solutions to problems has caused users to demand maps in digital form [Jaksic 1976, Leatherdale and Keir 1979, Case 1981, Petrie 1981, Green 1990]. Nowadays, spatial information is being used for large scale data analysis to solve problems in diverse areas, such as environmental monitoring, regional development, land use planning, facilities management and construction, exploration, and management of the earth's resources. Evidently, computer technology has not only improved the production of the map, but also improved its use, thereby increasing the demand for spatial databases.

Perhaps, the following extract from an advertisement in the February 1991 edition of the Journal of Photogrammetric Engineering and Remote Sensing vividly illustrates the typical changes taking place in the mapping industry today (emphasis added). "It is such a shame to be left behind in today's Digital world. Yesterday's painful investments in fine analog stereoplotters are suddenly Antiques. Many long-term clients are out-of-reach now because they want a Disk instead of a manuscript. Maps are now Files and the good old cartography is now GIS... You could try to convince your Banker for another long-term loan to buy one of those New machines and start all over again, or look into the affordable digital mapping system for analogue stereoplotters" (International Systemap Corp.1991). From this quotation, a number of points are apparent. Firstly, digital technology has so much revolutionised mapping practices that practitioners need to learn new techniques to remain in business. Secondly, users of maps now want digital rather than paper maps. Thirdly, old mapping tools are rapidly getting out-of-phase with today's mapping methodology. And fourthly, because the new instruments are expensive, manufacturers are patching up the old ones, thereby assisting users to apply new technology to old procedures, a process which may be justifiable on grounds of economy; however, the rapidly expanding improvements in both hardware and software technologies are making the new methodology more affordable.

Nevertheless, all of these points underline the fact that, although photogrammetric and cartographic mapping procedures have long been in place, there is a need now for a renewed approach to map production in the light of modern developments. Application of new technology to the old procedures would just not meet today's demands for faster delivery, greater accuracy, lower production cost, and varied product types. The sequential and often isolated or independent method of producing maps in times past must give way to coherently integrated and ergonomically optimized mapping systems.

Technically, the process of integration implies optimization of the photogrammetric production line. It concerns hardware and software integration, information and data handling compatibilities for efficient communication and quality control, as well as proper planning of operational procedures to achieve a balance between performance and economy of the mapping system. In recognition of these facts, the International Society for

Photogrammetry and Remote Sensing (ISPRS) has therefore stressed the importance of integration and optimization of photogrammetric systems at its 1988 Congress in Kyoto, Japan, where for Commission II Working Group six, the study of the development of integrated systems and their optimization was recommended [Makarovic 1988]. Included in the terms of reference for further study are: (1) context and classification of integrated systems, (2) design and development of new systems, (3) hardware architectures and components (4) software structures and modules (5) interactions and human factors, (6) performance and reliability of existing systems (7) experimental tests of accuracy and efficiency of existing systems [ibid].

This study is concerned with the development and optimization of the software components of such an integrated system. It is part of the research activities of the Canadian Laboratory for Integrated Spatial Information Research and Engineering (CanLab INSPIRE) at the Department of Surveying Engineering, at the University of New Brunswick (UNB). The study is aimed in part, at achieving an optimal operational configuration for the Kern DSR-11 analytical plotter as a photogrammetric workstation and in part at achieving the integration of the workstation to the Computer-Aided Resource Information System (CARIS), from Universal Systems Limited (hereinafter referred to as CARIS GIS) for real time operational communication.

The approach followed for the optimization of the photogrammetric software design is to look at the modern analytical photogrammetric system as a collection of tools and procedures for transforming photographic images into object space information. Taking the hardware component for granted, the major tools of such a system are software-based subsystems employing mathematical algorithms to perform designated tasks according to some chosen procedural sequence. We then consider all the software modules which execute the basic tasks such as the measurements of image coordinates, triangulation, model resolution, model restoration, and digitizing as making up the system's engine. Because many approaches are available for the execution of these basic tasks, different configurations resulting from their interplay will constitute different system engines which will rate differently according to some specified performance criteria. The optimum configuration is therefore to be selected from among a number of alternatives using a scientific approach. The selection criteria of minimum operator participation, minimum project time (or speed of operation), and accuracy of spatial data are applied. It is noted however, that in general, the software suite needed for such an investigation is not always available in one location. Therefore, several packages have been developed to supplement those originally provided with the analytical plotter.

On the other hand, the approach for achieving the integration of the workstation into the CARIS GIS environment is to address the issue of data structure disparity between the photogrammetric unit and the CARIS GIS; we then consider the pros and cons of using either an active or a passive interface between the two units. With consideration for speed of operation, particularly as the human operator is involved in the process, we settled for a passive interface which, while serving as a bi-directional communication channel, does not decode or interpret the information but achieves an integration between the DSR-11 as a workstation and the CARIS GIS. Indeed, this work constitutes the first effort made to assemble an optimized analytical photogrammetric data capturing system using such two top-of-the-line products.

1.2 Motivation for this Study

The motivation for this study has developed from the author's experience at using the DSR-11 analytical plotter at UNB for special studies in photogrammetry, a graduate level course which deals mainly with camera calibration and close range techniques. It was found that while it was easy to carry out relative and absolute orientations on the plotter using the Kern software, tasks such as photo and model coordinate measurements were not so easy to perform. Because the programs for these tasks were not available, unusual methods were being used. For example, to measure photo coordinates, one had to perform interior orientation and relative orientation. Any point whose photo coordinate was required had to be included in the relative orientation process. Also, to retrieve the coordinates, one had to edit the relative orientation data file to delete the unwanted entries and attach point numbers, and this must be done for each stereopair otherwise, the previous measurements get overwritten. Not only that, one was limited to a maximum of 25 points for the relative orientation, and this meant a limitation on the number of points that could be measured. Convinced that this was not the proper procedure, the author developed a new package soley for measuring photo-coordinates with the DSR-11. This program made it possible to acquire data for applications other than model orientation, and this was the beginning of the author's interest in the analytical system. However, with the implementation of the measuring program, different inadequacies of the system began to become apparent. Because applications other than model orientations were not available on the DSR-11, often, the measured coordinates had to be transferred to other computer systems where they were used for other applications such as triangulation and camera calibration.

Meanwhile, the CanLab INSPIRE (CanLab) research programme had set as one of its objectives the integration of the DSR-11 with the CARIS GIS so that they can communicate preferably in real time. The resulting system was required to have the capabilities for all the operations necessary to produce a digital map file from photographs (Lee 1988). Having implemented the coordinate measuring package, the author became interested in this research program, viewing it as an opportunity to acquire a working knowledge of the new mapping technology which is of immense advantage to a developing country like the author's. He then set about finding out what was available and what was required for the task. This revealed a gross inadequacy of the existing software configuration to achieve a useful integrated system. For one thing, the setup could only process a project for which control extension had been done. The process of initial data reduction by triangulation could not be performed because as said before, the program was not available on the system, although provision was made for it in the Kern software menu. Furthermore, the data structure of the MAPS200 digitizing package which was available was incompatible with that of the CARIS GIS, thus, its output could not be directly connected to the CARIS GIS for real time operation. It then became apparent that integrating the present setup of the DSR-11 with CARIS GIS would not produce an efficient system. What was needed was a total mapping system or a workstation that could handle all the tasks required for mapping from the photograph to the final digital map, all at one location. This was the challenge the author set out to accomplish. Towards achieving this goal, two questions were posed: (1) what software configuration is best to make the DSR-11 an efficient and cost effective workstation, providing highly accurate spatial information with minimum operator participation, (2) what type of interface will be the most suitable for linking the workstation to the CARIS GIS so that they communicate in real time with minimum delay ?

Searching the literature for specific designs for a photogrammetric workstation and its integration into a GIS, it is evident that different organizations use different configurations determined by the architecture of both the hardware and the software tools they have (Jaksic 1980, Reece & Kleinn 1980, Detwiler 1980, Kreiling 1982, Hobbie & Rudenauer 1984, Chapuis et al 1988, Kratky 1988, Cogan & Polasek 1988). Forgetting about the interface for the moment, and talking about the configuration for the photogrammetric workstation, two general types of system configuration are identified. On the one hand, there are systems which are organized around computer-supported analogue plotters [Dorrer 1976, Makarovic 1976, Pillmore et al 1981, Petrie 1972, 1990a]. These systems do not include the triangulation task in their configurations but done separately on another computer, and only the extended control points are brought to the system's environment to be used for single model reoriention by the operator. Since the hardware of the configuration is not computer-controllable, systems of this type have limited configurational possibilities. On the other hand are systems which employ the analytical plotter, and in addition to the common orientation programs, also include or may include a rigorous block triangulation package which may be used in the system's environment. As will be demonstrated later, because the hardware component of this type of system is computer-driven, the block triangulation package may be used in a variety of ways for different configurations [Detwiler 1980, Jensen and Kleinn 1980, Kreiling 1982, Chapuis et al 1988, Cogan and Polasek 1988]. Yet, some systems use non rigorous block adjustment procedures to reduce the demands on the host computer [Riley 1980]. Therefore, for this type of system, the configuration options are not as many, since the triangulation package can only supply one type of output. However, in most cases, the analytical plotter system is delivered with just the basic orientation tools. The application programs are purchased by the user often from a company other than the one that supplied the plotter; for example, the aerotriangulation package may be supplied by company A, while other components of the system are delivered by company B. The result is that the user gets an assemblage of different components which, though functional, but is nevertheless, unoptimal. This invariably reduces the advantages of the analytical system to mere stereotyped procedures. The following sequence of operations for using the original configuration of the DSR-11 at UNB for an experimental block of 4 photographs is given to illustrate the above point. This configuration is typical of the setups on many of the so called workstations.

A manuscript was to be prepared from a block of four photographs with a few available ground control points. The Kern DSR-11 analytical plotter running on the microVax-II minicomputer was used. To extend the control, photo-coordinates were measured on the plotter and downloaded to the mainframe computer on which the 'best' aerotriangulation package was installed. The extended control was then taken back to the plotter in order to orient each model for plotting. The set-up of the orientation scheme on the plotter required that the photo-coordinates be remeasured for each model, from which the relative orientation parameters were computed, then the model coordinates were measured, and the absolute orientation data was computed to restore each model for compilation.

The point to note here is that, despite the use of the 'best' aerotriangulation package, this configuration is deficient for two reasons: (1) the transfer of aerotriangulation data from one system to another is not ideal; (2) remeasurements of coordinates for reorientation is an avoidable duplication. It is noted however, that the above arrangement does not represent a hardware limitation of the DSR-11, but rather the limitation of the application software configuration available for this particular set-up [Kern unpublished documentation 1988]. Now, let us reconfigure this system by bringing the triangulation package to the environment of the system. It is not hard to see that this immediately removes the first problem in the previous configuration. Introducing yet another innovation, let the measurements made during initial mensuration be saved for later use in an analytical reorientation [Detwiler 1980, Dorrer 1981, Kreiling 1982, Hobbie and Rudenauer 1984, Dequal 1984]. The relief brought to the operator by this configuration is better experienced than imagined. However, other approaches are also possible, for example choosing a triangulation scheme whose output data could be used directly to set up models without remeasurements or recomputation other than for interior orientation in the case when the photographs have been removed from the stage [Slama 1982, Helava 1982]. This configuration also promises remarkable improvement over the base configuration. Evidently, the choice of the triangulation scheme, and its integration with the orientation processes affect the optimal performance of the photogrammetric system.

For many years, the point-coordinate discrepancies have traditionally been used to assess the performance of block triangulation schemes. However, in the modern systems concept, today's analytical plotters are typical state-of-the-art mapping workstations in which triangulation is only a tool (a very important one though). And in the context of integration, such a workstation is considered as an input-output system whose performance can be optimized and tested as a unit. Against this background, the conclusions from past investigations about a single component of the system do not necessarily translate into criteria of optimality of the entire system. Therefore, new tests are urgently needed. Such tests will not be based on the 'out-of-context' triangulation accuracy criteria alone or of any other component for that matter, but also on ergonomic factors of the system of which it is a part. Such tests will reveal the abilities of the various block triangulation procedures for example to function economically and efficiently in an integrated environment. Todate, partly because integrated systems are a new concept in mapping, such tests are not yet available in the literature. The search for a specific configuration for our analytical workstation has lead to the discovery of a yawning gap for such a concept in the literature. This study is therefore designed in part to fill this gap. It is directed towards investigating the optimal operational configuration of the photogrammetric production line for the Kern DSR-11 plotter.

Although the importance of optimization of analytical systems has been recognized [Jaksic 1980, 1984a, 1984b, Slama 1982, Makarovic 1982, Fritz 1984], no significant studies have been made in this direction. And as Slama [1982] rightly pointed out, an analytical system, if optimized has effects on almost all operations including project planning and final data extraction. However, research efforts up to now have concentrated on triangulation techniques for analytical plotters [Gruen 1982, 1984, Kratky 1982, Marton 1984, Holm 1988, Jacobson 1988, Corcodel 1988]. The general dirth of material on this important topic in the literature is a further indication that investigations have not been conducted in this area.

Concerning the interfacing of the photogrammetric workstation to the GIS environment, it is observed that direct integration to a GIS is not a common feature of most of the existing workstations. However, almost all have provisions for making hardcopy maps or optionally digitizing into a file for later processing by another program [Zarzycki 1978, Case 1981, Green 1990]. More recently, in response to users' demand for digital data bases, some of these systems are being linked to Computer-Aided Design (CAD) and

Data Base Management System (DBMS) packages to produce digital map files in various formats [Klaver 1984, Zutter 1984, Hodgson et all 1989, Roberts 1989, Green 1990]. However, while the use of such tools represents an acceptable way to produce data bases, the full potential of the analytical plotter is realized most dramatically when operated as a workstation in an integrated spatial information environment. In recognition of this fact, a relatively new trend has started in which manufacturers of photogrammetric systems now also produce and integrate their own GIS. The first such system has been developed by the Wild company and was named System9 [Bonjour et all 1988, Parker 1990]. The configuration of the system includes an editing station (system9-E), a table digitizing station (system9-D), an analytical plotter station for digitizing and editing (system9-AP), and a file server module (system9-S). However, no information is provided as to the internal details of the photogrammetric workstation and its interfacing to the GIS module of system9. Nevertheless, the significance of this package is the emphasis it places on the importance of direct integration of the photogrammetric workstation to a GIS.

To close this section and to further clarify the motivation for this study, the following synopses are presented:

1. though there are many analytical plotter systems in use, each installation operates a configuration determined by the architecture of its hardware and software. In general such configurations are an assemblage of components from different vendors and are not optimized for system operation. The attempt to install the most efficient configuration for the DSR-11, has been the motivating factor for this investigation designed to choose from among the various possibilities, the most optimal in terms of cost, speed of operation, minimum operator participation and accuracy of the derived spatial information.
2. It is now generally accepted that the key to unlocking the possibilities of a GIS is a good and up-to-date database which is provided primarily by photogrammetric methodologies. To meet modern requirements, an analytical workstation must have

the capability to produce map data in computer compatible form and to communicate with a spatial information processing system. However, while the use of CAD/DBMS is an acceptable way to produce data bases, the full potential of the analytical plotter is utilized only when linked directly to a GIS. And this was the motivation to link the Kern DSR-11 analytical plotter to the CARIS GIS.

3. personally, a major motivation for this study is the fact that developing countries are in an ever increasing need for maps to support their development programs. The author will have a feeling of accomplishment for building the necessary technical skill and expertise to make a substantial contribution towards improving his country's geo-information production.

1.3 Scope and Objective of this Study

In order to establish a frame of reference for discussing an integrated spatial information system vis-à-vis the objectives of this study, the entire mapping process is considered here as a system which comprises two main subsystems namely: (1) input subsystem, and (2) output subsystem.

1. The input subsystem consists of all photogrammetric operations, hardware and software for data mensuration and processing, up to the digitizing of features. This will be called the photogrammetric workstation or simply the workstation.
2. The output subsystem consists of all cartographic procedures, hardware and software for editing, symbolization, graphic display and printing, and possibly data management. This will be called the CARIS GIS or simply the GIS.

Viewed within this framework, it is obvious that to achieve an integrated photogrammetric mapping system, three phases of integration must be addressed:

1. internal integration of the input (photogrammetric) subsystem
2. internal integration of the output (digital cartographic) subsystem

3. external integration of the input subsystem with the output subsystem.

On the one hand, internal integration concerns the ergonomics of the operation of the various components, both hardware and software, of either the input subsystem or the output subsystem. On the other, external integration involves the establishment of a communication channel between the input subsystem and the output subsystem so that there is an uninterrupted flow of data and information between them, preferably in real time. From this classification, this work will include all the processes involved in the optimization of the photogrammetric system, that is, the internal integration of the input subsystem. And towards this end, the study will implement a number of software designs through the development and integration of the basic photogrammetric processes into different operational configurations. The research will then conduct an empirical investigation to compare these configurations using both real and simulated data. The criteria of automation, speed of operation, production cost, and achievable accuracy of derived data are proposed to be used to examine these systems through a detailed systems analysis involving the use of mathematical programming. The results are expected to lead to the selection of the most efficient software design for the workstation. Furthermore, this work will also include the external integration of the workstation to the CARIS GIS, for real time operational communication. For this purpose, menu-driven software packages are developed to integrate the data capture and the cartographic modules into a unified spatial information system.

However, this study does not include matters related to the internal integration of the CARIS GIS. Such issues are beyond the scope of this investigation. The CARIS GIS package is used as a "black box" in all of the developed applications.

For clarity, the objectives are enumerated as follows:

1. internal integration - to investigate the optimal operational configuration of the photogrammetric production line, including the type of block triangulation

procedures and their integration with the orientation processes, with a view to minimizing duplication of effort in the production process.

2. software development - to generate all the necessary software packages in the VMS environment. This includes packages for mensuration and processing of data in the photogrammetric system; this is necessitated by the fact that programs to achieve the above objective were not available at the commencement of this study.
3. external integration - to develop the software packages necessary for interfacing the Kern DSR-11 analytical plotter to the computer-aided resource information system (CARIS), so that spatial data can be acquired and edited in real time.

The rest of this thesis deals with the details of the work done. Thus, chapter 2 presents the integrated spatial information system in the context of a generalized systems concept. The notion of an information, a spatial information and methods for its collection and processing are addressed. This is designed to provide a framework within which subsequent work is developed. And in this manner, the analytical photogrammetric system integrated into a GIS is presented as a workstation for the collection and manipulation of 3D spatial and attribute information. In chapter 3, the vector space approach to the mathematical formulations which form the bedrock of modern analytical systems are discussed. In chapter 4, the various software configurations are developed; their composition, their optimization and results are presented in detail. Chapter 5 describes the procedure employed to achieve the external integration; while chapter 6 presents a description of all the packages developed by the author and gives the details of their interaction. In chapter 7, conclusions from this investigation are presented together with some closing remarks.

CHAPTER 2

Concepts and Components of a Spatial Information System

The integration of the process of collection and the cartographic treatment of spatial data naturally results in the evolution of a bigger and more complex system. Yet, the expanding growth of digital technology, which continues to broaden the range of possibilities and alternatives in both the collection and processing devices, and the desire to take advantage of these alternatives results in increased system complexity. Thus, the optimization of such a system after it is developed becomes an important task. However, to achieve integration and optimization, the system must first be thoroughly understood. Its components and their functions must be clearly analyzed. In this chapter, a detailed meaning of spatial information, and methods for its collection and processing are explored. Also, the idea of a spatial information system is presented in the general context of systems and concept, and the methods available for constituting such a system by photogrammetric technology are discussed as background material for introducing the analytical plotter-based system addressed in the study.

2.1 Spatial Information

To begin with, the word information, in a generic sense, is a data element of some kind which is useful in some way to an individual in his decision making process [Rosove 1967, Kent 1971]. Emphasis is on the word useful, because as it is often said, one man's noise is another man's signal. A radar operator in a system of air defence who is tracking an unidentified aircraft regards radar returns from clouds as noise, not information; but to a meteorologist attempting to forecast weather, radar returns from clouds are not noise - they are the information he needs to achieve his objectives. Thus, information is not just data, it is useful data. The author's name and book-title in the index of a large library are data elements needed by a user of the library to locate a particular book on the library's shelf; a bank needs information about its customers such as name, address, account number, etc. in order to locate the record of that customer. A medical Doctor needs the records of his patients for the administration of treatment. City planning agencies need such information as the 'effects of siting a shopping mall in an area on traffic flow'. The highway engineer requires topographical information about the route of a proposed highway to evaluate alternative designs; etc. It is clear then, that information is an essential ingredient in decision making since it enables one to reach a decision precisely, or to take the proper course of action in light of competing alternatives. An important attribute of information is that it is always related to a reference system, i.e., it indicates the position or condition of an entity on some reference scale, otherwise, it is impossible to distinguish between two entities or to judge their relative significance. For example, when the weather man says the temperature is 15 ^{o} C, he gives out an information which indicates the current weather condition on a 0-100 ^{o} Celcius scale which is almost intuitively understood by everyone. The book-title and author's name in a library index represent alternative indicators for the location of a book in the library's author and title index lists. A pilot who maintains his airplane at a certain flight altitude makes use of height information in a height reference system. The reference system to which information refers may be physical, such as distance from a fixed point or line, may be numerical, such as the numbering system used for example by a bank for its customers, may be alphabetic, such as ordered names of persons, etc. However, the same entity may be referenced to one or more systems; for instance, a bank may obtain information about a particular customer either by name or by account number. In general, the reference system represents a scheme in which entities are ranked as to position, size, condition, etc.

Against the background provided by the general definition, it is a simple matter to extrapolate the definition of spatial information. In its simplest form, spatial information can be defined as data describing an entity within a space. But what is space?; Guralnik [1979] defines space as the continuous boundless expanse extending in all directions or in three dimensions, within which all things exist (see Figure 2-1). But because human imagination of space extending beyond three dimensions is practically difficult, the space in everyday usage is limited to three dimensions. And since entities within a space may change form or interact with each other in time to produce more things and events, the concept of a space-time system is introduced; and it is defined as a continuum having the three dimensions of space and that of time, in which any entity or event can be located. However as already alluded to, spatial information describes not only the locations of entities and events within a space, but also their conditions, attributes, and relationships on some reference axes. Consequently, a generalized spatial reference system must be a multi-dimensional space-time framework in which each object is associated with a set of data which uniquely describes its position within the space. It is this complex multi-dimensional dataset that is referred to as spatial information.

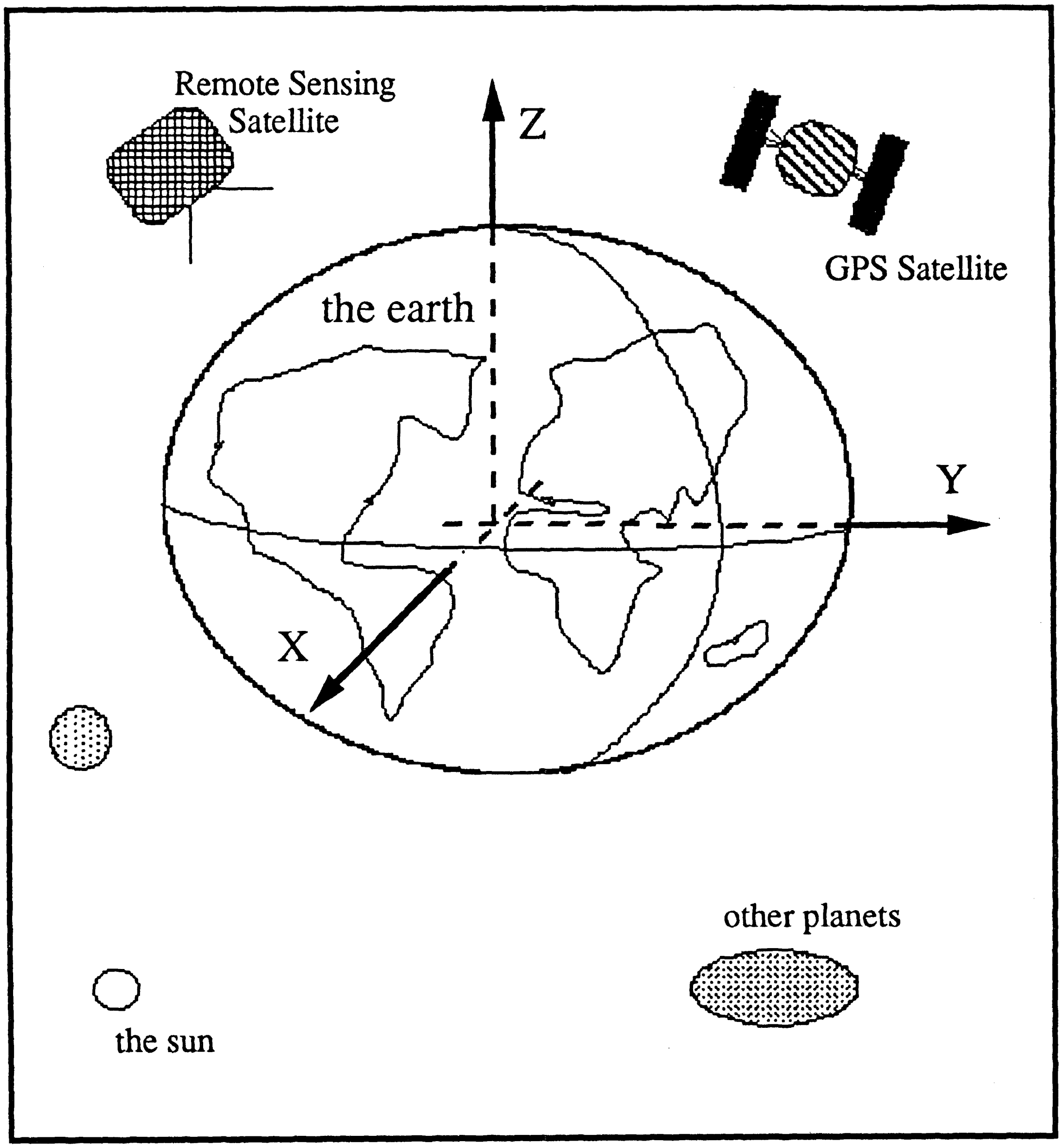


Figure 2-1. A representation of the earth-centred universe as an example of a space

However, viewed in the context of information theory, a data set is not information unless it is useful; undoubtedly, spatial information is of immense utility, and as a matter of fact, its use cuts across many professions. It is needed by the military for missile launches and other tactical operations; the highway engineer needs it for the design and construction of roads; government agencies, in formulating development policies often need to know for example, the effect that the construction of a dam in one section of a river will have on farming practices in areas further down stream; land use planners need spatial information for planning and management of land use; forestry departments often need to know the effects of deforestation on wildlife before granting permission for the felling of timbers etc. In all of these, spatial information is an essential ingredient. Yet, many land (or spatially) related decision making such as those involving distance, direction, adjacency, relative location, and more complex spatial concepts are made by individuals on a regular basis. Spatial data bases are used for these and a wide variety of other purposes. In general, a GIS package (see Fig. 2-2) enables the computer to use the data base to answer complex spatial queries, and to manipulate spatial forms to show the effects of certain actions and represent such forms graphically. Obviously, the combination of complex, critical, and spatially related decision making problems with the appearance of machines which seem to be capable of providing the information required for their solution is the cause of the great explosion of activity in the area of information system design, and of course the popularity of the term "spatial information" [Marble et al 1984, Dale & McLaughlin 1989].

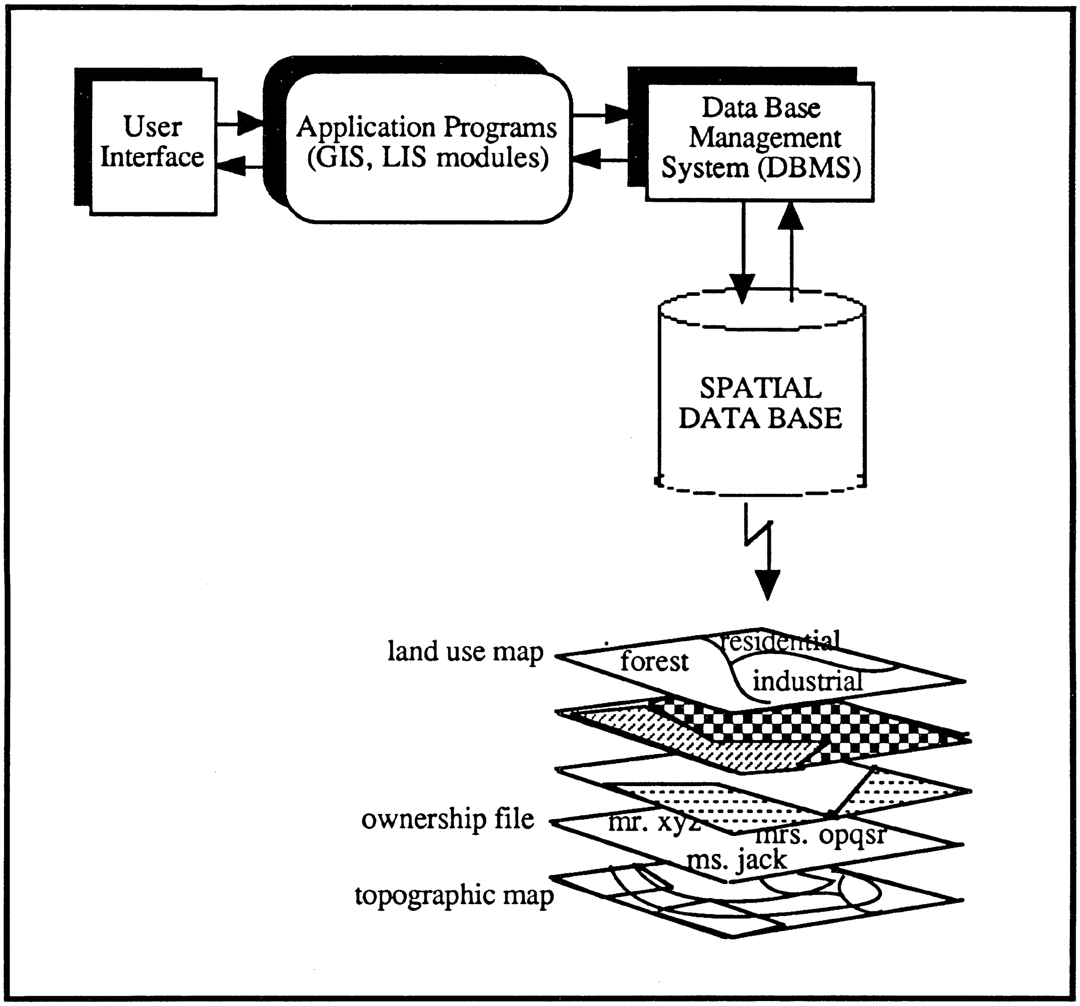


Figure 2-2: A GIS package enables the manipulation of spatial data

2.2 Collection and Processing of Spatial Information

In practice, to simplify the collection, presentation and use of such complex data, the multi-dimensional space in which the elements reside is unconsciously decomposed into smaller homogeneous subspaces (see Fig. 2-3) in which the needed information is collected and processed using simple devices and methods. For locational information, the entities of interest are considered to be composed of points, lines, and polygons, whose locations are determined based on the principles of Euclidian geometry [Marble et al 1984].

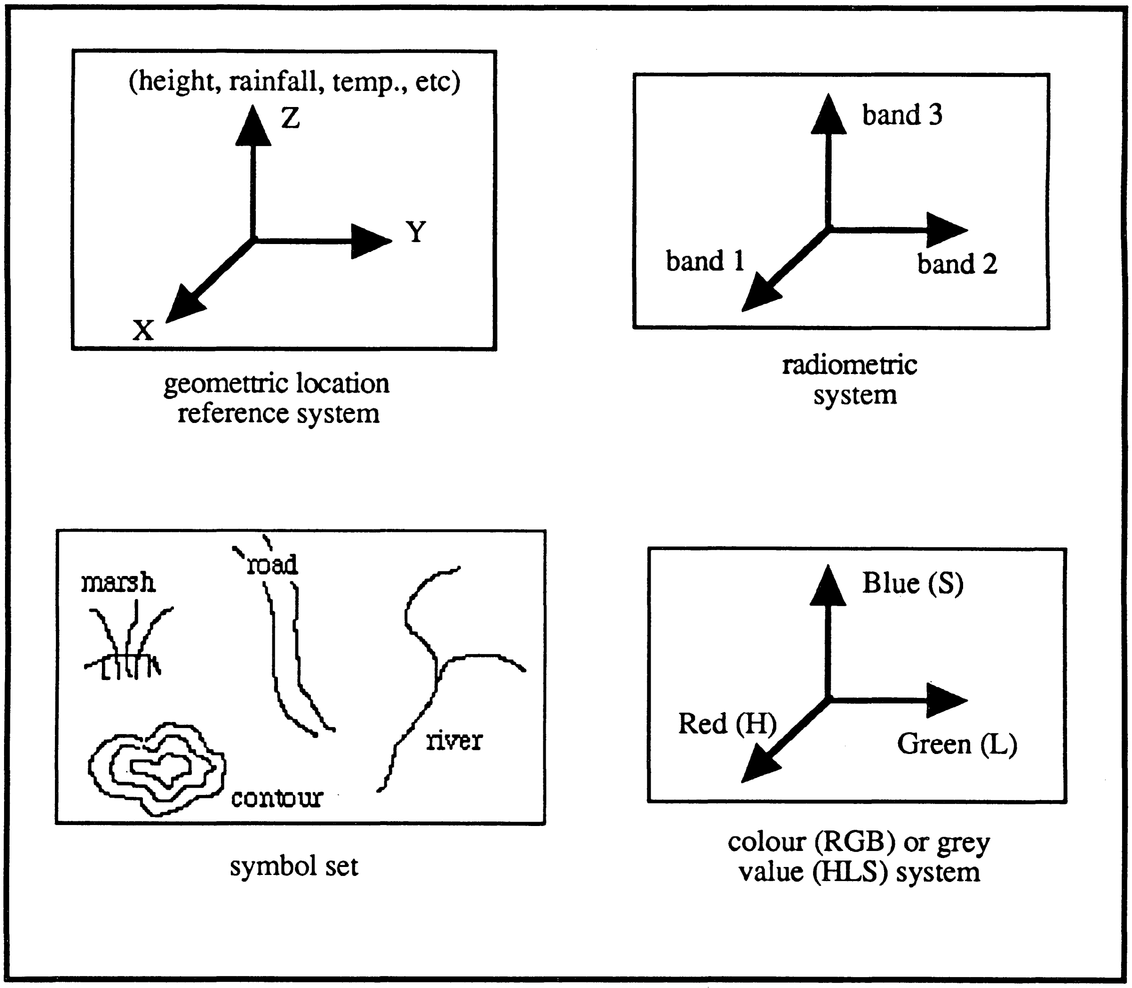


Figure 2-3: Reference systems for representing and processing spatial information.

The locations are commonly defined on the basis of a standard coordinate system (latitude, longitude, elevation, or X, Y, Z), and a time system (usually based on some observable epochs such as the passage of celestial objects through specified points within the space). From the locational information, other properties or attributes of the entities such as area, volume, slope, deformation as well as time dependent derivatives can be obtained. Usually, the information regarding attributes, and conditions of entities are referenced to some systems based on experience; for example, using well defined symbology to represent certain shapes, and different shades of grey, or radiometric values to represent certain conditions (Fig. 2-3) [Raisz 1962, Swain & Davis 1978]. In addition, the principles of graph theory are often employed to express the topological relationships or the relative location of various spatial elements [Wall 1972, Unwin 1981, Gasson 1983].

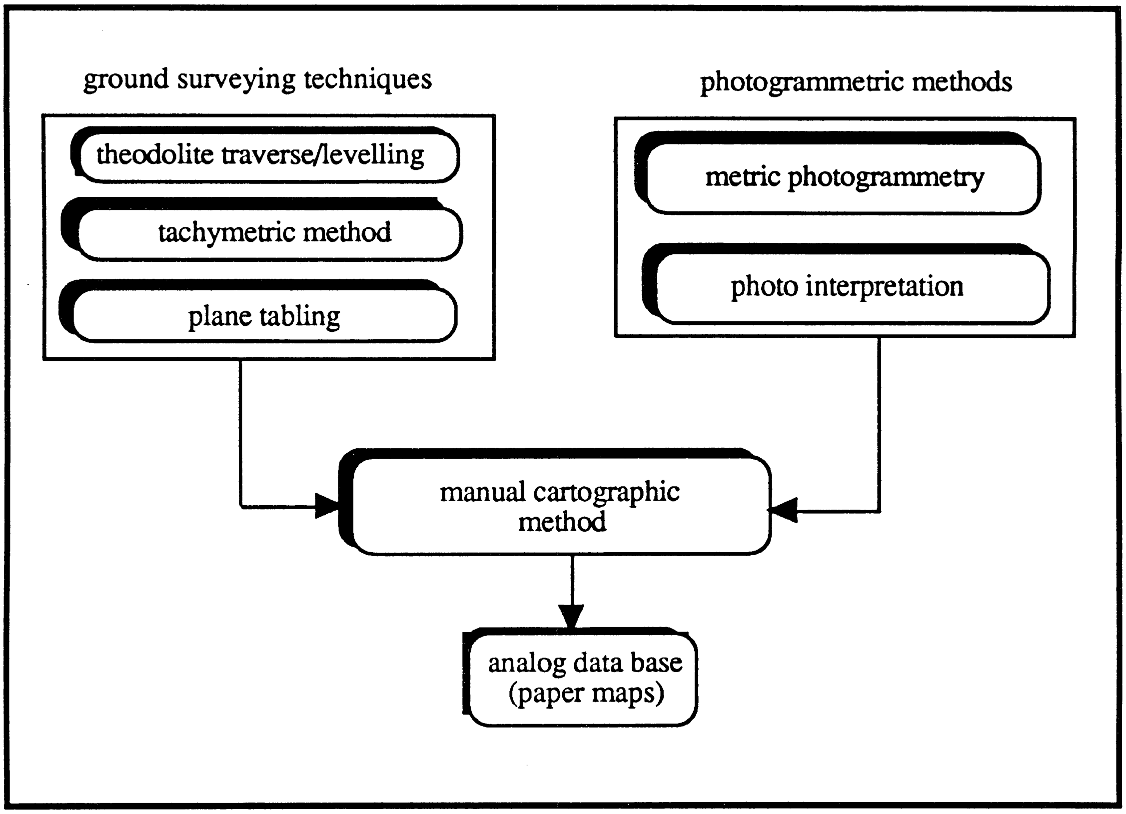


Figure 2-4: The traditional techniques for collecting and processing spatial data.

Over the years, man has developed various methodologies for collecting and processing spatial information. The early methods consisted mainly of simple linear, angular, and photographic measurements. These methods include theodolite traversing and levelling, plane-tabling, tacheometric surveying, metric and interpretative photogrammetric methods, etc. (Fig. 2-4). Correspondingly, the most common medium of storing such coordinate-based information has traditionally been the paper map, which in the language of modern information processing, is referred to as analog spatial database, produced by manual cartographic processes. In this form, retrieval and analysis of the data normally involve visual inspection of the map document coupled with intuitive analysis, occasionally aided by simple measurement tools such as the use of thread to measure map distances between spatial entities [Maling 1989]. In practice, experience has demonstrated that while it is easy to retrieve small amounts of data in this way, the retrieval of larger numbers of map elements or any attempt to examine complex relationships existing between a number of map elements is very slow [Monmonier 1982, Marble et al 1984]. Yet, analogue databases are expensive and time consuming to change when updates need to be made to the data they contain.

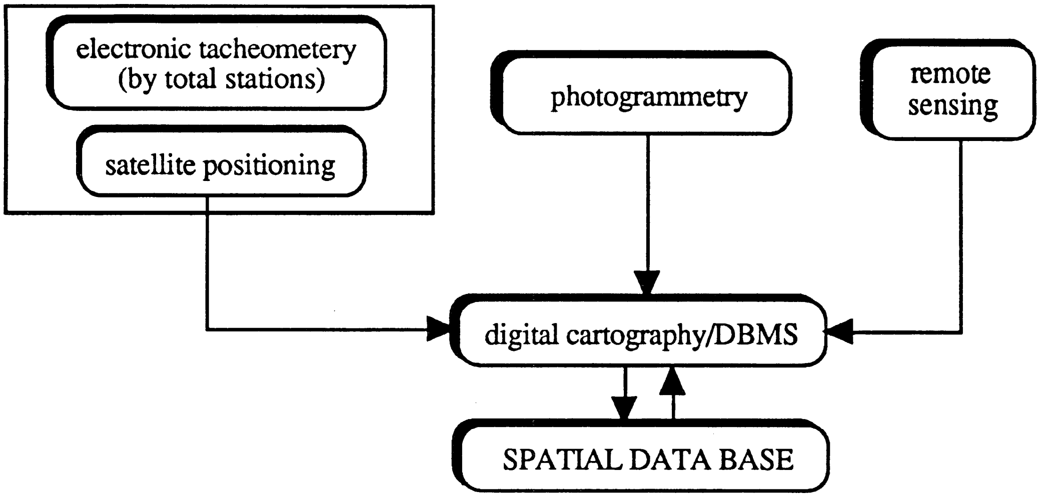


Figure 2-5: recent methods for collecting and processing spatial data.

However, in recent times, a number of increasingly precise tools and techniques have been developed to determine and to process this information with a high degree of accuracy, flexibility, and frequency (Fig. 2-5). For instance, the locations of both static and dynamic spatial entities can nowadays be determined precisely and easily using the GPS positioning technology [Richardus, 1984]; while more detailed information can be accurately and conveniently collected using the modern analytical photogrammetric methods [Ghosh 1988, Petrie 1990c]. Not only that, orbiting remote sensing satellites now provide information about the conditions of entities and of events almost as soon and for as long as they occur using radiation sensing devices [Swain and Davis 1978, Szekielda 1988]. Altogether, the rate and volume of such information are now almost too large for man to cope with and use effectively with the traditional manual methods. Fortunately, however, the development of digital information processing, and the realization of the feasibility of representing spatial data in digital form have enabled the computer not only to store and retrieve data, but also to perform complex analyses to assist man to use such data more efficiently. Today, map information is stored in digital form. The data is stored in records usually by entities and events (different types of information are usually overlaid on the positional information which is represented in coordinate form by points, lines, and polygons). These records are usually arranged orderly into larger groups, called files. The file or collection of files which contain such data records is called a database or an information base (Figure 2-2). Moreover, since the data is spatial, it is also called a spatial database.

Nevertheless, the integration of the processes of collection, storage and use of spatial data into a single system implies bringing into compatible operation, a number of independent processes. Thus, it is helpful to understand the concept of a system in general terms so that the nature and composition of a spatial information system may be better appreciated. In the next section, the concepts of a spatial information system are discussed against the background of the general systems concepts.

2.3 Systems Concept and the Spatial Information System

The Funk & Wagnalls new standard dictionary of the English language [Funk 1963], gives many definitions of the word 'system', which include the following: “an orderly combination or arrangements of parts or elements into a whole, especially, such combination according to some rational principle or organic idea giving it unity and completeness; any methodic arrangements of parts; the connection or manner of connection of parts as related to a whole, or the parts collectively so related; a whole as made up of constitutive parts". From these definitions, it is obvious that the word 'system' has many interpretations depending on the context in which it is used. It may be used for anything or a collection of things that performs a specified function. It may mean for example, a procedure, a process or its control, a network, or a computer-based data processing package.

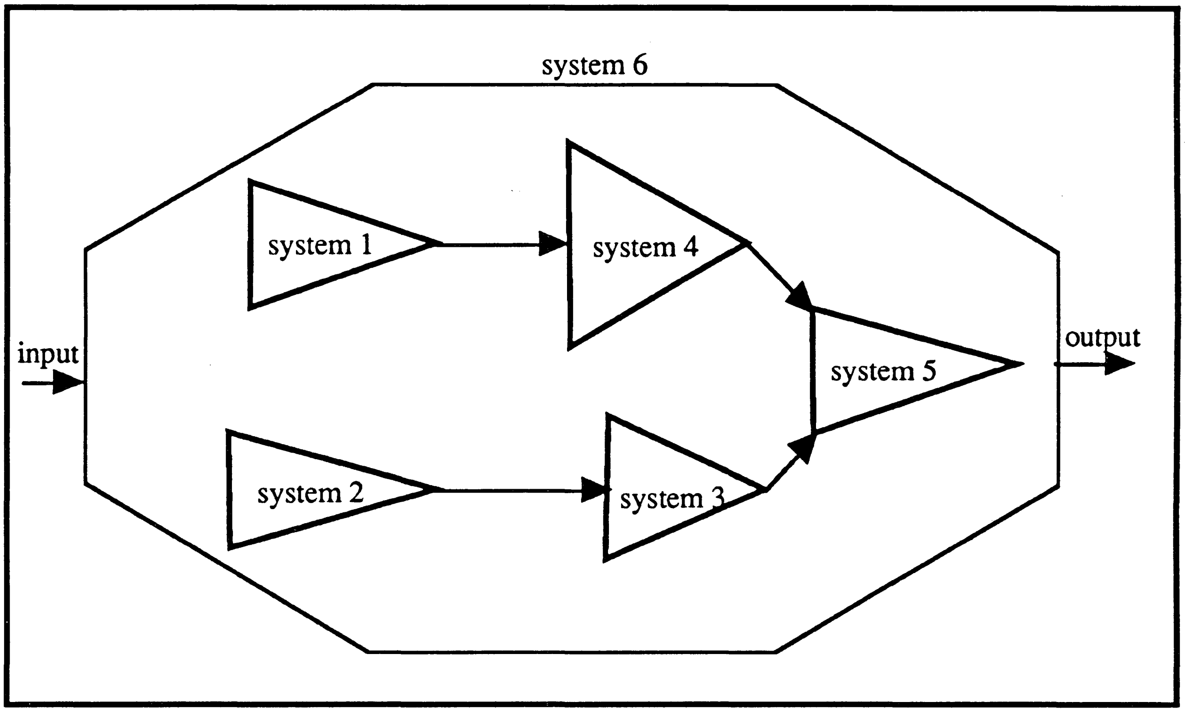


Figure 2-6: A system is comprised of other systems.

The problem of defining a system is clearly a problem of drawing boundaries about what is to be distinguished as system (Fig. 2-6), because nearly everything is a system. A gearing system is part of a car; a car with a man driving it is a system; a number of cars moving along a road make up a system; a collection of roads and vehicles is a system. In each case the system is a collection of entities interacting with each other. Each entity may be merely a thing, or may itself be a system and so one goes down the scale until one gets to the place at which there is no more system, but only things. And these things interact with each other in a recognizable pattern to make a system. The word has been used in different disciplines to refer to processes, objects, or things in such disciplines. For example, an automobile engineer refers to the automobile as a system that transports passengers. The network that distributes electrical energy into homes is called a system by the electricity authorities. An electronic engineer refers to the radio receiver set as a system. The sun, together with the planets revolving round it is called the solar system by planetary scientists etc. [Arguilar 1973, McEntyre 1979, Bailey 1982, Wilson 1984].

In the mapping sciences, there are various items that are called systems. The different methods of spatial data capture and processing for the purpose of producing a database are systems. For example, the photogrammetric process of making a map is a system. Remote sensing techniques of collecting spatial information represent a system. The collection of hardware and software for storage, retrieval, and processing of geographic data is called a geographic information system (GIS). Yet, there is a range of techniques which also constitute specialized systems such as analogue-based, hybrid, fully-analytical, digital-image photogrammetric systems, etc. All these are systems which are composed of other systems.

The concepts of systems have developed from the attempt to meet the challenge of management of complex processes involving different objects, components, and possibly people working together; a problem created by a society which is producing more people, more materials, more things, and more information than ever before, and which requires the integration of processes for effective utilization of resources. Chestnut [1965] noted, that "our emphasis on developing high-energy sources, rapid transportation, new materials, fast-acting control, as well as globe-circling communications, has brought into conditions of interdependence materials, equipment, and people which previously existed on a more or less completely independent basis, if in fact they existed at all". Systems approach is therefore used to bring into compatible operation seemingly unrelated equipment and people in order to achieve intended goals.

In a similar manner, spatial information systems have evolved from a number of causes, which include the desire to integrate all the previously isolated processes of mapping to avoid duplication; to make use of modern computing facilities and other new technologies in order to automate the production process for better accuracy and cost effectiveness; and to meet the increased demand for map information. More importantly, the emergence of spatial information systems has been considerably influenced by the desire to be able to manage and use more efficiently the huge volume of spatial data being collected, and to employ the capability of the computer to assist in solving complex spatially related problems. Fortunately, the development of digital information processing which has immensely facilitated the use of map data in digital form, has provided the impetus for the systems approach in spatial information processing. Thus, data collection and processing, which in the past were done in isolation are now being integrated into systems.

In systems concepts, a spatial information system, at the highest hierachical level, may be represented as a two-component system with a data collection subsystem and a data processing/management subsystem. However, going down the system's hierachy, it disintegrates into a number of complex subsystems, each of which has its own unique characteristics. Nevertheless, a systematic analysis of the components of the system will help to simplify its structure and aid in understanding its functions. This is the task of the next section.

2.4 General Structure of a Spatial Information System

Generally, a spatial information system will include a data input or data collection unit (hardware and software), a data processing/management unit (hardware and software), human and the database components (see Fig. 2-7) [Dale and McLaughlin 1989].

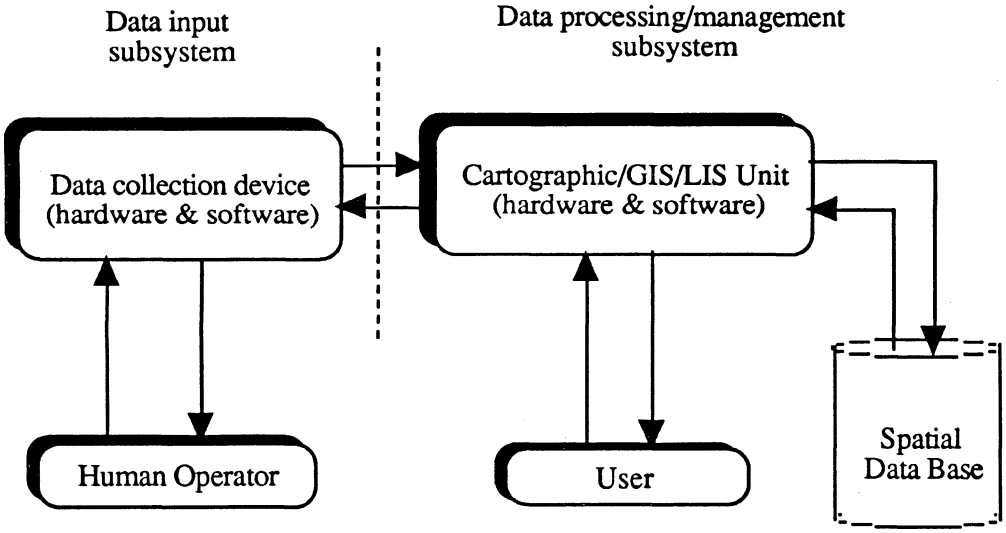


Figure 2-7: A spatial information system is comprised of data input and data processing units, the human operator and the database.

The human component is involved since the system is of the man-machine type in which both man and computer-based machines interact to perform the system functions. For example, in a photogrammetry-based spatial information system, the human is involved in all the orientation operations, feature digitizing and editing operations, and in the applications.

The hardware component includes the computer and all the necessary peripheral devices such as printers, plotters, graphics terminals, etc. Again, for a photogrammetry-based information system, these will also include for example, the stereo plotter (analogue or analytical) with all its supporting equipment. Equally important is the software component which includes the computer programs to perform specified tasks, such as orientation programs, aerotriangulation program, feature digitizing and DTM collection packages, and also the digital cartographic software package to edit and to store the collected data in the data base. The data base component is basically the collection of files containing the spatial data records, properly organized in a standard data structure.

On a more abstract note however, a system, in addition to components, also has attributes, and relationships [Arguilar 1973, Coutinho 1977, Blanchard & Fabrycky 1981]. While components are the physically operating parts of a system, attributes and relationships dictate the rules of such operation. Attributes are the properties or discernible manifestations of the components of a system. These attributes characterize the functional parameters of the overall system. For example, in a photogrammetry-based system, the time response of the system is characterized by the time responses of its component parts. On the other hand, relationships are the links between component attributes and system attributes, i.e., they define the functional link between the component attributes and total system attributes [Blanchard and Fabrycky 1981]. For system optimization, it is necessary that the attributes of the system be describable, even if only approximately, by a set of parameters. Thus, in addition to its components, a system generally requires the concept of state for complete description. This concept is explained in detail in chapter 4 where an empirical functional representation of the state of a photogrammetric production system is derived in the context of system optimization. However, as already mentioned, defining a system involves defining what is included within the boundaries of the system. Having identified the four main components of a generalized system (data input, data processing/management, human, database), it is helpful to examine in greater detail and to identify more specifically, the components of both the data input and the data processing/management units of this system. In this way, a set of possible configurations for a specialized spatial information system will be identified. Figure 2-8 shows the diagramatic representation of the components of the data input and processing units of a generalized spatial information system. These components are briefly described.

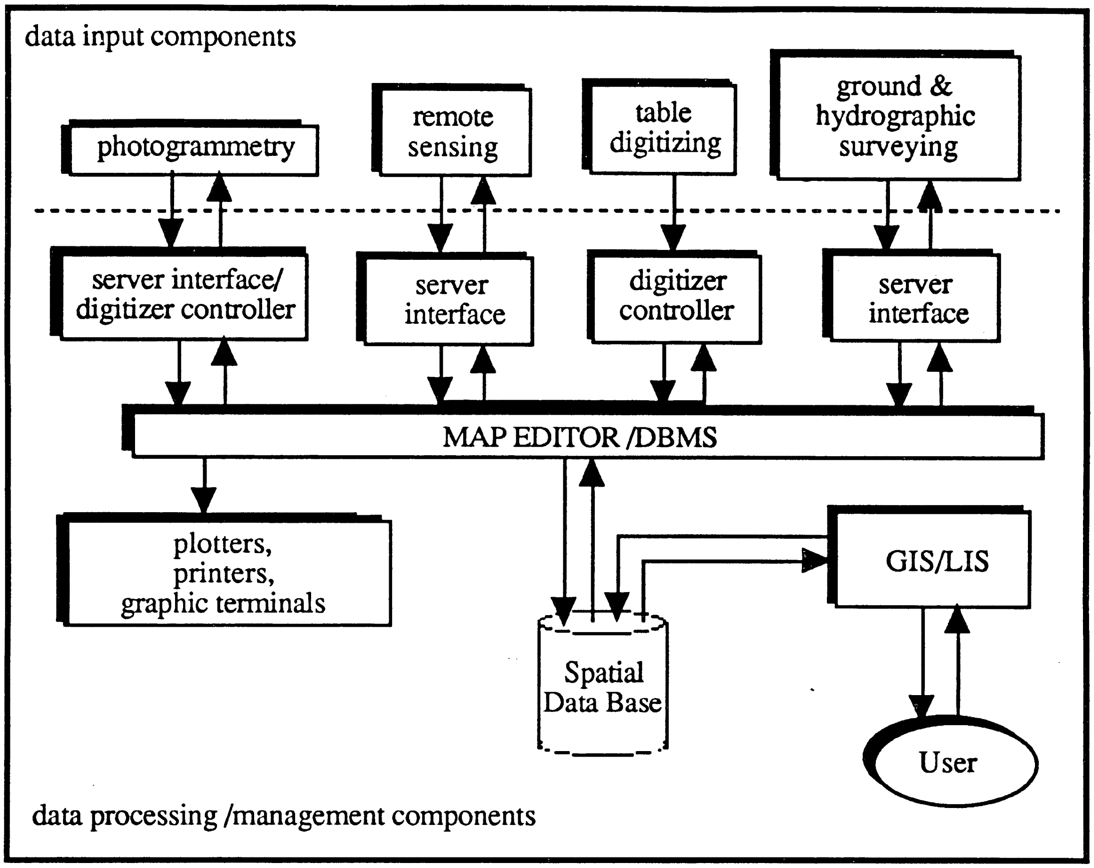


Figure 2-8: Input and output components of a general spatial information system.

2.4.1 Input Units

The first set of components consists of the data input units, each of which is a system designed to accept or collect raw data, pre-process such data, and supply digital or textual information in a suitable format to the processing/management unit of the system of which it is a part. Included in this group are:

1 Photogrammetric mapping system which is used for topographic mapping of large areas, and nowadays also for surveying and monitoring of industrial processes and facilities [Burnside 1979, Cooper and Robson 1990, Stirling 1990, Petrie 1990c]. The raw input data is captured with an optical or digital camera mounted in an aircraft or on

ground-based platforms. It is then pre-processed by the system, according to some specified rules and procedures, to produce digital data about the objects in the area concerned. The rules and procedures used are usually determined based on some criteria. The data input unit employed in this study belongs to this category, and therefore its components and their various configurations are discussed in detail in section 2.4

2. Remote sensing system which uses a varied range of technologies both for spectral response data acquisition and for processing. Although, this method is not new, the recent developments in digital techniques have enhanced the procedure significantly. In particular, the development of artificial satellites which carry radiation sensing devices has made possible the collection of enormous quantities of digital data about the surface of the earth. And since this data is in computer-compatible form, its integration into the modern spatial information systems is greatly facilitated [Swain and Davis 1978, Sabins 1987, Szekielda 1988]. Moreover, the development of digital image processing techniques has made the extraction of spatial information from such data by the computer a reality, thus making the method a rather useful tool for monitoring the earth's resources. Because of the vantage orbital position, the multi-channel design, operational flexibility, and improved resolution of some types of remote sensor systems, this technique is proving to be a cost-effective method for collecting a variety of information over large areas of the earth. It is being used to investigate regional engineering phenomena such as surface drainage, ocean water pollution, ice cover, etc.; and more localized phenomena, such as landslides, volcanic erruptions, building heat loss and so on [Petrie 1990c]. For map production, it is being used, although to a limited extent, for small scale topographic mapping. In addition, it is used widely and successfully to produce thematic maps for engineering projects, since for such applications users are satisfied with a relatively lower level of positional accuracy and completeness in comparison with the requirements for topographic mapping.

Nonetheless, this system constitutes an important data capture technique, and with the rapidly expanding digital technology, its potential as an economical source of spatial data appears to be unlimited. However, further details of this system are outside the scope of this study.

3. Digitizing system which is an analogue-to-digital conversion system used to retrieve digital data from existing paper maps (analog databases). Data input to this system is the paper map, which is mounted on an electronically gridded table and digitized manually by the operator. More recent methods scan the map and extract the information by digital data processing [Sprinsky 1987]. This system is so far the most common type of data capturing unit used with spatial information systems. Apparently, because most users of spatial information systems are at the transition stage, great attention and investment are devoted to table digitizers for the conversion of the existing stock of paper maps into digital databases [Masry 1972, Case 1981, Sprinsky 1987]. However, it is expected that new mapping projects will be executed with modern automated analytical techniques which are capable of providing digital data that are more accurate, and which offer more options and flexibility than table digitizing. These advantages have been stressed by Masry [1972] and Case [1981]. Nevertheless, this method is expected to continue to be popular for data capture until exiting stocks of maps are digitized.
4. Ground survey system which provides direct geometric observations to determine the spatial locations of physical features. Unlike remote sensing systems, this system requires contact with the spatial entities being measured, and since the measurements are usually geometric, the spatial entities are marked with points, lines and polygons before the necessary linear and angular measurements are taken. Traditionally, this approach uses methods such as theodolite traversing, tacheometric surveying, and plane tabling. More recent methods make use of the modern technology to provide more flexible and more accurate determination of locations [Richardus 1984, Vanicek &

Krakiwsky 1986, Kahmen & Faig 1988]. Among these are the NAVSTAR-GPS positioning, and electronic tacheometry (total station). With the modern communication systems, it is possible to integrate ground survey systems into an information system so that the measurements are processed in real time (automatic data flow). However, further details of these systems are not within the scope of this study.

5. Hydrographic survey systems which are designed to collect locational information about water bodies. These systems are of prime importance because nearly three-quarters of the earth's surface is covered by water [Ross 1978, Armstrong & Ryner 1981]; and it is believed that a huge amount of the earth's resources is buried under it. Thus, it must be properly mapped so that such resources can be tapped and used. Moreover, man has learned to use the oceans as a means of transportation between continents, hence proper charting of the sea bed is needed to guide navigation [ibid]. Technically, the methods of water mapping are somewhat similar to those for ground surveys, but because no visible beacons are available (and in fact, none can be placed) at sea, other specialized methods and instruments are used to supplement or replace ground survey methods. Furthermore, the analysis and use of such data require more specialized approaches, which are handled by specially trained personnel. Typical systems for hydrographic positioning include electronic positioning systems such as Loran C, Omega, GPS, NNSS, etc. for offshore positioning, and less specialised systems such as the sextant-echo sounder combination for nearshore positioning [Budlong 1978, Thomson et al 1979, Tetley and Calcutt 1986, Sonnenberg 1988].

2.4.2 Data Processing/Management Unit

This is the set of components used for an orderly arrangement of spatial data records into the database and subsequent application of such data in the management of the environment (Fig. 2-9). The central hardware component is the computer, which is supported by a host of peripheral devices for command and control, visual display, plotting and printing, editing, storage and retrieval facilities. Of course, the software controls the activities of the hardware; so the main software components include cartographic processing unit for editing of map elements, GIS/LIS application modules and a database management module (DBMS) [Monmonier 1982, Chrisman 1988].

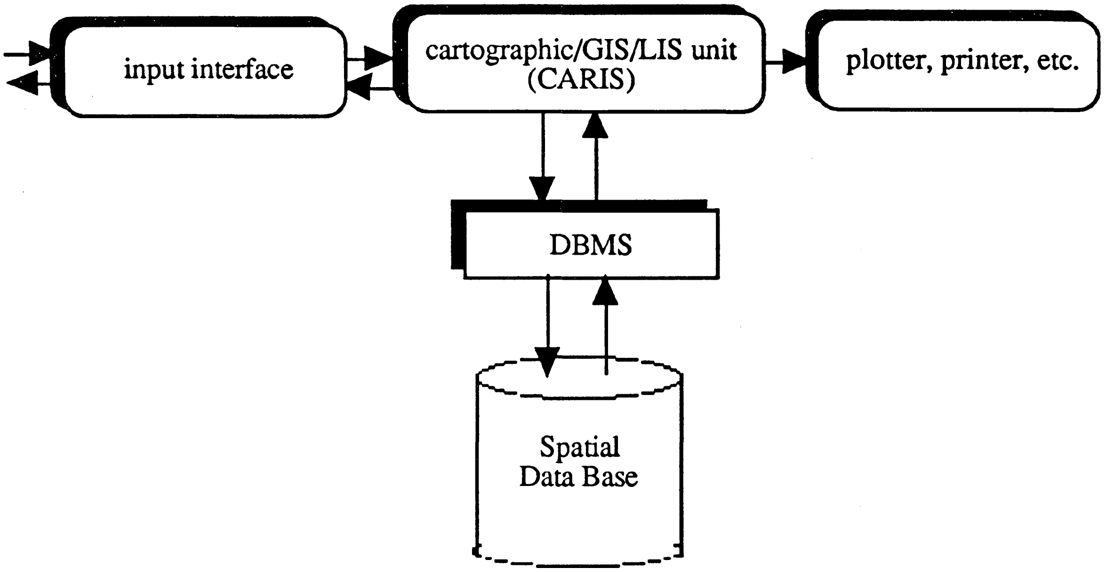


Figure 2-9: Data processing/management unit for a spatial information system.

It also contains an interface module through which it communicates with the input unit for data input. At this juncture, it is helpful to point out the difference between a cartographic processing system and a geographic information system. A cartographic system is basically the collection of hardware and software for editing, symbolization, storage, and retrieval of map data, i.e. it creates a digital spatial database using the data supplied by a collection unit. A geographic information system, on the other hand is a tool (hardware and software) designed to use and to manage a spatial database to provide solutions to spatial problems, for example, the planning of land use or the management of forest resources. Thus, a GIS system is concerned more with the mathematics and analyses of spatial forms than with the storage and editing of data, although, some cartographic capabilities are usually needed and indeed are incorporated in a GIS to store and display derived results. Yet, despite this distinction, some cartographic systems have dual capabilities, i.e. they serve as both a cartographic and a GIS processing system. An example of such system is the CARIS GIS, which accepts data from a number of input systems for editing and storage, and also is able to perform spatial analysis and modelling [CARIS documentation 1990]. Thus, the CARIS GIS is both a cartographic system and a GIS. Nonetheless, since this research is concerned primarily with data collection and storage, the discussions of applications of information systems are beyond the scope of this study.

In systems concept, many types of photogrammetric data reduction systems exist, each of which is an option for acquiring data into a GIS. These options are briefly described in the next section.

2.5 Options for a Photogrammetry-based Data Capture System

Figure 2-10 shows the various arrangements of a photogrammetry-based spatial data capture system. Each configuration basically consists of a data input unit with preprocessing capability, and an output unit represented as a cartographic module in this case.

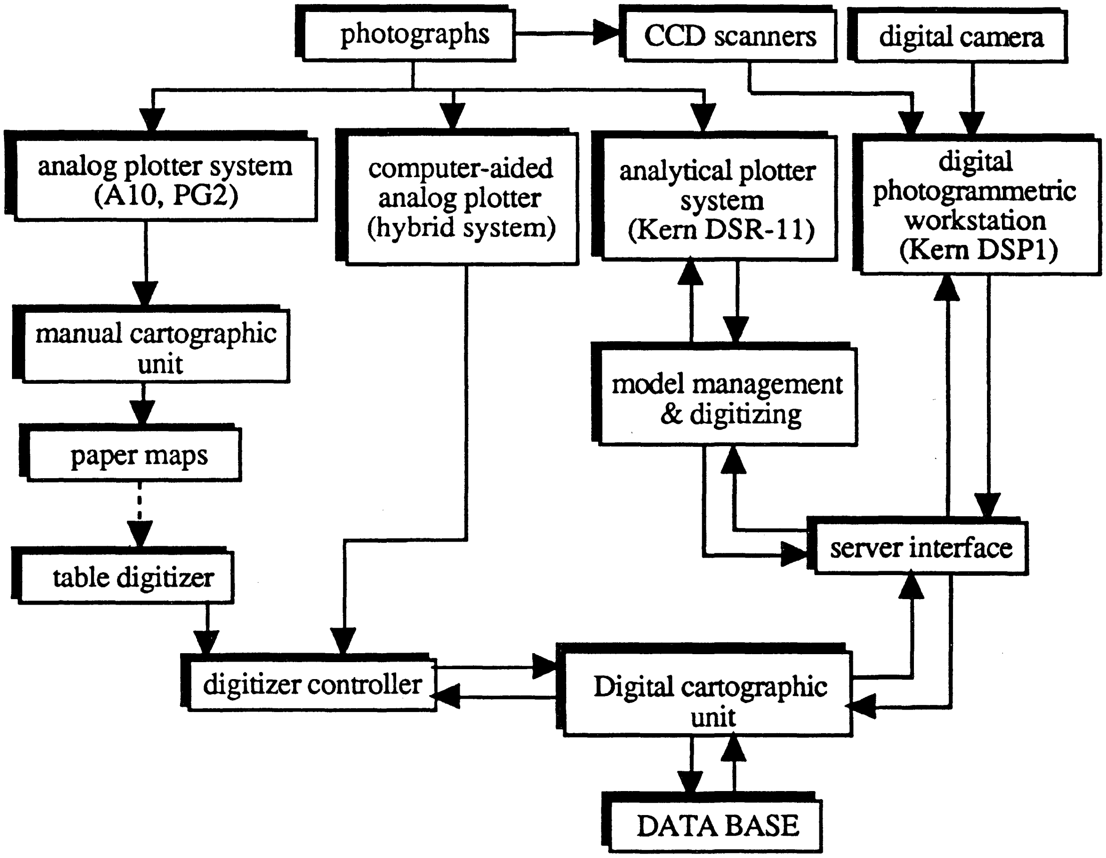


Figure 2-10: Configurations of a photogrammetry-based data collection systen

The input unit is the distinguishing component of each configuration. In terms of modern computer terminology, each of these arrangements can be characterized as either an analog or computer-assisted arrangement. The analog configuration is simply the traditional process of mapping using analogue stereo plotters. This approach involves largely labour intensive operations in which the operator does everything manually, from orientation to measurement and manuscript preparation. He moves the measuring marks, sets all the knobs, takes decisions and does the measurement and recording. This method has been used widely in the past to produce maps of varying scale and detail, the bulk of which constitutes the so called analog spatial databases that are now being converted to digital form. Nevertheless, expanding growth of digital technology has made this approach obsolete, although it is still being used in the developing countries.

The computer-assisted methods on the other hand make use of modern digital technology to relieve the operator of some of the chores of the mapping process. The degree to which the operator is assisted determines the level of automation of the configuration. In general, modern mapping procedures are based on the computer-assisted systems; and these are the systems that are of significance to this study. Therefore, further reference to photogrammetric systems will imply the computer assisted type, the different types of which are described in the next section.

2.6 Types of Computer-Assisted Photogrammetric Units

A computer-assisted photogrammetric unit could be of many different configurations [Dorrer 1976, Zarzycki 1978, Jaksic 1983, Makarovic 1988, Petrie 1990a,c]. However, only three configurations are important in terms of the modern concept of computer-assisted mapping. These configurations include the hybrid system, the analytical system, and the digital image system.

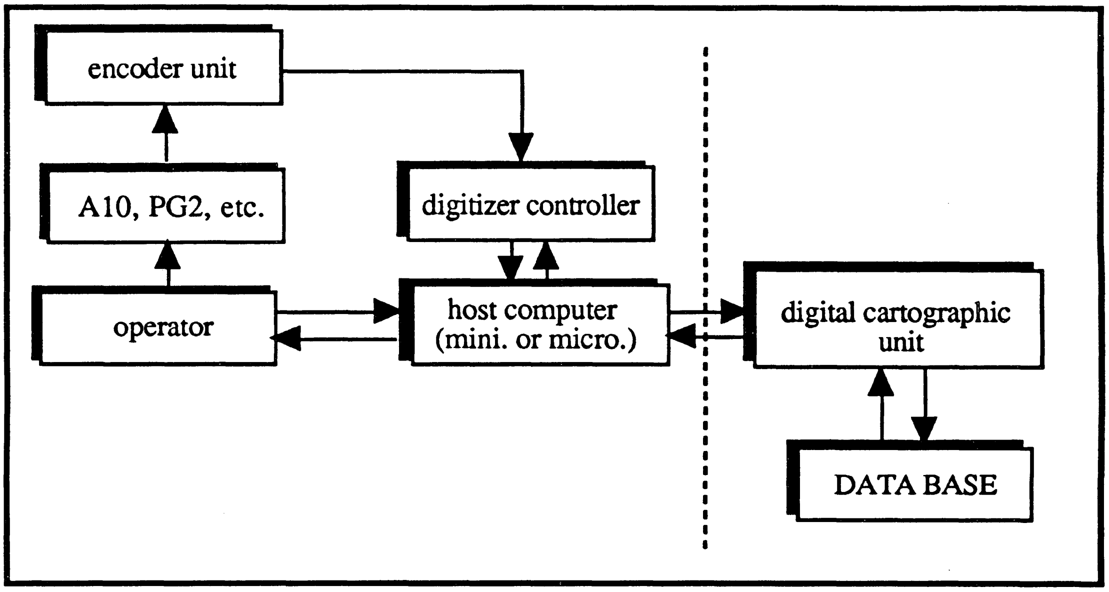


Figure 2-11: Configuration of a hybrid photogrammetric system.

A hybrid photogrammetric system is one which is formed using a computer-aided analogue instrument. The components of such a system (see Fig. 2-11) include an analogue stereo plotter such as the Wild A10 or the Kern PG2, a human operator who performs the orientation and setting-up operations, a digital readout unit which accepts coordinates from an encoder and feeds it into the host computer. The host computer performs all the necessary support computations and also transfers data to the cartographic unit. This kind of arrangement is often called an 'open-loop' system because there is no feedback from the host computer to the analog instrument (Dowman 1977, Petrie 1990a). Because the analog stereo plotter is not computer-controllable, the operator's participation in the system is high and the level of automation is therefore very low.

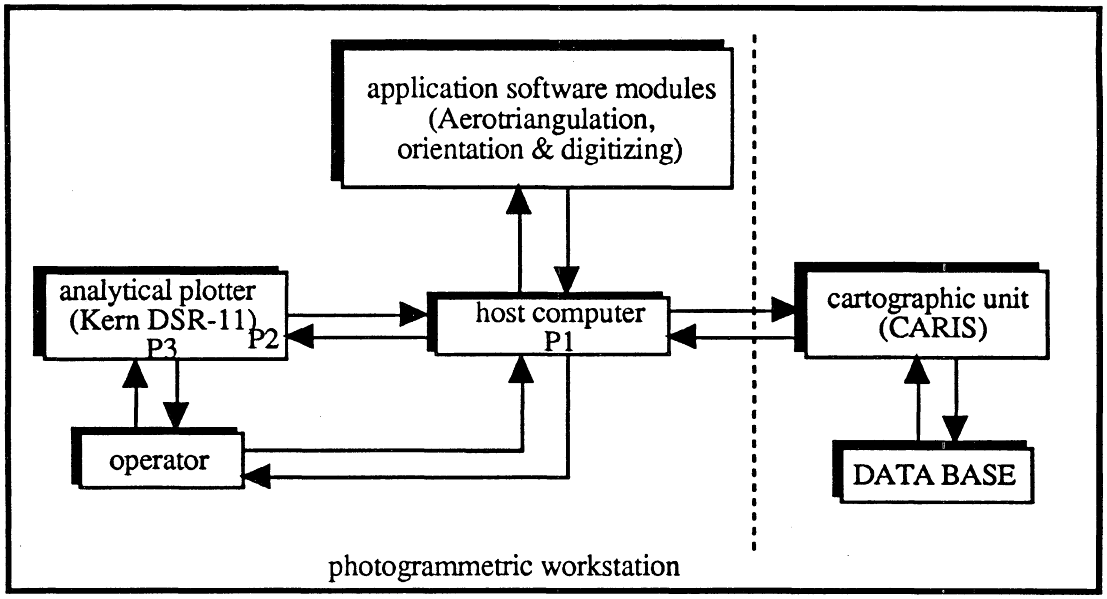


Figure 2-12: Configuration of an analytical photogrammetry-based system.

An analytical photogrammetric unit is one which is assembled around an analytical stereo plotter (Fig. 2-12). Its components include the analytical stereo plotter with its electronic operator (processor P2), the host computer (processor P1), a human operator, and a server interface for communication with the cartographic unit. The analytical stereo plotter communicates with the human operator through a processor (P3), and with the host computer through its electronic operator processor P2. This type of unit is often called a 'closed loop' system since there is a feedback from the host computer to the plotter [ibid]. In this system, the analytical plotter is computer-controllable, and so the human operator's participation is lower than in the hybrid system. Thus, it has a higher automation content, which will increase with developments in digital technology. Systems of this type have a great potential for automated data collection.

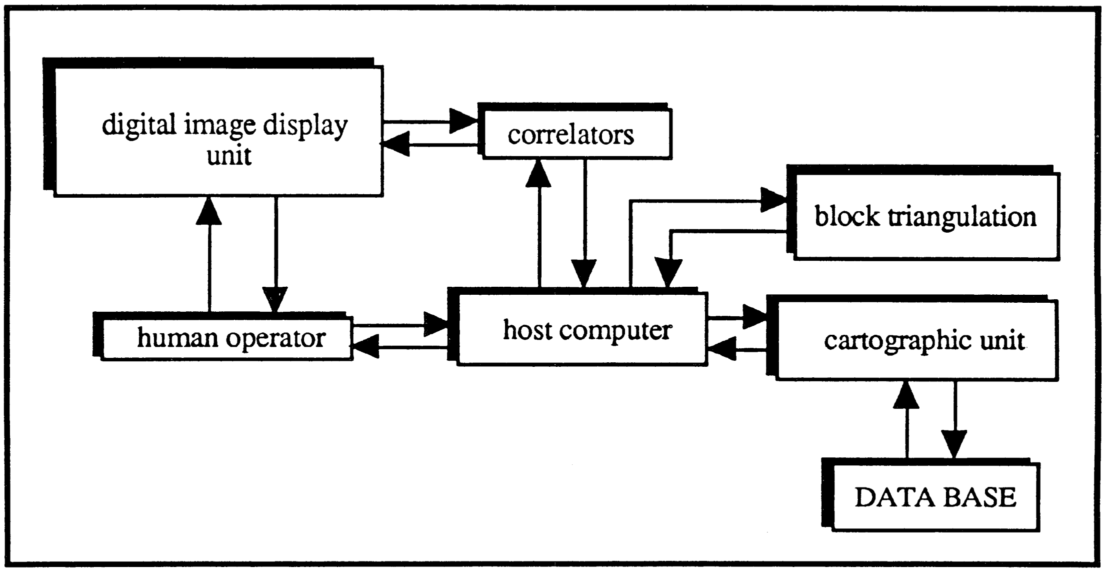


Figure 2-13: Configuration of a digital image photogrammetry-based system.

A digital image photogrammetric unit (Fig. 2-13) is one that uses an image analysis system to process the raw digital photographic image either automatically by image correlators or semi-automatically by operator assisted image correlation. The input data to such a system consist of a digital image, obtained by scanning the original analogue image or by using a digital camera to capture the scene (Helava 1988, Makarovic 1988). The system also comprises a host computer (processor P1), and a server interface through which communication with the cartographic unit is achieved. Also included is a processor P2 (or in some systems a set of transputers or an array of parallel processors) for fast correlation and data extraction purposes [Dowman 1990]. Because of the high computer component and the possibility of operating without operator's assistance (ie. using artificial intelligence), systems of this kind have the highest potential for total automation of the spatial data collection process.

Nevertheless, for the development of photogrammetric mapping input units today, the hybrid configurations are mostly used for reasons of "economy" and to preserve investments in the old analog technology. However, with the rapidly decreasing cost of analytical plotters (Petrie 1990a), and the increasing power and sophistication of modern computing systems, analytical plotter-based systems will become the only viable tool for building photogrammetry-based spatial data capture systems. This is more so because digital image systems, even though they have higher automation content, are still very expensive, and because of the inability of image correlators to recognize features, coupled with the low resolution of most scanners and digital cameras [Helava 1988], their use for building commercially viable photogrammetric mapping systems is at present, not feasible. Consequently in this thesis, further reference to data input units is limited to the analytical plotter-based type. And because the architectures of different analytical plotters are not all the same, unless otherwise noted, further reference to the analytical plotter is to the Kern line of plotters, particularly, the DSR-11 used in this study. In the next section, the details of the composition of the analytical plotter-based system, are presented.

2.7 An analytical Plotter-Based Mapping System

Figure 2-14 shows the components (hardware and software) of an analytical plotter-based data capture system. It comprises an analytical plotter which is simply a two-stage comparator operated by an electronic operator (commonly called processor P2) and assisted by a human operator.

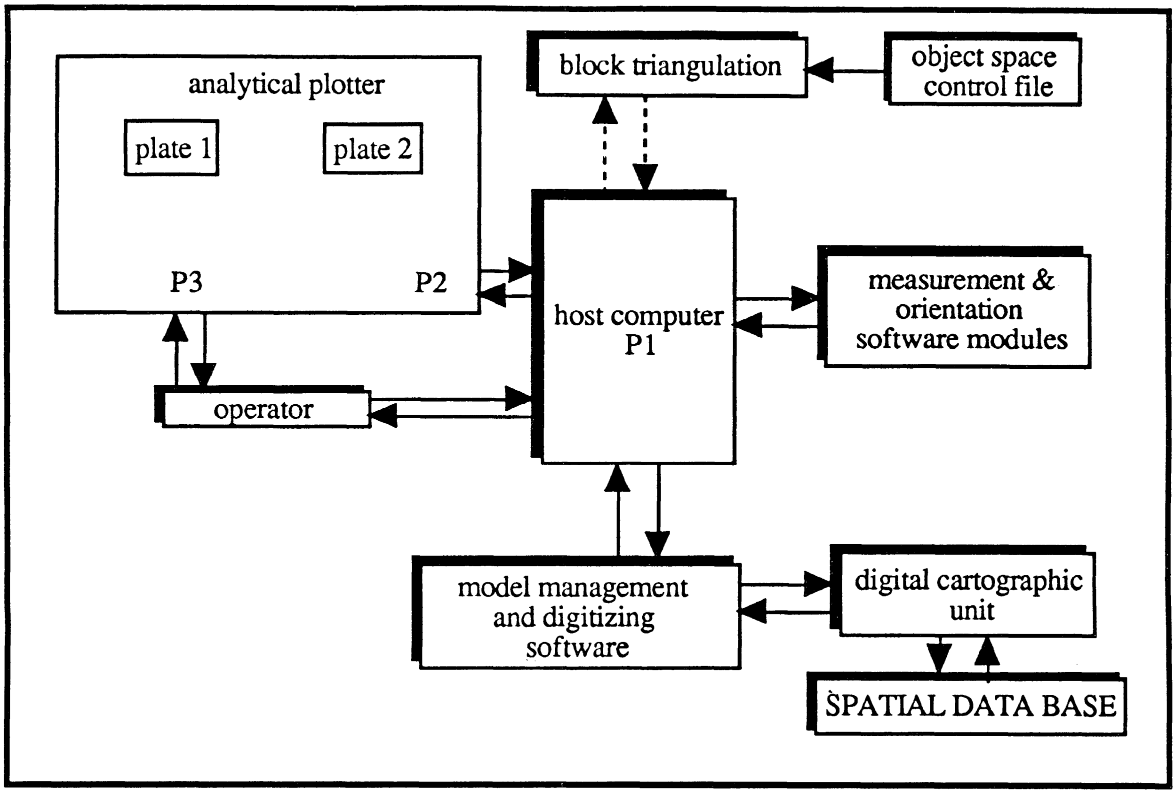


Figure 2-14: Components of an analytical plotter-based system.

The plotter system communicates with the human operator through a processor (P3); and both the human operator and the electronic operator (processor P2) communicate directly with the host computer (processor P1). The electronic operator performs most of the driving operations which the human operator manually performs on analog instruments; while the human operator performs functions which require the use of intelligence. The host computer (processor P1) executes all the needed operational computations and also manages the transfer of data to and from the plotter system. The host computer used in this study is the microVax-II minicomputer.

The software components include: (1) block triangulation program, a major tool in the workstation, which simultaneously transforms blocks of models or photographs from the model space or the image space respectively to the object space system. It supplies the camera's exterior orientation parameters and control point coordinates to the workstation for subsequent operations. (2) interior orientation, photo-coordinate measurement and exterior orientation programs. The interior orientation module establishes the image space reference system (with its distortion characteristics) on the instrument so that correct measurements can be taken. This operation is the most frequently required of all the orientations involved in the restitution of stereo pairs of photographs. The photo-coordinate measurement module is used to measure image or model coordinates of points for triangulation and for certain types of model restoration techniques as we shall see later in chapter 4. The exterior orientation programs determine the setting up parameters for each stereopair of photographs either directly by resection (2-photo resection), or sequentially (relative orientation followed by absolute orientation). (3) the digitizing package which allows the collection of vector data about entities appearing in a restored stereo model. As implemented in this work, the digitizing module communicates with the analytical plotter, and through a server interface also with the cartographic unit (see Figure 4-14). It also has the duty of resetting the models for digitizing. This is done to ensure maximum integration between model reorientation and digitizing. However, the functioning of the application software packages is facilitated by the operating system module. The operating system represents the nerves of the system which allows the hardware component to respond to the commands of the application software components. The Kern Company, which manufactured the DSR-11 has developed a number of hardware-addressing subroutines which are supplied as basic system software, and also some data reduction packages for orientation and digitizing as optional extras [Cogan & Polasek 1988, Hunter & Smart 1988]. This operating system software enables the development and use of application programs for various mapping purposes.

Indisputably, it makes good economic sense to organize the workstation in such a way as to achieve a cost effective operation. This is a desirable quality and one which creates a necessity for the assessment and optimization of the station's configuration (hardware and software). However, in systems optimization, the software components are often the more easily and more economically changeable components of the system. And because they can be manipulated to suit a desired operational process, they consequently determine the performance and economy of the system. For example, in some stations, as noted in the introduction, the block triangulation program is often run off-line on a mainframe computer with the results downloaded to the workstation. While such an arrangement gets the job done, it may not be economical in terms of cost and human labour involved. The improved power of modern computing devices has made available other options. All the processes involved in data reduction can now be performed within the same environment (at least for moderately sized blocks); thus, enabling the integration of all mapping processes into a user-friendly spatial data collection system which may be assessed as a unit. Moreover, as will be discussed in chapter 4, depending on the availability of necessary software packages, there are many types of block triangulation schemes available for use, and also many ways in which the results may be utilized in subsequent processes to achieve the interior and exterior orientations needed for an analytical restitution of each stereogram. Therefore, an evaluation of all of the possible software design options is essential for optimizing the configuration of the photogrammetric workstation.

Another important quality of the workstation is the possibility for real time communication with a GIS. Apart from the economy and operational flexibility of such an an arrangement, the need for an immediate editing of the collected data to ensure freedom from errors, and to tag the graphic elements with attribute data at the point of acquisition is compelling. In practice, as will be addressed in chapter 5, the issues of data incompatibility between the data source and the GIS environment, isolated (or solitary) operation of the photogrammetric software modules and limitations of the computing facilities often present problems which make such integration difficult or impossible. The technique used to circumvent these problems is described in chapter 5.

In the next chapter, an analysis of the various mathematical modules used at all the stages of the photogrammetric process, and the computational algorithms employed in the software development are presented.

Chapter 3

Mathematical Foundation of Analytical Photogrammetric Systems: The Vector Space Approach

An analytical photogrammetric system is a collection of tools and procedures for transforming photographic measurements into corresponding object space locations and vice versa. Using the principles of analytical geometry, the projective relationship between the image space and the object space is mathematically constructed and used as a numerical bridge between the spaces. Apart from the analytical plotter hardware and the host computer, most of the components of an analytical plotter-based system are software modules carrying out specific functions assisted by an operator. From a mathematical point of view, almost all the processes involve a transformation from one coordinate system to another, an operation for which both the elements of interior and exterior orientations of the image space are needed. While some of the modules derive these transformation parameters from the input data (usually not in real time), others use them in real time mode to deliver the needed spatial information. The software modules which perform these tasks are developed based on appropriate mathematical formulations.

The literature suggests that it was probably Schut from the National Research Council of Canada who first employed vector calculus in the treatment of photogrammetric problems [Schut 1964, 1966, 1973]. He applied this approach to explain the relative orientation problem using the coplanarity condition. More recently, Konecny and Lehman [1984] applied the vector technique in the German edition of their book on photogrammetry; and Ghosh (1988) employed the vector approach to simplify the presentation of some basic concepts of analytical photogrammetric computations. However, these excellent texts do not emphasize the practical application of the vector method in that the final formulations were often reduced to the usual long hand form. In general, authors' prefer the long hand approach in which symbols are used to represent single variables in a tedious algebraic manipulation [Wong 1980, Wolf 1983, Methley

1986]. During the course of this research, the author has come to realize that the traditional renditions of analytical photogrammetric formulations lead to an incredibly complicated computational process. Simply stated, the presentation and use of these formulations without vector symbology is comparable to the use of least squares without employing matrices. In this study, an attempt is made to correct this situation by recasting existing formulations purely in vector form and adopting new and simplified approaches to the derivation of working algorithms for their applications. While the basic concepts of such functional models are not new, the vector-based representation adopted here stimulates the understanding of the basic technologies and facilitates the implementation of efficient programs for their applications.

This chapter discusses aspects of mathematical concepts, including functional analysis and linear space algebra, which underlie the development and functioning of the software modules performing the basic photogrammetric tasks in an analytical system.

3.1 The Concepts of Mathematical and Stochastic Models

In a simple sense, mathematical modelling is a process of using mathematics to represent physical realities, objects or events. From historic experience, we have come to realize that observation is the basis of our understanding of the world we live in. However, observation only provides information about the specific events which we observe; alone, it provides little help for dealing with new situations. Useful knowledge results from our ability to identify common properties of different events, isolate the important characteristics, and generalize our experience. Generalization enables us to operate effectively in new situations by using inferences drawn from past experience. This generalization leads to a statement of some principle, often in the form of equations which express the relationships between physical properties involved in the phenomenon being observed. These equations are what over the years have been referred to as mathematical models. Therefore, a mathematical model may be described as an abstraction which associates parameters and processes of the real world with expressions and operations in a mathematical structure. More succinctly, it is a mathematical characterization of a phenomenon or process.

The practice of exposing a film in a camera to record the image of an object is a phenomenon which from past experience and study has lead to the principle that the image point in the camera, the optical centre of the camera lens, and the corresponding object point lie on a straight line. This principle is a generalization of the previous experience of observing the phenomenon of the pin-hole camera. The principle which in mathematical language is referred to as the collinearity condition is used to derive computational formulas for the reduction of photographic images; and it is this formula that is called the mathematical model in this instance.

From an application standpoint, the purposes of mathematical models are twofold. In the first case, mathematical models are used to provide a convenient method for performing certain computations, e.g. the computation of ground coordinates by intersection of rays (or bundles). Moreover, frequently there are more than one way of formulating the mathematical model for an observed phenomenon or at least different levels of refinement of such a model. Nevertheless, the choice of a model for the purpose of computation is often directly influenced by available computing facilities. For example, the collinearity equation may be formulated rigorously by extending it to include additional parameters or it may be used in an approximate form by making an assumption that the photographic image is vertical and free from systematic distortions. Secondly, mathematical models are useful for investigation and prediction, for example, simulation of real world phenomena with physical models. Taking the collinearity equation again as a case study, it could be formulated in the object space for the purpose of simulating photographic measurements. In summary, it may be said that a mathematical model is a functional representation of a physical phenomenon which enables us to study in a scientific and convenient manner, problems of related nature in our effort to better understand the world we live in.

Yet, mathematical models, though desirable and rather indispensable, nonetheless ignore certain aspects of the real world. For example a functional model does not take care of the stochastic properties of the variables involved in the mathematical model. In real world processes, repeated observations of a phenomenon or some physical parameters, made in quick succession give rise to different measured values. These inconsistencies cannot be removed by refining the functional model or by applying corrections but are often said to be caused by random errors, whose size and sign are subject to random events [Bevington 1969, Meyer 1975, Fuller 1987]. It is customary therefore to regard measurements as random variables and to describe the random effects by means of a stochastic model. A stochastic model is thus a statement of the statistical behaviour of the variables involved in the mathematical model of a given phenomenon. Its purpose is to introduce the dimension of probability into the model in order to be able to specify the degree of certainty of predicted values from the model. In adjustment methodology, this stochastic model is often presented in the form of covariance estimates [Mikhail 1976, Vanicek and Krakiwsky 1986]. In the use of a mathematical model to solve a problem, both the values of the observable quantities and their covariances make up the input data for the computation process. The output of the estimation process includes the predicted (computed) values of the unknown variables together with their covariance estimates. These covariance estimates may then be used to judge the confidence level or reliability of the parameter estimates for use in other applications [Mikhail 1976, Kavouras 1982, Cooper 1987]

Having provided a basis for the concept of mathematical modelling of observed processes, it is helpful to know the methods of characterizing the real-world spaces (or domains) in which such physical phenomena are observed and modelled. Of particular importance in this regard is the nature of the elements of such spaces and the type of mathematical operations that may be performed on them both to formulate the mathematical model and to evolve computational schemes for their manipulation. Therefore, the next section describes the key properties and algebra of the Hilbert vector space.

3.2. The Hilbert Vector Space Concepts

A vector space is a linear space in which entities exist which may be represented by a set of numbers referred to a coordinate system. The concept of a vector space has emerged as a special case of the abstract set theory through the definition of a set whose elements are composed of a collection of real numbers referred to a cartesian system of coordinates [Halmos 1958, Collatz 1966, Maddox 1988]. Moreover, the notion of vector space operations is related to functional analysis, a branch of mathematics, dealing with operations on abstract sets, i.e. the manipulations of the elements of a set according to some stated rules or axioms. A similar operation in a vector space is often called vector analysis [Eliezer 1963, Coulson 1970, Hoffman 1975, Giles 1987]. In the general functional analysis theory, it is usual to specify a space and the type of operations (functionals) allowed in the space. Thus, in practice, there are a variety of spaces defined, each with a unique combination of operators which characterize it. Of particular interest to us is the so called Hilbert space which combines the unique characteristics of a linear space and a normed space, thus, providing a basis for the formulation of the vector operators applicable to analytical photogrammetry. Conceptually, a definition of the Hilbert space must necessarily be consequent upon the definitions of the linear and the normed vector spaces since it possesses the properties of these spaces; therefore, the important characteristics of these spaces are summarized in what follows.

1. A **linear space** X is a non-empty set in which all linear operations such as addition and scalar multiplication on its elements are permitted [Luenberger 1969,

Maddox 1988]. For example: for all $\mathbf{x}, \mathbf{y} \in X$ and $s, s_1, s_2 \in R$, where R is the space of real numbers:

$$\mathbf{x} + \mathbf{y} = \mathbf{y} + \mathbf{x}$$

$$s(\mathbf{x} + \mathbf{y}) = s\mathbf{x} + s\mathbf{y} \tag{3-1}$$

$$f(s_1\mathbf{x} + s_2\mathbf{y}) = s_1 f(\mathbf{x}) + s_2 f(\mathbf{y})$$

f is any linear vector function.

2. A **normed space** is defined as a linear space X with an associated norm ($\|.\|$) defined. Usually the normed space is represented as (X, $\|\cdot\|$) and understood to consist of a linear space X and a norm $\|.\|$: X-> R such that

   for all $\mathbf{x},\mathbf{y} \in X$ and $\lambda \in R$:

   $\|\cdot\| = 0$ only for a null vector

   $$\|\mathbf{x}\| \geq 0$$

   $$\|\lambda\mathbf{x}\| = |\lambda|\,\|\mathbf{x}\|$$

   $$\|\mathbf{x}+\mathbf{y}\| \leq \|\mathbf{x}\| + \|\mathbf{y}\| \tag{3-2}$$

   This space is some times also called a **metric** space where the metric is said to be norm induced [Giles 1987, Maddox 1988].

3. The **Hilbert space** is a normed space whose norm is generated by an inner product [Luenberger 1969, Maddox 1988]. It is often represented as (X, $<\cdot,\cdot>$), where $<\cdot,\cdot>$ means inner product. This space is sometimes also called an **inner product** space. In most applications of the Hilbert space concept, the inner product is simply defined as the Euclidean distance between any two elements of the space. In general, an inner product on a real vector space (X) is a function that associates a real number $<\mathbf{a},\mathbf{b}>$ with each pair of vectors **a**, and **b** in X in such a way that the following axioms are satisfied for all vectors **a**,**b**, and **c** in X and all scalars k:

   $<\mathbf{a},\mathbf{b}> = <\mathbf{b},\mathbf{a}>$ (symmetry axiom)

$<\mathbf{a}+\mathbf{b},\mathbf{c}> = <\mathbf{a},\mathbf{c}> + <\mathbf{b},\mathbf{c}>$ (additivity axiom) (3-3)

$<k\mathbf{a},\mathbf{b}> = k<\mathbf{a},\mathbf{b}>$ (homogeneity)

$<\mathbf{a},\mathbf{a}> \geq 0$ and

$<\mathbf{a},\mathbf{a}> = 0$ if and only if $\mathbf{a} = 0$ (positivity axiom)

Based on these axioms therefore, the following linear operations and identities may be applied to the elements of a vector space possessing the Hilbert space properties [Coulson 1970]:

given $\mathbf{a}, \mathbf{b}, \mathbf{c} \in X$ and $m,t \in R$ where R is the real axis, then

$$\mathbf{a}\cdot\mathbf{b} = \mathbf{a}^T\mathbf{b} = <\mathbf{a},\mathbf{b}> = a_1b_1 + a_2b_2 + a_3b_3 \tag{3-4}$$

$$\mathbf{a}\cdot(\mathbf{b} + \mathbf{c}) = \mathbf{a}\cdot\mathbf{b} + \mathbf{a}\cdot\mathbf{c} \tag{3-5}$$

$$\frac{\partial(\mathbf{a}+\mathbf{b})}{\partial t} = \frac{\partial \mathbf{a}}{\partial t} + \frac{\partial \mathbf{b}}{\partial t} \tag{3-6}$$

$$\frac{\partial(m\mathbf{a})}{\partial t} = \frac{\partial m}{\partial t}\mathbf{a} + m\frac{\partial \mathbf{a}}{\partial t} \tag{3-7}$$

$$\frac{\partial(\mathbf{a}\cdot\mathbf{b})}{\partial t} = \mathbf{a}\cdot\frac{\partial \mathbf{b}}{\partial t} + \frac{\partial \mathbf{a}}{\partial t}\cdot\mathbf{b} \tag{3-8}$$

$$\frac{\partial(\mathbf{a}\mathrm{x}\mathbf{b})}{\partial t} = \mathbf{a}\mathrm{x}\frac{\partial \mathbf{b}}{\partial t} + \frac{\partial \mathbf{a}}{\partial t}\mathrm{x}\mathbf{b} \tag{3-9}$$

$$\frac{\partial(\mathbf{a}\cdot\mathbf{b}\mathrm{x}\mathbf{c})}{\partial t} = \mathbf{a}\cdot\mathbf{b}\mathrm{x}\frac{\partial \mathbf{c}}{\partial t} + \mathbf{a}\cdot\frac{\partial \mathbf{b}}{\partial t}\mathrm{x}\mathbf{c} + \frac{\partial \mathbf{a}}{\partial t}\cdot\mathbf{b}\,\mathrm{x}\,\mathbf{c} \tag{3-10}$$

Moreover, In order to take account of the stochastic nature of the variables of the mathematical model, recourse is often made to the concept of symmetric operator in an inner product space [Maddox 1988, p.183]. By this, the inner product definition may be adjusted to include the covariances of the observed (or estimated) elements of the space. The inner product is in this case defined as

$$<a,a> = a^TWa \tag{3.11}$$

where W is the weight matrix which in most cases is symmetric. This form of the inner product is the most often used in weighted least squares adjustment, also called Hilbert space optimization [Vanicek and Krakiwsky 1986].

3.3 Photogrammetric Vector Spaces and Coordinate Systems

Generally, two spaces are involved in the phenomenon of collecting terrain information with a camera. These include the image space which is defined by the camera cone and the filmbase, and the terrain or object space which is the environment of the object being mapped. Yet, in the process of reducing the captured images, a third space is often defined to simplify the reduction process. This space, called the model space, is intermediate between the image space and the object space. When suitable coordinate systems are defined for these spaces, they may be referred to as photogrammetric vector spaces, which may be characterized as 3-D Hilbert spaces (or simply Euclidean spaces). This permits us to use all the axioms of that space in the derivation of the mathematical models and computational algorithms to be discussed shortly. The main characteristics of these spaces are summarized by way of describing their associated coordinate systems.

3.3.1 The image vector space.

This is geometrically a three-dimensional system composed of the x-y plane of the photograph (negative or positive) and the direction of the optical axis of the camera lens (Fig. 3-1), which in an ideal case, is perpendicular to the image plane.

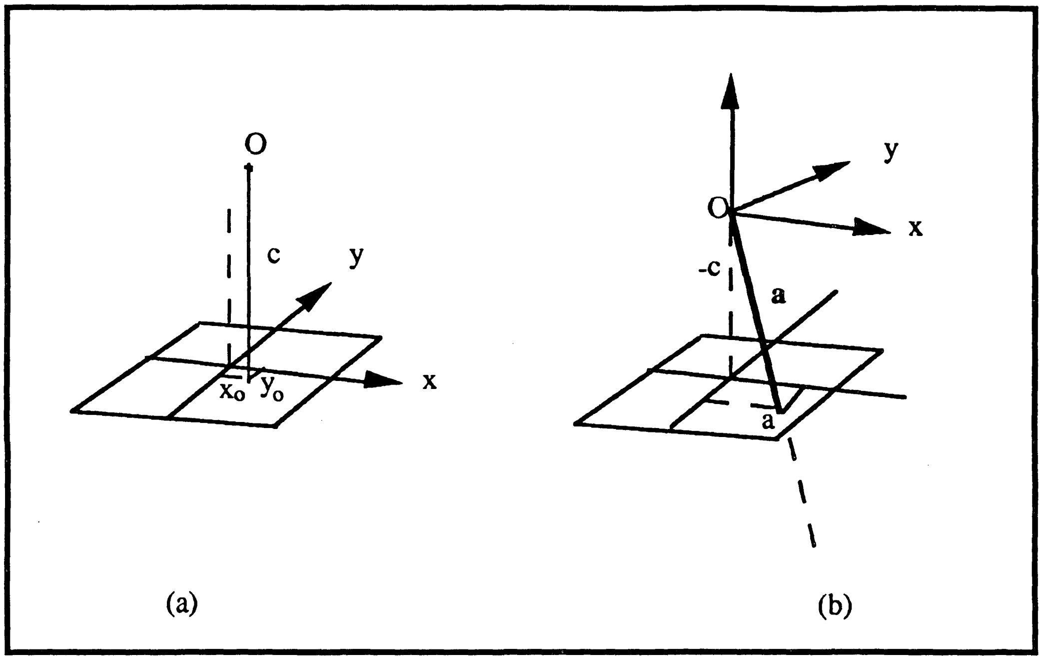


Figure 3-1: Image space system: (a) photographic axes (b) image space representation

The x-axis is assumed aligned along the general direction of the camera base or flight line. The y-axis and the z-axis are chosen so that the three axes define a three-dimensional right-handed Cartesian system with origin at the rear nodal point (perspective centre) whose orthogonal image on the photograph is at the principal point, and whose shortest distance to the focal plane is the camera constant denoted by c (Fig. 3-1b). In this system, any point such as 'a' on the photograph defines an image space vector **a**, whose components are given by (x_a, y_a,-c). Usually, the x,y coordinates are referenced to a system defined by a set of fiducial lines marked on the photograph, and whose intersection (fiducial centre) serves as the origin of the fiducial system. The coordinates of the principal point in the fiducial system are determined by calibration and may be denoted as x_o,y_o (Fig. 3-1a). Thus, a small shift equal to (x_o,y_o) is applied to each point so that it is referenced to the principal point. Consequently, a vector in the image space system is a position vector whose components are (x_a-x_o,y_a-y_o,-c).

However, it is necessary to note the importance of the elements of interior orientation which include the principal point coordinates (x_o,y_o), the camera constant (c), the coefficients of radial and decentering lens distortion functions (k_1, k_2, k_3, p_1, p_2) respectively. In addition to defining the geometry of the image space, these parameters also provide the correction for the systematic distortions within this space so that the correct location of each image point is achieved. As already noted, these elements are determined through camera calibration. Furthermore, the distortion of the film base is important , and it is determined by measuring distances between known marks on the image [Schut 1973, Ghosh 1988].

3.3.2 The model vector space.

This is a three-dimensional space created when two image spaces are brought together relatively such that corresponding vectors intersect at a point (see Fig. 3-2a). The origin and orientation of its coordinate system are usually taken to be those of the left hand image space and the scale is commonly chosen close to the photographic scale. Figure 3-2b shows a model coordinate system with origin at the left hand projection centre, the scale is determined by the base-length **b**. A position vector $\mathbf{p}_m$ in this space has three components (x_m,y_m,z_m), which may be computed from two image space vectors such as $\mathbf{T}_1$, $\mathbf{T}_2$ (Fig. 3-2a) when the elements of relative orientation have been determined. As the name suggests, this system represents a stereoscopic model in which terrain objects may be viewed in three-dimensions and are also easily transformed mathematically into object locations. On most analytical plotters, this is the space used for digitizing and collection of data for digital terrain models (DTM).

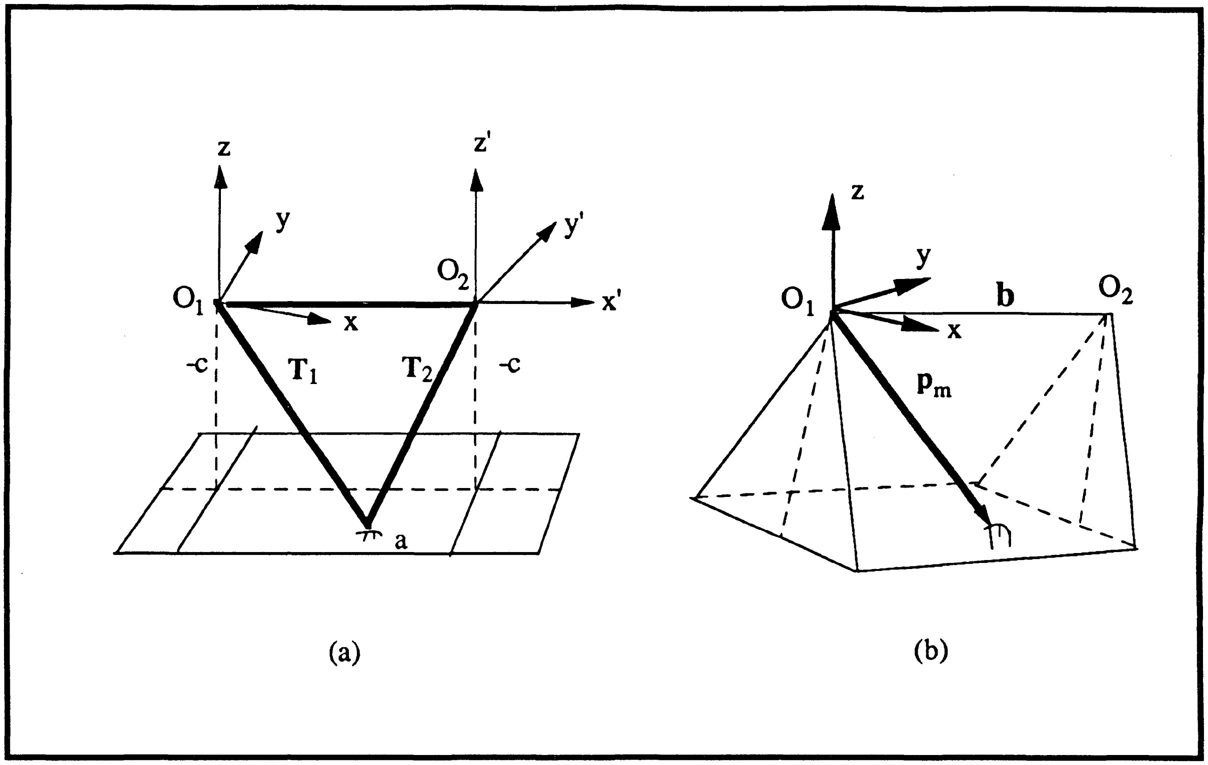


Figure 3-2: Model vector space (a) two conjugate rays intersect (b) the model space

3.3.3 The object vector space.

This is normally a three-dimensional right handed Cartesian system related to the object being mapped (Fig 3-3). A point in this space will be defined by a vector whose components are given as (X_A,Y_A,Z_A). The XYZ system may be geocentric Cartesian, which is useful for mapping a large portion of the earth's surface. Preferably, the system may be a local three-dimensional Cartesian system whose origin and orientation are suitably selected so that all parts of the object have positive and manageable coordinate values (Ghosh 1988). A three-dimensional similarity transformation may be used to transform between the local and the geocentric systems. The object space system may also be a geodetic system (φ,λ,h), which is not used directly in photogrammetry because it is not an orthogonal system, and two of the coordinates are in angular units. If the object space is described in such a system, it is usually transformed to either a geocentric or a local Cartesian system using standard formulas [Vanicek and Krakiwsky 1986]. The object space system could also be in a map projection system (E,N,h). Since these are derived from the geodetic system coordinates, and are related to the spheroid surface, they are also non-orthogonal. However, they may be used if earth curvature is accounted for, or they may be returned to the Cartesian system if the projection system is known.

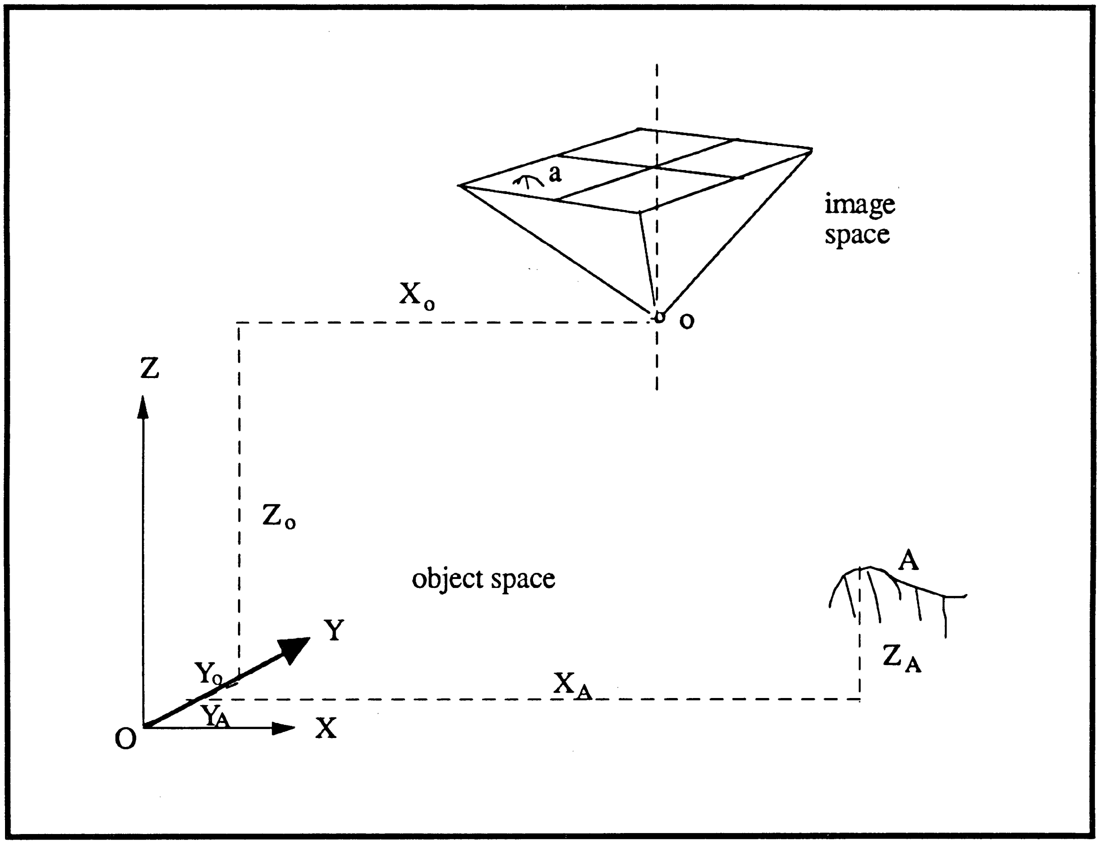


Figure 3-3: The object vector space

3.4 The Basic Transformation Operator

To begin with, vector scaling, rotation and translation which are so fundamental to all the derivations that follow are obtained from an application of elementary vector algebra to Figure 3-4.

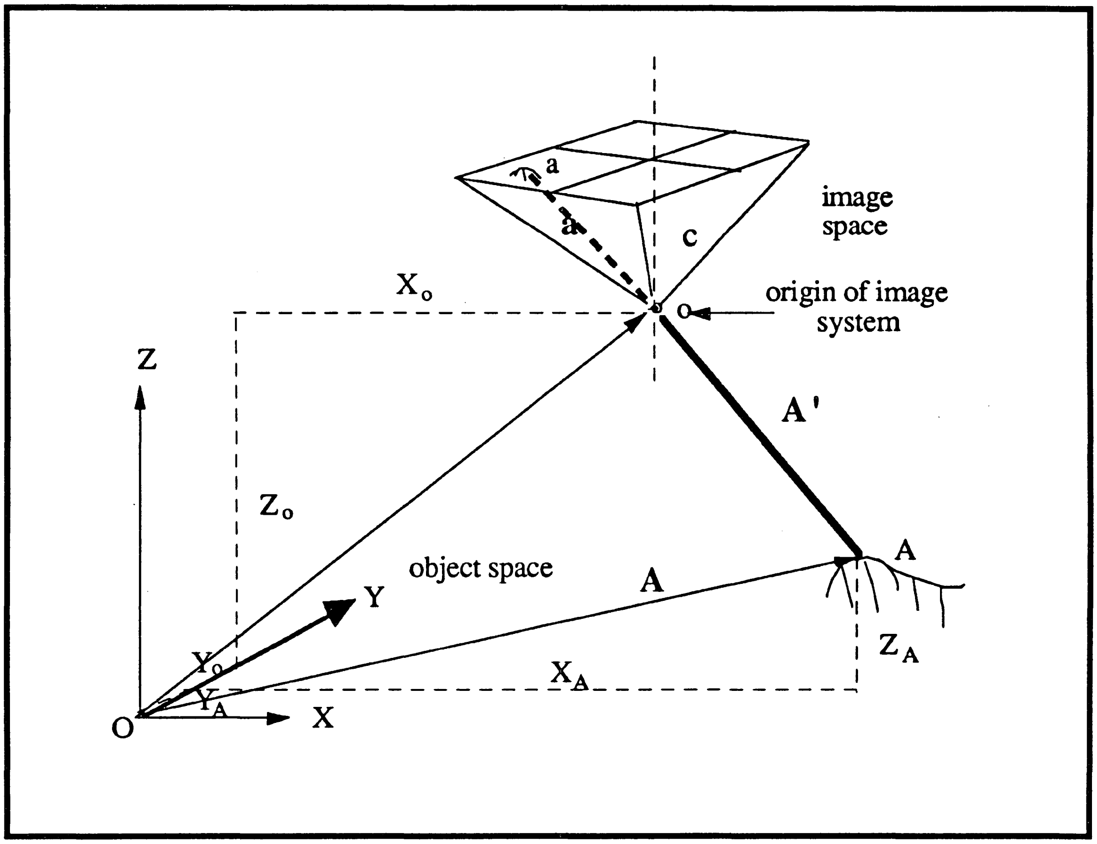


Figure 3-4. Vector Space Transformations

OA is a position vector in the object space Cartesian coordinate system with components (X_A, Y_A, Z_A). The corresponding image space vector oa is shown in an image space Cartesian system with components (x_a, y_a, z_a). Imagine that the image space system is shifted such that its origin (o) coincides with the origin (O) of the object space system. Let this image space be scaled to the size of the object space such that:

$$oA = A' = s\,\mathbf{a} = s\begin{bmatrix} x_a \\ y_a \\ z_a \end{bmatrix} = \begin{bmatrix} X'_a \\ Y'_a \\ Z'_a \end{bmatrix} \qquad (3\text{-}12)$$

where s is the scale. Equation 3-12 expresses the collinearity between the vectors **a** (or oa) and vector A' (or oA). Basically, when two vectors are collinear, one is a scalar multiple of the other.

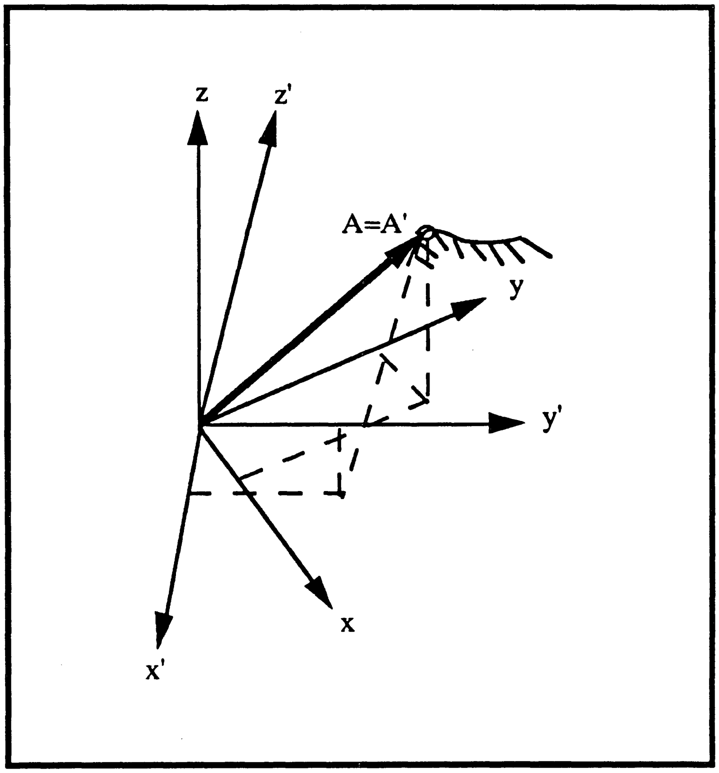


Figure 3-5: Coincident object space and expanded image space systems

In the coincident origin situation, both the object space vector **A** and the scaled image space vector **A**' are the same (i.e. A = A'), but in differently oriented systems (Figure 3-5). In Cartesian form we may write

$$\begin{bmatrix} X_A \\ Y_A \\ Z_A \end{bmatrix} = \begin{bmatrix} X'_a \\ Y'_a \\ Z'_a \end{bmatrix} \qquad (3\text{-}13a)$$

Let the unit vectors of the object space system be i, j, k and those of the image space system i', j', k', then Equation (3-13a) may be written as

$$X_A i + Y_A j + Z_A k = X'_a i' + Y'_a j' + Z'_a k' \qquad (3\text{-}13b)$$

Now taking the inner product of (3-13b) with the object space unit vector i we obtain

$$X_A i \cdot i + Y_A j \cdot i + Z_A k \cdot i = X'_a i' \cdot i + Y'_a j' \cdot i + Z'_a k' \cdot i$$

which, since $j \cdot i = k \cdot i = 0$ may be written as

$$X_A = X'_a i' \cdot i + Y'_a j' \cdot i + Z'_a k' \cdot i \qquad (3\text{-}14a)$$

Similarly for unit vectors j and k we have

$$Y_A = X'_a i' \cdot j + Y'_a j' \cdot j + Z'_a k' \cdot j \qquad (3\text{-}14b)$$

$$Z_A = X'_a i' \cdot k + Y'_a j' \cdot k + Z'_a k' \cdot k \qquad (3\text{-}14c)$$

Equations (3-14) may be put in the form

$$\begin{bmatrix} X_A \\ Y_A \\ Z_A \end{bmatrix} = \begin{bmatrix} i' \cdot i & j' \cdot i & k' \cdot i \\ i' \cdot j & j' \cdot j & k' \cdot j \\ i' \cdot k & j' \cdot k & k' \cdot k \end{bmatrix} \begin{bmatrix} X'_a \\ Y'_a \\ Z'_a \end{bmatrix} \qquad (3\text{-}15)$$

but $\begin{bmatrix} X'_a \\ Y'_a \\ Z'_a \end{bmatrix} = s \begin{bmatrix} x_a \\ y_a \\ z_a \end{bmatrix}$ from (3-12), thus (3-15) is the same as

$$\begin{bmatrix} X_A \\ Y_A \\ Z_A \end{bmatrix} = s\,M \begin{bmatrix} x_a \\ y_a \\ z_a \end{bmatrix} \qquad (3\text{-}16a)$$

or, in symbolic vector form, the vector scaling and rotation to bring an image space vector to the object space system is:

$$\mathbf{P}_A = s\mathbf{M}\mathbf{P}_a \qquad (3\text{-}16b)$$

where s is the scale factor and M is the rotation matrix.

Consider again Figure 3-4 the situation in which the origin of the image space is not coincident with the object space origin. Then, applying vector addition to the vector triangle formed by Oo, oA, OA, we may write:

$$OA = Oo + oA \tag{3-17a}$$

and applying (3-16b) to oA, equation (3-17a) becomes

$$OA = Oo + sM(oa). \tag{3-17b}$$

But

$$OA = \mathbf{P}_A = \begin{bmatrix} X_A \\ Y_A \\ Z_A \end{bmatrix}, Oo = \mathbf{P}_o = \begin{bmatrix} X_o \\ Y_o \\ Z_o \end{bmatrix}, oa = \mathbf{P}_a = \begin{bmatrix} x_a \\ y_a \\ z_a \end{bmatrix}$$

therefore we may write (3-17b) in component form

$$\begin{bmatrix} X_A \\ Y_A \\ Z_A \end{bmatrix} = \begin{bmatrix} X_o \\ Y_o \\ Z_o \end{bmatrix} + s\,M \begin{bmatrix} x_a \\ y_a \\ z_a \end{bmatrix}$$

or in symbolic vector form

$$\mathbf{P}_A = \mathbf{P}_O + sM\mathbf{P}_a \tag{3-17c}$$

where s is the scale, $\mathbf{P}_O$ is the translation vector and M is the matrix of direction cosines denoted as:

$$M = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{23} & m_{33} \end{bmatrix}$$

The elements of M are usually expressed in terms of rotations around the coordinate axes. Equation 3-17c is the basic transformation operator (the linear type) which scales, rotates and translates an element of the image space system into an element in the object space system, and it serves a major purpose in the derivation of all mathematical models which follow. Note that because M is orthogonal, equation 3-17c may be written in inverse form as:

$$\mathbf{P}_a = \frac{1}{s} M^T \Delta \tag{3-18}$$

where

$$\Delta = \mathbf{P_A} - \mathbf{P_o}$$

We recognize the simplicity of equations 3-17c and 3-18 as compared to the single variable approach. Nevertheless, these equations do not offer significant insight into the geometrical structure of the transformation process. To enhance their significance and provide a set of working tools, equations 3-17c and 3-18 will be developed into a set of generalised vector space relations in the next section.

3.4.1 The ARDOVS Relations of Vector Spaces

In this section we develop a new methodology for using vector symbology in analytical photogrammetry. This concept which we shall call ARDOVS, a short name for Apparent Reciprocal Directions Of Vector Spaces" is based on an intuitive understanding of the geometrical interpretations of equations 3-17c and 3-18, and it regards the two vector spaces involved in a transformation as being a combination of an **R**-space (fixed space) and a **C**-space (movable space). However, rather than embark on rigorous mathematical proof, we content ourselves with the derivation of equations 3-15, 3-17c and 3-18, and simply express the theoretical basis for this concept in form of a set of hypotheses whose validity may be tested by intuition.

To provide a basis for the hypotheses, we perform a manipulation of eqns. 3-17c and 3-18 by representing the rotation matrix in terms of both its column vectors and its row vectors:

$$\mathrm{M} = [\ \mathbf{C}_1 \quad \mathbf{C}_2 \quad \mathbf{C}_3\ ] \tag{3-19}$$

and

$$\mathrm{M} = \begin{bmatrix} \mathbf{R}_1 \\ \mathbf{R}_2 \\ \mathbf{R}_3 \end{bmatrix} \tag{3-20}$$

where

$$\mathbf{C}_1 = \begin{bmatrix} m_{11} \\ m_{21} \\ m_{31} \end{bmatrix}, \quad \mathbf{C}_2 = \begin{bmatrix} m_{12} \\ m_{22} \\ m_{32} \end{bmatrix}, \quad \mathbf{C}_3 = \begin{bmatrix} m_{13} \\ m_{23} \\ m_{33} \end{bmatrix} \tag{3-21}$$

$$\begin{aligned} \mathbf{R}_1 &= [\, m_{11} \;\; m_{12} \;\; m_{13} \,], \\ \mathbf{R}_2 &= [\, m_{21} \;\; m_{22} \;\; m_{23} \,], \\ \mathbf{R}_3 &= [\, m_{31} \;\; m_{32} \;\; m_{33} \,] \end{aligned} \tag{3-22}$$

Using (3-20) in (3-17c) and (3-19) in (3-18), a simple manipulation leads to the following illuminating equations:

$$\mathbf{P}_A = \mathbf{P}_o + s \begin{bmatrix} \mathbf{R}_1 \cdot \mathbf{P}_a \\ \mathbf{R}_2 \cdot \mathbf{P}_a \\ \mathbf{R}_3 \cdot \mathbf{P}_a \end{bmatrix} \tag{3-23}$$

$$\mathbf{P}_a = \frac{1}{s} \begin{bmatrix} \mathbf{C}_1 \cdot \Delta \\ \mathbf{C}_2 \cdot \Delta \\ \mathbf{C}_3 \cdot \Delta \end{bmatrix} \tag{3-24}$$

where $\mathbf{P}_a$ and $\mathbf{P}_A$ are image and object space vectors respectively, and $\mathbf{R}_1, \mathbf{R}_2, \mathbf{R}_3$, $\mathbf{C}_1, \mathbf{C}_2, \mathbf{C}_3$ are the row and column vectors of M respectively. Equation (3-23) reveals the hitherto hidden fact that the object space position vector is in fact the projection of the image space vector onto the row space of the rotation matrix plus a shift, while (3-24) shows that the image space position vector is nothing but the projection of the object space position vector referred to the camera position, onto the column space of the rotation matrix; in each case appropriate scales are applied.

The following hypotheses allow us to state some general rules, which are then applied to the formulation of the universal relations between any two photogrammetric vector spaces (refer to Figure 3.6):

1. Any vector space has a set of natural (or real) direction axes as seen by the elements within the space, and a set of apparent direction axes as seen by elements in a neighbouring space. The apparent direction axes of two vector spaces are reciprocal and signify the way that the natural axes of one space are seen from the other.
2. Between any two vector spaces, one space is fixed while the other is moving or movable.
3. For 3-D vector spaces, the apparent direction axes of the fixed space are constituted by vectors $\mathbf{R}_1$, $\mathbf{R}_2$, $\mathbf{R}_3$ and are in fact the row vectors of the familiar rotation matrix. In the ARDOVS concept, this space is called the **R**-space (Figure 3.6a), where the R simply indicates that its apparent direction axes are the row vectors of the rotation matrix.

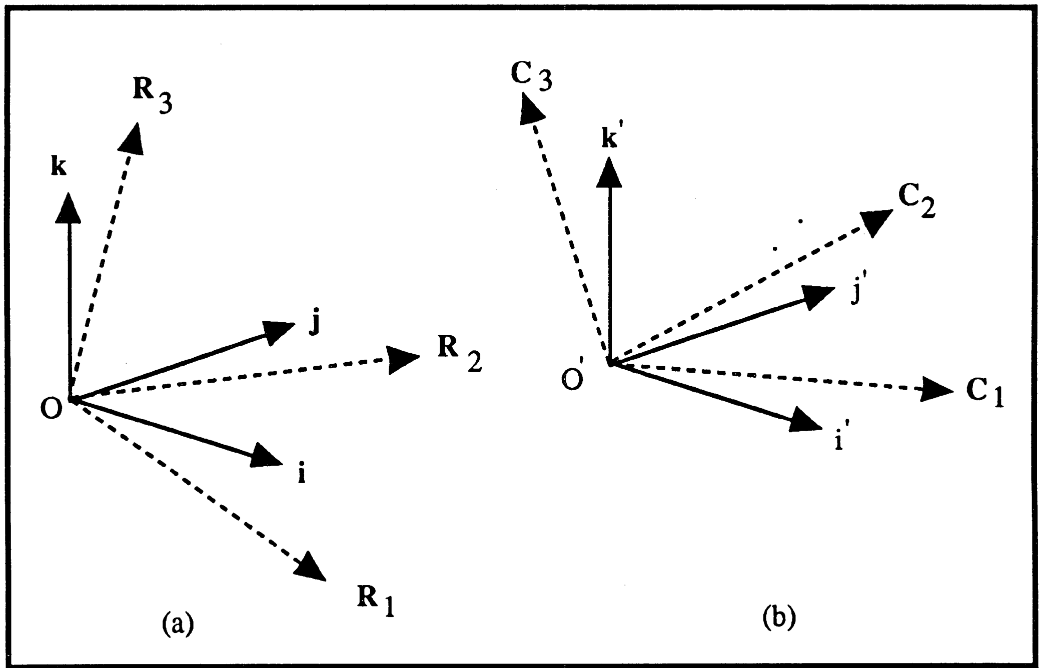


Figure 3-6: Natural and Apparent Direction Axes of Vector Spaces: (a) the R-space (b) the C-space

4. For 3-D vector spaces, the apparent direction axes of the movable space are constituted by vectors $\mathbf{C}_1$, $\mathbf{C}_2$, $\mathbf{C}_3$ and are in fact the column vectors of the familiar rotation matrix.

In the ARDOVS concept, this space is called the **C**-space (Figure 3.6b), where the C signifies that its direction axes are the column vectors of the rotation matrix.

5. The origin of the **C**-space is a vector element in the **R**-space.

Based on these hypotheses, the following rules are established:

1. An element of one space can only cross to the other space by projection onto the apparent directions of the new space. If the scales of the two spaces are different, the projection must be scaled to conform to the new space.
2. Any element moving from the **R**-space to the **C**-space must reduce itself before projection, by the vector which locates the origin of the **C**-space.
3. Any element moving from the **C**-space to the **R**-space must add to itself after projection, the vector which locates the origin of the **C**-space.

Using rules 1 and 3, and rules 1 and 2, we state the two ARDOVS relations as follows:

1. ARDOVS relation 1: **C**-space to **R**-space:

$$\mathbf{P}_A = \mathbf{P}_o + s \begin{bmatrix} \mathbf{R}_1 \cdot \mathbf{P}_a \\ \mathbf{R}_2 \cdot \mathbf{P}_a \\ \mathbf{R}_3 \cdot \mathbf{P}_a \end{bmatrix} \qquad (3\text{-}23a)$$

2. ARDOVS relation 2: **R**-space to **C**-space:

$$\mathbf{P}_a = \frac{1}{s} \begin{bmatrix} \mathbf{C}_1 \cdot (\mathbf{P}_A - \mathbf{P}_o) \\ \mathbf{C}_2 \cdot (\mathbf{P}_A - \mathbf{P}_o) \\ \mathbf{C}_3 \cdot (\mathbf{P}_A - \mathbf{P}_o) \end{bmatrix} \qquad (3\text{-}24b)$$

where $\mathbf{P}_a$ is an element of the **C**-space and $\mathbf{P}_A$ an element of the **R**-space. $\mathbf{C}_1, \mathbf{C}_2, \mathbf{C}_3$ are the apparent directions of the **C**-space and $\mathbf{R}_1$, $\mathbf{R}_2$, $\mathbf{R}_3$ of the **R**-space.

We therefore have a powerful and universal tool for all applications in photogrammetry including development of analytical computational algorithms. Furthermore, we observe that these relations serve a dual purpose from a mathematical viewpoint. Firstly, they may be used to transform elements between two vector spaces if the apparent directions and other parameters are given; and secondly they may be used to derive computational algorithms to determine the apparent direction vectors and other parameters. In any case, to apply them to a particular problem, all we have to do is to identify which space is the fixed or **R**-space and which is the movable or **C**-space. In this way we remove the arbitrariness often involved in the use of transformation equations in computational photogrammetry. The power of ARDOVS will be demonstrated when applied to specific problems in the next section. However, it is helpful to state the analytic forms of the direction vectors and their derivatives. This will simplify matters later in the applications. Considering the rotation matrix in equation 3-15 and the direction vectors in Figure 3-6, the geometric relationships of the direction vectors may be deduced. Using the method of systematic rotations around coordinate axes, the standard analytic forms of the row and the column direction vectors may be derived [Methley 1986, Moffitt and Mikhail 1980]. These are stated as follows:

C-space direction vectors:

$$\mathbf{C}_1 = \begin{bmatrix} \cos\phi \cos k \\ -\cos\phi \sin k \\ \sin\phi \end{bmatrix}, \quad \mathbf{C}_2 = \begin{bmatrix} \sin\omega\sin\phi \cos k + \cos\omega\sin k \\ -\sin\omega\sin\phi \sin k + \cos\omega\cos k \\ -\sin\omega\cos\phi \end{bmatrix},$$

$$\mathbf{C}_3 = \begin{bmatrix} -\cos\omega\sin\phi \cos k + \sin\omega\sin k \\ \cos\omega\sin\phi \sin k + \sin\omega\cos k \\ \cos\omega\cos\phi \end{bmatrix} \qquad (3\text{-}25)$$

the gradients of the **C**-space direction vectors with respect to the rotation elements $(\omega, \varphi, \kappa)$:

$$\mathbf{C}_{1\omega} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{C}_{2\omega} = -\mathbf{C}_3, \quad \mathbf{C}_{3\omega} = \mathbf{C}_2$$

$$\mathbf{C}_{1\phi} = \begin{bmatrix} -\sin\phi \cos k \\ \sin\phi \sin k \\ \cos\phi \end{bmatrix}, \quad \mathbf{C}_{2\phi} = \begin{bmatrix} \sin\omega\cos\phi \cos k \\ -\sin\omega\cos\phi \sin k \\ \sin\omega\sin\phi \end{bmatrix},$$

$$\mathbf{C}_{3\phi} = \begin{bmatrix} -\cos\omega\cos\phi \cos k \\ \cos\omega\cos\phi \sin k \\ -\cos\omega\sin\phi \end{bmatrix}$$

$$\mathbf{C}_{1k} = \begin{bmatrix} -\cos\phi \sin k \\ -\cos\phi \cos k \\ 0 \end{bmatrix} \quad \mathbf{C}_{2k} = \begin{bmatrix} -\sin\omega\sin\phi \sin k + \cos\omega\cos k \\ -\sin\omega\sin\phi \cos k - \cos\omega\sin k \\ 0 \end{bmatrix}$$

$$\mathbf{C}_{3k} = \begin{bmatrix} \cos\omega\sin\phi \sin k + \sin\omega\cos k \\ \cos\omega\sin\phi \cos k - \sin\omega\sin k \\ 0 \end{bmatrix} \tag{3-26}$$

R-space direction vectors:

$$\mathbf{R}_1 = \begin{bmatrix} \cos\phi \cos k \\ \sin\omega \sin\phi \cos k + \cos\omega \sin k \\ -\cos\omega \ \sin\phi \cos k + \sin\omega \sin k \end{bmatrix}^T$$

$$\mathbf{R}_2 = \begin{bmatrix} -\cos\phi \sin k \\ -\sin\omega \sin\phi \sin k + \cos\omega \cos k \\ \cos\omega \ \sin\phi \sin k + \sin\omega \cos k \end{bmatrix}^T$$

$$\mathbf{R}_3 = \begin{bmatrix} \sin\phi \\ -\sin\omega\cos\phi \\ \cos\omega\ \cos\phi \end{bmatrix}^T \tag{3-27}$$

The gradients of the **R**-space direction vectors with respect to the rotation elements:

$$\mathbf{R}_{1\omega} = \begin{bmatrix} 0 \\ \cos\omega\sin\phi\,\mathrm{cosk} - \sin\omega\,\mathrm{sink} \\ \sin\omega\ \sin\phi\,\mathrm{cosk} + \cos\omega\,\mathrm{sink} \end{bmatrix}^T ,$$

$$\mathbf{R}_{2\omega} = \begin{bmatrix} 0 \\ -\cos\omega\sin\phi\,\mathrm{sink} - \sin\omega\,\mathrm{cosk} \\ -\sin\omega\ \sin\phi\,\mathrm{sink} + \cos\omega\,\mathrm{cosk} \end{bmatrix}^T ,\ \mathbf{R}_{3\omega} = \begin{bmatrix} 0 \\ -\cos\omega\cos\phi \\ -\sin\omega\ \cos\phi \end{bmatrix}^T$$

$$\mathbf{R}_{1\phi} = \begin{bmatrix} -\sin\phi\,\mathrm{cosk} \\ \sin\omega\cos\phi\,\mathrm{cosk} \\ -\cos\omega\ \cos\phi\,\mathrm{cosk} \end{bmatrix}^T ,\mathbf{R}_{2\phi} = \begin{bmatrix} \sin\phi\,\mathrm{sink} \\ -\sin\omega\cos\phi\,\mathrm{sink} \\ \cos\omega\ \cos\phi\,\mathrm{sink} \end{bmatrix}^T ,$$

$$\mathbf{R}_{3\phi} = \begin{bmatrix} \cos\phi \\ \sin\omega\sin\phi \\ -\cos\omega\ \sin\phi \end{bmatrix}^T$$

$$\mathbf{R}_{1k} = \begin{bmatrix} -\cos\phi\,\mathrm{sink} \\ -\ \sin\omega\sin\phi\,\mathrm{sink} + \cos\omega\,\mathrm{cosk} \\ \cos\omega\ \sin\phi\,\mathrm{sink} + \sin\omega\,\mathrm{cosk} \end{bmatrix}^T ,$$

$$\mathbf{R}_{2k} = \begin{bmatrix} -\cos\phi\,\mathrm{cosk} \\ -\ \sin\omega\sin\phi\,\mathrm{cosk} - \cos\omega\,\mathrm{sink} \\ \cos\omega\ \sin\phi\,\mathrm{cosk} - \sin\omega\,\mathrm{sink} \end{bmatrix}^T ,\ \mathbf{R}_{3k} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}^T \tag{3-28}$$

Practically, this technique is fundamentally revolutionary in that the whole concept of coordinate transformation in photogrammetry is portrayed in a new light with incredible simplicity. This will change the way we understand and apply vector transformations in analytical work. However, the advantages can be appreciated only when applied to specific problems, and this we shall do in the rest of this chapter.

3.5 Vector-Based Mathematical models.

In this section, we apply the ARDOVS relations to derive the mathematical models essential for analytical photogrammetry. These tools are applied by specifying the **R**-space and the **C**-space for each problem. We then apply relation 1 or 2 and simplify the resulting equations for the photogrammetric task.

3.5.1 2-D Collinearity Equations.

The collinearity equations are an analytic function which connect the image and the object vector spaces. Perhaps it is necessary at this juncture to examining the existing formulation of the collinearity equations in the illumination provided by the new ARDOVS concept. It is observed that many authors unconsciously assign the image space (the photograph) as the **R**-space and the object space as the **C**-space. They then formulate the collinearity equations in the **R**-space (image space) by projecting the object space points to the image space to avoid the complexity of combined adjustment [Wong 1980, Wolf 1983, Methley 1986, Ghosh 1988]. While this practice does not affect the computed coordinates of points, the rotation angles from such procedure are the negative of what they really are. In the implementation of the traditional formulation one had to mechanically reverse the signs of the angles to obtain the correct result. This is not a convenient thing to do particularly when a large block of photographs is to be prepared for processing by the real time subsystem of a photogrammetric workstation. In this study, we shall assign the **R**-

space to the object space and the C-space to the image space; this is the natural phenomenon since it is the camera that moves and not the object. We then formulate the collinearity equations in the image space (or the observation space) by sending the object space points to the image space. This has the advantages of avoiding combined adjustment, and providing the orientation angles in their correct attitude, thereby eliminating completely the confusion so often associated with negative angles, particularly when preparing data for restoration of stereomodels for the collection of the vector data.

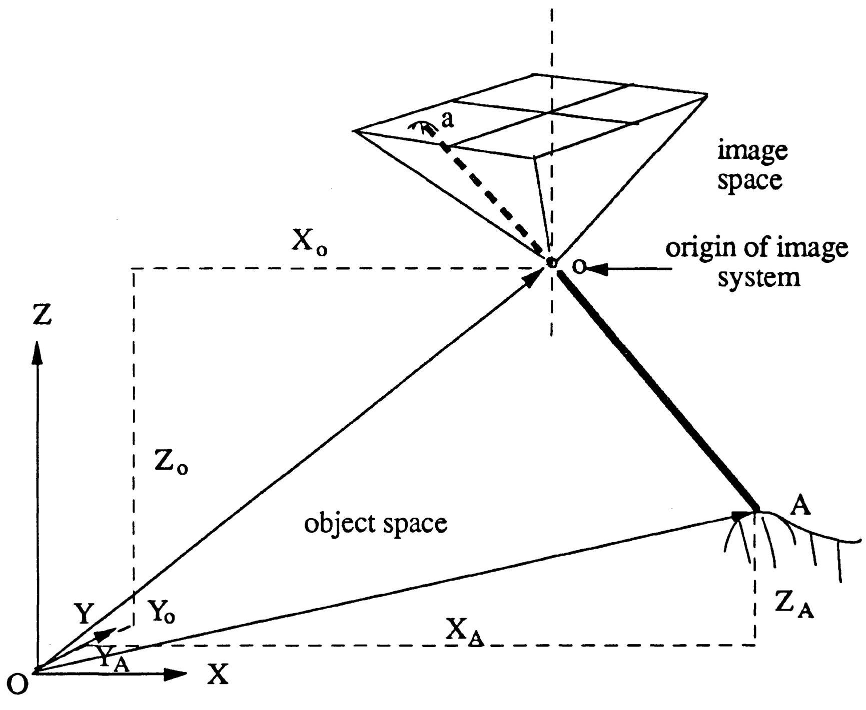


Figure 3-7. 2-D Collinearity by R-space to C-space Transformation

Let us specify the image space (C-space) equivalent $\mathbf{P_a}$ of an object space (R-space) point $\mathbf{P_A}$ as:

$$\mathbf{P_a} = \begin{bmatrix} x_a \\ y_a \\ -c \end{bmatrix}$$

and using the ARDOVS relations 2 (eqn. 3-24a) we have:

$$\begin{bmatrix} x_a \\ y_a \\ -c \end{bmatrix} = \frac{1}{s} \begin{bmatrix} C_1 \cdot (P_A - P_o) \\ C_2 \cdot (P_A - P_o) \\ C_3 \cdot (P_A - P_o) \end{bmatrix} \tag{3-29}$$

Now we are completely in the C-space (inside the camera if you wish) and equation (3-29) would be the equation we wanted if the image space were a true 3-D space. But from a practical standpoint, only the x and y photographic coordinates are measured for each point, the z-coordinate is a constant for all points in the image space. From an adjustment point of view, the first two equations of (3-29) are the primary observation equations which contribute new information to an adjustment process with every measured point in the image space. The third equation may be viewed as a functional constraint equation since it is constant for all points on the photograph. It enforces the perspectivity of the projection. Nevertheless, it is advantageous to reduce the number of equations from three to two for each point imaged on one photograph by dividing each of the first two equations (in 3-29) by the third (i.e. 3D to 2D). This technique reduces the number of unknowns by the removal of the scale factor s while still retaining the constraint.

Thus, we have:

$$\begin{aligned} -\frac{x_a}{c} &= \frac{C_1 \cdot (P_A - P_o)}{C_3 \cdot (P_A - P_o)} \\ -\frac{y_a}{c} &= \frac{C_2 \cdot (P_A - P_o)}{C_3 \cdot (P_A - P_o)} \end{aligned} \tag{3-30}$$

Equations (3-30) are the 2-D collinearity equations (which may be more appropriately called reduced collinearity equations) in vector notation. By rationalization, (3-30) may be written as

$$\begin{aligned} (k_x C_3 + C_1) \cdot P_A - (k_x C_3 + C_1) \cdot P_o = 0 \\ (k_y C_3 + C_2) \cdot P_A - (k_y C_3 + C_2) \cdot P_o = 0 \end{aligned} \tag{3-31}$$

where $k_x = x_a / c$ and $k_y = y_a / c$.

Applying the distributive property of inner product of vectors, we may put (3-31) in the equivalent form:

$$
\begin{aligned}
&k_x C_3 \cdot \mathbf{P}_A + C_1 \cdot \mathbf{P}_A - k_x C_3 \cdot \mathbf{P}_o - C_1 \cdot \mathbf{P}_o = 0 \\
&k_y C_3 \cdot \mathbf{P}_A + C_2 \cdot \mathbf{P}_A - k_y C_3 \cdot \mathbf{P}_o - C_2 \cdot \mathbf{P}_o = 0
\end{aligned}
\qquad (3\text{-}32)
$$

Equations (3-32) are the C-space formulation of the collinearity condition functional model which describes the image space to object space relationship mathematically. Note the simplicity of this set of equations in representing the object space ray between an object point (with position vector $\mathbf{P}_A$) and the camera location (with position vector $\mathbf{P}_O$), i.e. the ray oA = **A'** (Figure 3-7) as a series of projections onto the apparent direction axes of the image space. Note also the symmetry of these equations. In general, equation 3-32 makes the geometrical concept of analytical photogrammetry very transparent. What is more important, all the derivatives of the collinearity equation with respect to all the variables involved therein may be obtained by standard vector gradient operators which are few in number and may be routinely applied (see equations 3-25 - 3-28). This is one demonstration of the power of the ARDOVS methodology.

The functional form of the collinearity equation (3-32) may be represented by (see appendix I)

$$f(\mathbf{X}, \mathbf{L}) = 0$$

where **X** is the vector of unknown parameters (depending on the chosen application) and **L** is the vector of observed quantities.

The linearised mathematical model is

$$\mathbf{A}\delta + \mathbf{B}\mathbf{v} - \mathbf{f} = 0$$

where

$$\mathbf{A} = \frac{\partial f(\mathbf{X},\mathbf{L})}{\partial \mathbf{X}}, \quad \mathbf{B} = \frac{\partial f(\mathbf{X},\mathbf{L})}{\partial \mathbf{L}}, \quad \mathbf{f} = -f(\mathbf{X}^o,\mathbf{L}^o)$$

all evaluated at the approximate values, and

$$\delta = \mathbf{X} - \mathbf{X}^o \;,\quad \mathbf{v} = \mathbf{L} - \mathbf{L}^o$$

In order to explicitly express the elements of matrices **A**, **B**, and of the vector **f**, the derivatives of the vectors involved in (3-32) with respect to the variables involved in it are first obtained by applying the vector differential operators in equations 3-6 - 3-10:

Let

$$\mathbf{H}_x = (k_x C_3 + C_1)$$

$$\mathbf{H}_y = (k_y C_3 + C_2)$$

then

$$\mathbf{H}_{x\omega} = \frac{\partial(k_x C_3 + C_1)}{\partial\omega} = (k_x \frac{\partial C_3}{\partial\omega} + \frac{\partial C_1}{\partial\omega}) = k_x C_{3\omega} + C_{1\omega}$$

$$\mathbf{H}_{x\phi} = \frac{\partial(k_x C_3 + C_1)}{\partial\phi} = (k_x \frac{\partial C_3}{\partial\phi} + \frac{\partial C_1}{\partial\phi}) = k_x C_{3\phi} + C_{1\phi}$$

$$\mathbf{H}_{xk} = \frac{\partial(k_x C_3 + C_1)}{\partial k} = (k_x \frac{\partial C_3}{\partial k} + \frac{\partial C_1}{\partial k}) = k_x C_{3k} + C_{1k}$$

and by similar consideration we have

$$\mathbf{H}_{y\omega} = k_y C_{3\omega} + C_{2\omega}$$

$$\mathbf{H}_{y\phi} = k_y C_{3\phi} + C_{2\phi} \qquad (3\text{-}33)$$

$$\mathbf{H}_{yk} = k_y C_{3k} + C_{2k}$$

where $C_{1\omega}$, $C_{2\omega}$, $C_{3\omega}$ are the derivatives of the column vectors of the rotation matrix with respect to omega (ω), and similarly for phi and kappa (as given in eqns. 3-25, 3-26). Note the simplicity of obtaining the vector derivatives in the vector format. The elements of the design matrices for the mathematical model may be formed as follows:

Matrix A: the elements with respect to camera orientation (ω,ϕ,k) and location (X_o,Y_o,Z_o) are (note that: $\frac{\partial \mathbf{P}_o}{\partial X_o} = \mathbf{i}, \frac{\partial \mathbf{P}_o}{\partial Y_o} = \mathbf{j}, \frac{\partial \mathbf{P}_o}{\partial Z_o} = \mathbf{k}$):

$$\begin{bmatrix} \mathbf{H}_{x\omega}\cdot\Delta & \mathbf{H}_{x\phi}\cdot\Delta & \mathbf{H}_{xk}\cdot\Delta & -\mathbf{H}_x\cdot\mathbf{i} & -\mathbf{H}_x\cdot\mathbf{j} & -\mathbf{H}_x\cdot\mathbf{k} \\ \mathbf{H}_{y\omega}\cdot\Delta & \mathbf{H}_{y\phi}\cdot\Delta & \mathbf{H}_{yk}\cdot\Delta & -\mathbf{H}_y\cdot\mathbf{i} & -\mathbf{H}_y\cdot\mathbf{j} & -\mathbf{H}_y\cdot\mathbf{k} \end{bmatrix} \qquad (3\text{-}34a)$$

and those with respect to the object point (X_A,Y_A,Z_A), are (noting also that: $\frac{\partial \mathbf{P}_A}{\partial X_A} = \mathbf{i}, \frac{\partial \mathbf{P}_A}{\partial Y_A} = \mathbf{j}, \frac{\partial \mathbf{P}_A}{\partial Z_A} = \mathbf{k}$):

$$\begin{bmatrix} \mathbf{H}_x\cdot\mathbf{i} & \mathbf{H}_x\cdot\mathbf{j} & \mathbf{H}_x\cdot\mathbf{k} \\ \mathbf{H}_y\cdot\mathbf{i} & \mathbf{H}_y\cdot\mathbf{j} & \mathbf{H}_y\cdot\mathbf{k} \end{bmatrix} \qquad (3\text{-}34b)$$

Matrix B: this is a scalar matrix, which is

$$\begin{bmatrix} \frac{\mathbf{C}_3\cdot\Delta}{c} & 0 \\ 0 & \frac{\mathbf{C}_3\cdot\Delta}{c} \end{bmatrix} = \frac{\mathbf{C}_3\cdot\Delta}{c}\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \qquad (3\text{-}34c)$$

and the constant vector **f** is given by

$$\begin{bmatrix} -\ \mathbf{H}_x\cdot\Delta \\ -\ \mathbf{H}_y\cdot\Delta \end{bmatrix}$$

where

$$\Delta = \mathbf{P}_A - \mathbf{P}_o$$

Note that since matrix B is regular and in fact a scalar matrix, the C-space formulation of the collinearity equation 3-32 is not a combined adjustment representation. The design matrices are completed by multiplying them by the inverse of matrix B. Note the simplicity of the design matrices whose entries are simply a series of projections of the R-space natural axes basis vectors (**i**, **j**, **k**) and its elements onto the apparent directions of the C-space. Furthermore the structured form of these matrices makes computer implementation a trivial process. The algorithms for some applications of this functional model are given in section 3.6.

3.5.2 3-D collinearity equations.

These are equations which represent the relationship between the model space and the object space. Since the model space is a real 3-D space in which every point is uniquely represented by a set of coordinates, it is similar to the object space in every respect except for its scale, orientation and its spatial location with respect to the object space. Thus, this transformation is often called a 3-D similarity transformation.

To apply the ARDOVS relation, we specify the **R**-space as the object space and the **C**-space as the model space. As stated before, formulation in the observation space has the virtues of computational simplicity and exactness. By replacing the image space in Fig. 3-7 with the model space shown in Fig. 3-2b, the model-object situation is depicted in Fig. 3-8 Defining the vector Pa as a model space position vector, i.e.

$$\mathbf{P}_a = \mathbf{P}_m = \begin{bmatrix} x_m \\ y_m \\ z_m \end{bmatrix}$$

and substituting this into the ARDOVS relation 2, (3-24a) we obtain:

$$\begin{bmatrix} x_m \\ y_m \\ z_m \end{bmatrix} = \frac{1}{s} \begin{bmatrix} \mathbf{C}_1 \cdot (\mathbf{P}_A - \mathbf{P}_o) \\ \mathbf{C}_2 \cdot (\mathbf{P}_A - \mathbf{P}_o) \\ \mathbf{C}_3 \cdot (\mathbf{P}_A - \mathbf{P}_o) \end{bmatrix} \qquad (3\text{-}35)$$

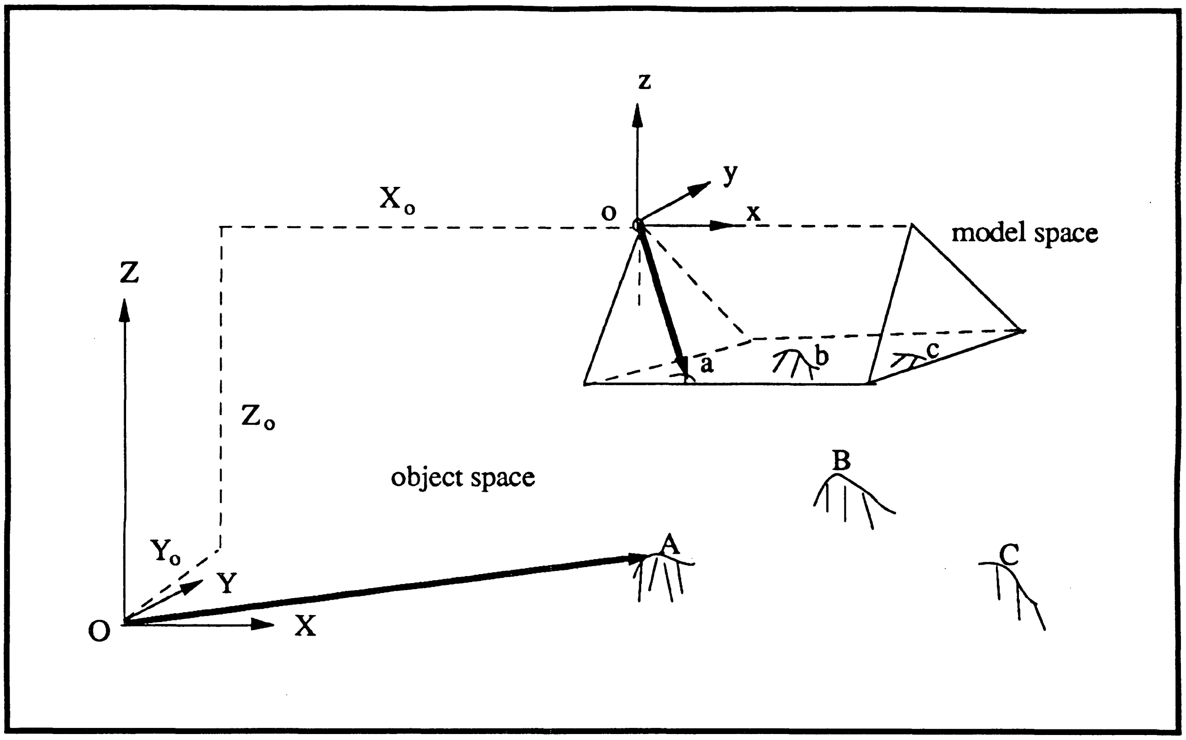


Figure 3.8: 3-D Collinearity by R-space to C-space Transformation

Now let $s' = 1/s$ then

equation (3-35) may be written in a simplified form :

$$
\begin{aligned}
&s'\mathbf{C}_1 \cdot \mathbf{P}_A - s'\mathbf{C}_1 \cdot \mathbf{P}_o - x_m = 0 \\
&s'\mathbf{C}_2 \cdot \mathbf{P}_A - s'\mathbf{C}_2 \cdot \mathbf{P}_o - y_m = 0 \qquad (3\text{-}36) \\
&s'\mathbf{C}_3 \cdot \mathbf{P}_A - s'\mathbf{C}_3 \cdot \mathbf{P}_o - z_m = 0
\end{aligned}
$$

Equation (3-36) is the **C**-space representation of the 3-D collinearity functional model which describes the relationship between the stereomodel and the object vector spaces. Transparently, (3-36) is seen to be a series of projections of the **R**-space elements onto the apparent directions of the **C**-space. This is another demonstration of the power of the new concept

Equation (3-36) may be represented in functional form by

$$f(\mathbf{X}, \mathbf{L}) = 0$$

where **X** is the vector of unknown variables to be determined (absolute orientation parameters) and **L** is the vector of observed model coordinates.

The linearised model is given by

$$\mathbf{A}\delta + \mathbf{B}\mathbf{v} - \mathbf{f} = 0$$

where

where **X** is the vector of unknown parameters and **L** is the vector of observed quantities.

$$\mathbf{A} = \frac{\partial f(\mathbf{X},\mathbf{L})}{\partial \mathbf{X}}, \quad \mathbf{B} = \frac{\partial f(\mathbf{X},\mathbf{L})}{\partial \mathbf{L}}, \quad \mathbf{f} = - f(\mathbf{X}^o,\mathbf{L}^o)$$

all evaluated at the approximate values, and

$$\delta = \mathbf{X} - \mathbf{X}^o, \quad \mathbf{v} = \mathbf{L} - \mathbf{L}^o$$

To express the elements of the design matrices **A**, **B**, and of the vector **f**, the column vectors of the rotation matrix and their gradients (as given in equations 3-25 and 3-26) are:

The design matrix with respect to orientation elements and scale (ω,ϕ,k, s') in that order is

$$\begin{bmatrix} s'C_{1\omega}\cdot\Delta & s'C_{1\phi}\cdot\Delta & s'C_{1k}\cdot\Delta & C_1\cdot\Delta \\ s'C_{2\omega}\cdot\Delta & s'C_{2\phi}\cdot\Delta & s'C_{2k}\cdot\Delta & C_2\cdot\Delta \\ s'C_{3\omega}\cdot\Delta & s'C_{3\phi}\cdot\Delta & s'C_{3k}\cdot\Delta & C_3\cdot\Delta \end{bmatrix} \qquad (3\text{-}37a)$$

and with respect to the origin of the model system (X_o,Y_o,Z_o)

$$\begin{bmatrix} -s'C_1\cdot\mathbf{i} & -s'C_1\cdot\mathbf{j} & -s'C_1\cdot\mathbf{k} \\ -s'C_2\cdot\mathbf{i} & -s'C_2\cdot\mathbf{j} & -s'C_2\cdot\mathbf{k} \\ -s'C_3\cdot\mathbf{i} & -s'C_3\cdot\mathbf{j} & -s'C_3\cdot\mathbf{k} \end{bmatrix} \qquad (3\text{-}37b)$$

while with respect to object point position (X_A,Y_A,Z_A) it is:

$$\begin{bmatrix} s'C_1\cdot\mathbf{i} & s'C_1\cdot\mathbf{j} & s'C_1\cdot\mathbf{k} \\ s'C_2\cdot\mathbf{i} & s'C_2\cdot\mathbf{j} & s'C_2\cdot\mathbf{k} \\ s'C_3\cdot\mathbf{i} & s'C_3\cdot\mathbf{j} & s'C_3\cdot\mathbf{k} \end{bmatrix} \qquad (3\text{-}37c)$$

The second design matrix **B** is the negative identity matrix in this case and confirms that the **C**-space formulation is not a combine adjustment case.

$$\mathbf{B} = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \quad , \tag{3-37d}$$

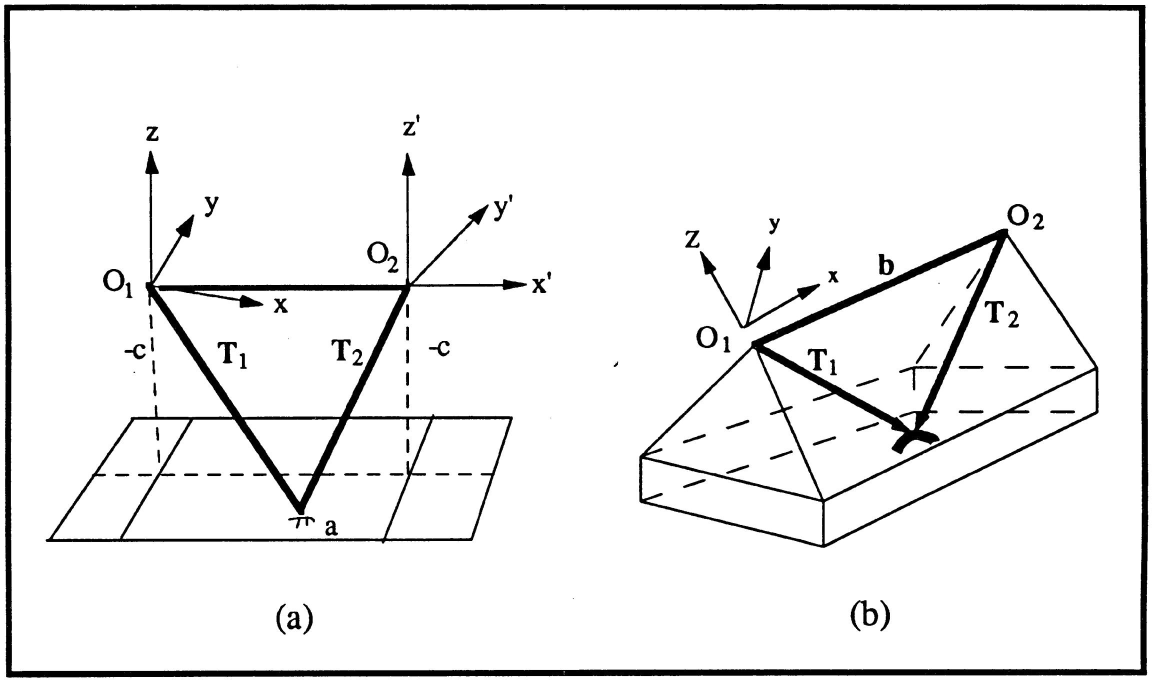
and the constant vector **f** is given by

$$\mathbf{f} = \begin{bmatrix} -\ (s'\mathbf{C}_1 \cdot \Delta\ -\ x_m) \\ -\ (s'\mathbf{C}_2 \cdot \Delta\ -\ y_m) \\ -\ (s'\mathbf{C}_3 \cdot \Delta\ -\ z_m) \end{bmatrix} \tag{3-37e}$$

If sufficient measurements are available, a least squares solution may be obtained. Note again the simplicity of obtaining the elements of the design matrices by simple vector projections.

3.5.3 Coplanarity condition equation

This formulation is based on the geometrical principle that two corresponding rays from an object point plus the camera air-base must lie in the same plane. In other words, the condition often called the coplanarity condition, stipulates that the two corresponding rays emanating from two exposure stations to the same object point must intersect at the location of the object point (see Fig. 3-9)



(a) (b)

Figure 3-9: Coplanarity condition and the stereomodel space

b is the vector connecting two camera stations (i.e. photobase) with the components

$\mathbf{b} = \begin{bmatrix} b_x \\ b_y \\ b_z \end{bmatrix}$ in a chosen Cartesian system (the model space), and $\mathbf{T}_1$ and $\mathbf{T}_2$ represent the two conjugate rays from the image spaces.

To derive the coplanarity equation using the ARDOVS concept, we recognize three spaces involved in the process namely: the left and the right image spaces and the model space.

The coplanarity condition is required to be fulfilled in the model space and so we have to formulate the equation in the model space. Thus, we have an **R**-space as the model space and two **C**-spaces as the left- and the right image spaces. The coplanarity condition requires that we project the image space elements onto the model space.

Applying the ARDOVS relation 1 (eqn. 3-23a) to both the left and the right **C**-spaces (assuming the dependent pair case) we have:

$$\mathbf{p}_a = \begin{bmatrix} x_a \\ y_a \\ -c \end{bmatrix}, \quad \mathbf{p}'_a = \begin{bmatrix} x'_a \\ y'_a \\ -c \end{bmatrix}$$

$$\mathbf{T}_1 = \begin{bmatrix} \mathbf{i} \cdot \mathbf{P}_a \\ \mathbf{j} \cdot \mathbf{P}_a \\ \mathbf{k} \cdot \mathbf{P}_a \end{bmatrix}, \quad \mathbf{T}_2 = \begin{bmatrix} \mathbf{R}_1 \cdot \mathbf{P}'_a \\ \mathbf{R}_2 \cdot \mathbf{P}'_a \\ \mathbf{R}_3 \cdot \mathbf{P}'_a \end{bmatrix} \tag{3-38}$$

Note that because the dependent pair relative orientation is used, the **R**-space is coincident with the left **C**-space, thus, its apparent direction vectors are coincident with the natural direction axes of the left **C**-space. Note also that the coplanarity condition requires only a rotation of each image space ray since they, together with the photobase (or airbase) are to satisfy the coplanarity condition, thus, the ARDOVS relation 1 has been applied without translation.

$\mathbf{P}_a$ = left **C**-space element; $\mathbf{T}_1$ = **R**-space element from rotated left **C**-space

$\mathbf{P}'_a$ = right **C**-space element; $\mathbf{T}_2$ = R-space element from rotated right **C**-space

From the concepts of vector analysis, the coplanarity condition is expressed as a scalar triple product [Eliezer 1963, Coulson 1970]

$$\mathbf{b} \cdot \mathbf{T}_1 \times \mathbf{T}_2 = 0 \tag{3-39}$$

Equation (3-39) is the fundamental mathematical model of the coplanarity condition. This equation is often used to recover the relative orientation elements of two overlapping photographs from measured coordinates of at least 5 points. The derivatives of the coplanarity equation with respect to all the variables involved therein may be obtained by the same standardised set of differential operators stated in equations (3-6) - (3-10) which may be routinely applied. We state here some of the derivatives of (3-39) with respect to the parameters and observables:

$$
\begin{aligned}
\frac{\partial f}{\partial b_y} &= \mathbf{b}_y \cdot \mathbf{T}_1 \times \mathbf{T}_2 \\
\frac{\partial f}{\partial b_z} &= \mathbf{b}_z \cdot \mathbf{T}_1 \times \mathbf{T}_2 \\
\frac{\partial f}{\partial \omega} &= \mathbf{b} \cdot \mathbf{T}_1 \times \mathbf{T}_{2\omega} \\
\frac{\partial f}{\partial \phi} &= \mathbf{b} \cdot \mathbf{T}_1 \times \mathbf{T}_{2\phi} \\
\frac{\partial f}{\partial k} &= \mathbf{b} \cdot \mathbf{T}_1 \times \mathbf{T}_{2k} \\
\frac{\partial f}{\partial x_a} &= \mathbf{b} \cdot \mathbf{T}_{1x} \times \mathbf{T}_2 \\
\frac{\partial f}{\partial y_a} &= \mathbf{b} \cdot \mathbf{T}_{1y} \times \mathbf{T}_2 \\
\frac{\partial f}{\partial x'_a} &= \mathbf{b} \cdot \mathbf{T}_1 \times \mathbf{T}_{2x} \\
\frac{\partial f}{\partial y'_a} &= \mathbf{b} \cdot \mathbf{T}_1 \times \mathbf{T}_{2y}
\end{aligned}
\qquad (3\text{-}40)
$$

where

$$\mathbf{b}_y = \mathbf{j}\,,\ \mathbf{b}_z = \mathbf{k}\,,\ \mathbf{T}_{1x} = \mathbf{i}\,,\ \mathbf{T}_{1y} = \mathbf{j}\,,$$

$$\mathbf{T}_{2x} = \begin{bmatrix} \mathbf{R}_1 \cdot \mathbf{i} \\ 0 \\ 0 \end{bmatrix},\quad \mathbf{T}_{2y} = \begin{bmatrix} 0 \\ \mathbf{R}_2 \cdot \mathbf{j} \\ 0 \end{bmatrix}$$

$$\mathbf{T}_{2\omega} = \begin{bmatrix} \mathbf{R}_{1\omega} \cdot \mathbf{P'}_a \\ \mathbf{R}_{2\omega} \cdot \mathbf{P'}_a \\ \mathbf{R}_{3\omega} \cdot \mathbf{P'}_a \end{bmatrix},\ \mathbf{T}_{2\phi} = \begin{bmatrix} \mathbf{R}_{1\phi} \cdot \mathbf{P'}_a \\ \mathbf{R}_{2\phi} \cdot \mathbf{P'}_a \\ \mathbf{R}_{3\phi} \cdot \mathbf{P'}_a \end{bmatrix},\ \mathbf{T}_{2k} = \begin{bmatrix} \mathbf{R}_{1k} \cdot \mathbf{P'}_a \\ \mathbf{R}_{2k} \cdot \mathbf{P'}_a \\ \mathbf{R}_{3k} \cdot \mathbf{P'}_a \end{bmatrix}$$

For the least squares process, the functional model of the coplanarity condition (3-39) may be represented by

$$\mathbf{f}(\mathbf{X}, \mathbf{L}) = 0$$

where **X** is the vector of unknown relative orientation parameters (i.e. by, bz, ω, ϕ, k), and **L** is the vector of observed quantities i.e. image space conjugate vectors ($x_a, y_a, z_a, x'_a, y'_a, z'_a$). Note that bx is arbitrarily chosen at this stage.

The linearised mathematical model is

$$\mathbf{A}\delta + \mathbf{B}\mathbf{v} - \mathbf{f} = 0$$

where

$$\mathbf{A} = \frac{\partial f(\mathbf{X},\mathbf{L})}{\partial \mathbf{X}}, \quad \mathbf{B} = \frac{\partial f(\mathbf{X},\mathbf{L})}{\partial \mathbf{L}}, \quad \mathbf{f} = -f(\mathbf{X}^{o},\mathbf{L}^{o})$$

all evaluated at the approximate values, and

$$\delta = \mathbf{X} - \mathbf{X}^{o}, \quad \mathbf{v} = \mathbf{L} - \mathbf{L}^{o}$$

The elements of the design matrices **A**, **B**, and of the vector **f**, may be formed as follows:

Matrix A: the elements with respect to the base components (by, bz) are

$$\begin{bmatrix} \mathbf{b}_y \cdot \mathbf{T}_1 \times \mathbf{T}_2 & \mathbf{b}_z \cdot \mathbf{T}_1 \times \mathbf{T}_2 \end{bmatrix} \qquad (3\text{-}41a)$$

and those with respect to the rotation elements (ω, ϕ, k) are

$$\begin{bmatrix} \mathbf{b} \cdot \mathbf{T}_1 \times \mathbf{T}_{2\omega} & \mathbf{b} \cdot \mathbf{T}_1 \times \mathbf{T}_{2\phi} & \mathbf{b} \cdot \mathbf{T}_1 \times \mathbf{T}_{2k} \end{bmatrix} \qquad (3\text{-}41b)$$

The elements of matrix B are

$$\begin{bmatrix} \mathbf{b} \cdot \mathbf{T}_{1x} \times \mathbf{T}_2 & \mathbf{b} \cdot \mathbf{T}_{1y} \times \mathbf{T}_2 & \mathbf{b} \cdot \mathbf{T}_1 \times \mathbf{T}_{2x} & \mathbf{b} \cdot \mathbf{T}_1 \times \mathbf{T}_{2y} \end{bmatrix}$$

(3-41c)

while the constant vector f is obtained as

$$\mathbf{f} = -\,\mathbf{b} \cdot \mathbf{T}_1 \times \mathbf{T}_2 \qquad (3\text{-}42)$$

Since the measured photo coordinates are usually assumed to be uncorrelated, each linearized coplanarity equation may be reduced to the parametric form by dividing it by the square of the second design matrix **B** i.e. $w = (\mathbf{B}\mathbf{B}^T)^{-1}$ which is always regular unless the two rays are near perpendicular to the base vector. In principle, five such equations are sufficient for a unique determination of the parameters, however, more than five points provide a least squares solution.

3.5.4 Recovery of Relative and Absolute Orientation Elements

Having discussed the three basic mathematical models which describe the relationships between the image space and the object space, the model space and the object space, and between the image space and the model space respectively, it is necessary to extract from such models working operators which are needed both to recreate a stereomodel and to transform image measurements into model space position vectors and model space vectors into object space position vectors. However, these operators require both the relative and absolute orientation elements for their operations. We therefore state the formulations for extracting the relative and absolute orientation elements from the three basic models presented above. For this, two approaches are possible. (1) they may be obtained indirectly from the exterior orientation elements (i.e. from the application of 2-D collinearity equations), or (2) directly by using the coplanarity and 3-D collinearity equations.

3.5.4.1 Recovery of Relative and Absolute Orientation Elements from 2-D Collinearity Equations

Using the collinearity equations (3-32), the exterior orientation elements (E.O.) of each photograph may be determined (either in a block adjustment procedure, or in a resection computation). The computational algorithm for the resection application of (3-32) is given later. Regardless of the method used to obtain the E.O., the required R.O. and A.O. elements may be recovered from it. From the E.O. elements for two photos i.e. X_{o1}, Y_{o1}, Z_{o1}, ω_1, ϕ_1, k_1 and X_{o2}, Y_{o2}, Z_{o2}, ω_2, ϕ_2, k_2 we may obtain the R.O. and A.O. elements as follows:

Relative Orientation:

$$\begin{aligned} d\omega &= \omega_2 - \omega_1 \\ d\phi &= \phi_2 - \phi_1 \\ dk &= k_2 - k_1 \end{aligned} \tag{3-43a}$$

$$\Delta = \begin{bmatrix} X_{o2} \\ Y_{o2} \\ Z_{o2} \end{bmatrix} - \begin{bmatrix} X_{o1} \\ Y_{o1} \\ Z_{o1} \end{bmatrix} \tag{3-43b}$$

Using the rotational elements of the E.O. elements of the left hand photograph, we construct the apparent direction vectors of the model space using expressions in (3.25) and then use the ARDOVS relation 2, to perform an R-space to C-space transformation of the airbase (eqn. 3-43b). This gives the photobase as:

$$\begin{bmatrix} b_x \\ b_y \\ b_z \end{bmatrix} = \frac{1}{s} \begin{bmatrix} C_1 \cdot \Delta \\ C_2 \cdot \Delta \\ C_3 \cdot \Delta \end{bmatrix} \tag{3.43c}$$

where s is a chosen fixed scale (the scale may be determined from a known line in both the image space and the object space, or a convenient value may be selected).

Absolute Orientation:

The elements of absolute orientation of the stereomodel resulting from the two photographs is simply the E.O. elements of the left hand photograph, augmented with the chosen fixed scale used in (3-43c). Thus, we may write:

$$\begin{aligned} \omega &= \omega_1 \\ \phi &= \phi_1 \\ k &= k_1 \end{aligned} \tag{3-43d}$$

$$\begin{bmatrix} X_o \\ Y_o \\ Z_o \end{bmatrix} = \begin{bmatrix} X_{o1} \\ Y_{o1} \\ Z_{o1} \end{bmatrix} \tag{3-43e}$$

3.5.4.2 Recovery of Relative and Absolute Orientation Elements from Coplanarity and 3-D Collinearity Equations

These approaches are the direct methods of obtaining the orientation elements.

Relative Orientation:

Using the coplanarity formulation (section 3.5.3), the relative orientation elements of a pair of overlapping photographs may be computed directly from measurements of a minimum of 5 points. The computational algorithm for this exercise is given in section 3.6.

Absolute Orientation:

Given sufficient object space control points and the measured or computed model coordinates of such points, the 3-D collinearity equations (see section 3.5.2) may be used to recover directly the absolute orientation elements of a stereomodel. The computational algorithm is given in section 3.6.

3.5.5 Formulations for Real Time Data Transformations

Having obtained the R.O. and the A.O. elements, we now give the mathematical formulations of the operators for **image-model** and **model-object** space transformations. By image to model transformation we understand the computation of the model coordinates from two conjugate image space measurements. Also, by model to object space transformation, we mean the computation of the object space position for a given model space position vector. Transformation of the first kind is achieved with the R.O. elements, and that of the second kind, with the A.O. elements. In general, transformation of photographic measurements to the object space is achieved by a series of **C**-space to **R**-space projections while the transformation of object space information to the image space is achieved by a series of **R**-space to **C**-space projections.

3.5.5.1 Image to Model Transformation

After the determination of the relative orientation parameters, it is of interest in analytical photogrammetric work to transform image space measurements into the model space for a variety of reasons explained in the next chapter. However, considering the geometric arrangement of the two image spaces and of the resulting model space (Fig. 3-10), there are a number of alternative formulations for computing the model space vector from its corresponding image space vectors. However they are all based on the concept of spatial intersection of corresponding rays and minimum distance between such rays at intersection. Two methods are considered in what follows. The first uses the **C**-space formulation of the collinearity equation in a rigorous computational process to achieve an intersection of conjugate rays while the second uses a **C**-space to **R**-space projection and parallax vector bisection.

3.5.5.1.1 Image Space Intersection

The C-space collinearity condition is enforced for each of the conjugate rays to compute the corresponding model coordinates by requiring an intersection in the image space space (Fig. 3-10).

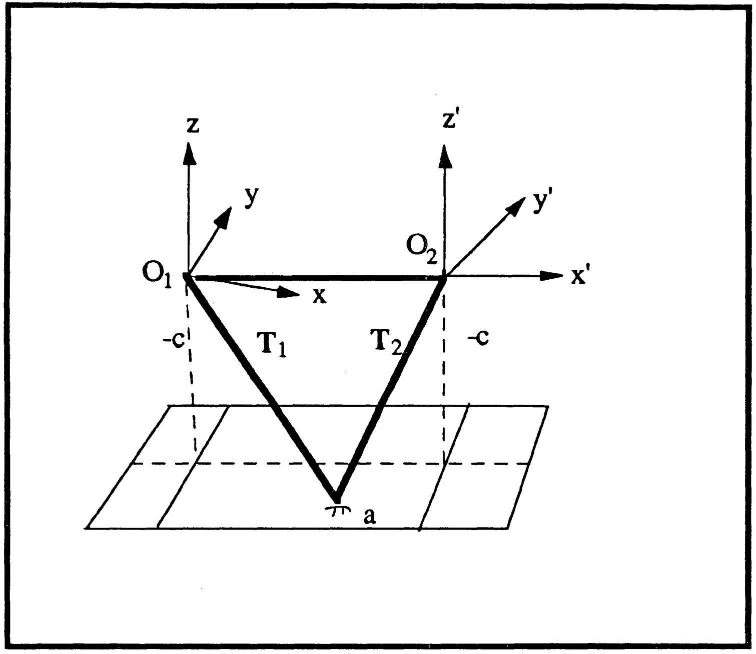


Figure 3-10: Image space intersection

Equation (3-32) is repeated here for convenience. Note that the direction vectors are the C-space apparent base vectors for either the left or the right photograph depending on which ray is being considered. The object space in this case is a model space whose origin may be arbitrarily located.

$$\begin{aligned} k_x C_3 \cdot P_A + C_1 \cdot P_A - k_x C_3 \cdot P_o - C_1 \cdot P_o &= 0 \\ k_y C_3 \cdot P_A + C_2 \cdot P_A - k_y C_3 \cdot P_o - C_2 \cdot P_o &= 0 \end{aligned} \tag{3-44}$$

We specialize 3-44 for the two image spaces as follows:

For the left photo (considering the dependent pair relative orientation):

$$\omega_1 = 0$$
$$\phi_1 = 0$$
$$k_1 = 0$$

$$\mathbf{C}_1 = \mathbf{i} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} , \; \mathbf{C}_2 = \mathbf{j} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} , \mathbf{C}_3 = \mathbf{k} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$\mathbf{P}_A = \mathbf{P}_m = \begin{bmatrix} x_m \\ y_m \\ z_m \end{bmatrix} , \; \mathbf{P}_o = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} , \; k_x = \frac{x_a}{c} \quad k_y = \frac{y_a}{c}$$

Putting these into equation (3-44), it produces the following two equations

$$\begin{aligned} k_x \, \mathbf{k} \cdot \mathbf{P}_m + \mathbf{i} \cdot \mathbf{P}_m &= 0 \\ k_y \, \mathbf{k} \cdot \mathbf{P}_m + \mathbf{j} \cdot \mathbf{P}_m &= 0 \end{aligned} \qquad (3\text{-}45a)$$

Note the simplicity of of the reduction process with the vector approach.

For the right photo we have:

$$\omega_2 = d\omega$$
$$\phi_2 = d\phi$$
$$k_2 = dk$$

then, the right C-space vectors $\mathbf{C}_1$, $\mathbf{C}_2$, and $\mathbf{C}_3$ are constructed using (3-25).

Also let

$$\mathbf{P}_A = \mathbf{P}_m = \begin{bmatrix} x_m \\ y_m \\ z_m \end{bmatrix} , \; \mathbf{P}_o = \mathbf{b} = \begin{bmatrix} b_x \\ b_y \\ b_z \end{bmatrix}$$

$$k_x = k_x' = \frac{x_a'}{c} , \; k_y = k_y' = \frac{y_a'}{c}$$

so that equation (3-44) reduces to two equations for the right image:

$$\begin{aligned} k_x'\mathbf{C}_3\cdot\mathbf{P}_m + \mathbf{C}_1\cdot\mathbf{P}_m - k_x'\mathbf{C}_3\cdot\mathbf{b} - \mathbf{C}_1\cdot\mathbf{b} &= 0 \\ k_y'\mathbf{C}_3\cdot\mathbf{P}_m + \mathbf{C}_2\cdot\mathbf{P}_m - k_y'\mathbf{C}_3\cdot\mathbf{b} - \mathbf{C}_2\cdot\mathbf{b} &= 0 \end{aligned} \qquad (3\text{-}45b)$$

Equations (3-45a, 3-45b), collected together in (3-45c), provide a least squares determination of the model coordinates for the point in question. The approximate model coordinates needed to start the adjustment process may be taken as the left photo coordinates for the point. The solution may be iterated for corrections to initial values.

$$\begin{aligned} k_x \mathbf{k} \cdot \mathbf{P}_m + \mathbf{i} \cdot \mathbf{P}_m &= 0 \\ k_y \mathbf{k} \cdot \mathbf{P}_m + \mathbf{j} \cdot \mathbf{P}_m &= 0 \\ k_x' \mathbf{C}_3 \cdot \mathbf{P}_m + \mathbf{C}_1 \cdot \mathbf{P}_m - k_x' \mathbf{C}_3 \cdot \mathbf{b} - \mathbf{C}_1 \cdot \mathbf{b} &= 0 \qquad \text{(3-45c)} \\ k_y' \mathbf{C}_3 \cdot \mathbf{P}_m + \mathbf{C}_2 \cdot \mathbf{P}_m - k_y' \mathbf{C}_3 \cdot \mathbf{b} - \mathbf{C}_2 \cdot \mathbf{b} &= 0 \end{aligned}$$

The computational algorithm is given in section 3.6.

3.5.5.1.2 Bisection of Parallax Vector

Another formulation for model coordinate computation is the bisection of the parallax vector approach. The method computes the shortest distance between the corresponding rays and computes the model location by adding half of the distance to the first ray. This approach has been described by Schut (1964, 1966, 1973), and Ghosh (1988). However, a simpler approach than those used by these authors is given here. The method uses direction vectors to compute the minimum parallax vector using a quasi least squares approach after Cooper (1987).

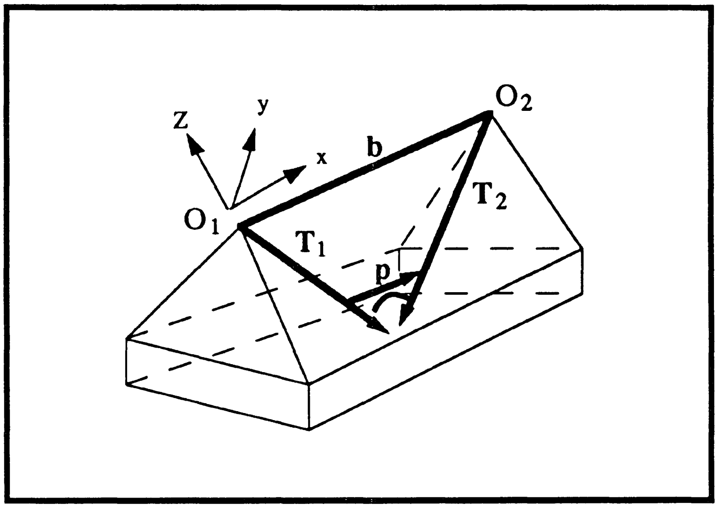


Figure 3-11: Image space intersection and parallax vector

Let **p** be the parallax vector signifying the want of intersection of the two conjugate rays (see Fig. 3-11). Also, let the unit vectors corresponding to $\mathbf{T}_1$, and $\mathbf{T}_2$ be represented by

$\overline{\mathbf{T}}_1$ and $\overline{\mathbf{T}}_2$ respectively, then from the vector polygon in Fig. 3-11 we can write the vector equation

$$\mathbf{p} = \mathbf{b} - s_1\overline{\mathbf{T}}_1 + s_2\overline{\mathbf{T}}_2 \tag{3-46}$$

Now select values for s_1, s_2 such that $\mathbf{p}$ satisfies some suitable condition e.g. minimum length, or that its x_m and the z_m components are zero. For the minimum length condition we have

$$\frac{\partial(\mathbf{p}\cdot\mathbf{p})}{\partial s_1} = 0$$

$$\frac{\partial(\mathbf{p}\cdot\mathbf{p})}{\partial s_2} = 0$$

substitution for $\mathbf{p}$ from (3.46) provides

$$\begin{gathered}\frac{\partial(\mathbf{b} - s_1\overline{\mathbf{T}}_1 + s_2\overline{\mathbf{T}}_2)\cdot(\mathbf{b} - s_1\overline{\mathbf{T}}_1 + s_2\overline{\mathbf{T}}_2)}{\partial s_1} = 0 \\[2ex] \frac{\partial(\mathbf{b} - s_1\overline{\mathbf{T}}_1 + s_2\overline{\mathbf{T}}_2)\cdot(\mathbf{b} - s_1\overline{\mathbf{T}}_1 + s_2\overline{\mathbf{T}}_2)}{\partial s_2} = 0\end{gathered} \tag{3-47}$$

Employing the vector differential operators (eqns. 3-6 - 3-10), we obtain the normal equations from (3-47) as

$$\begin{bmatrix} \overline{\mathbf{T}}_1\cdot\overline{\mathbf{T}}_1 & -\overline{\mathbf{T}}_1\cdot\overline{\mathbf{T}}_2 \\ -\overline{\mathbf{T}}_2\cdot\overline{\mathbf{T}}_1 & \overline{\mathbf{T}}_2\cdot\overline{\mathbf{T}}_2 \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} = \begin{bmatrix} \overline{\mathbf{T}}_1\cdot\mathbf{b} \\ -\overline{\mathbf{T}}_1\cdot\mathbf{b} \end{bmatrix} \tag{3-48}$$

but $\overline{\mathbf{T}}_1$ and $\overline{\mathbf{T}}_2$ are direction vectors, so that (3-48) may be written as

$$\begin{bmatrix} 1 & -\overline{\mathbf{T}}_1\cdot\overline{\mathbf{T}}_2 \\ -\overline{\mathbf{T}}_2\cdot\overline{\mathbf{T}}_1 & 1 \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} = \begin{bmatrix} \overline{\mathbf{T}}_1\cdot\mathbf{b} \\ -\overline{\mathbf{T}}_2\cdot\mathbf{b} \end{bmatrix} \tag{3-49}$$

The determinant of the normal equation (3-49) is given by

$$D = \begin{vmatrix} 1 & -\overline{\mathbf{T}}_1\cdot\overline{\mathbf{T}}_2 \\ -\overline{\mathbf{T}}_2\cdot\overline{\mathbf{T}}_1 & 1 \end{vmatrix} \equiv 1 - (\overline{\mathbf{T}}_1\cdot\overline{\mathbf{T}}_2)^2 \tag{3-50}$$

which is non-zero unless the two rays are parallel, then the solution to (3-49) is

$$\begin{bmatrix} s_1 \\ s_2 \end{bmatrix} = \frac{1}{D} \begin{bmatrix} 1 & \overline{\mathbf{T}}_1 \cdot \overline{\mathbf{T}}_2 \\ \overline{\mathbf{T}}_2 \cdot \overline{\mathbf{T}}_1 & 1 \end{bmatrix} \begin{bmatrix} \overline{\mathbf{T}}_1 \cdot \mathbf{b} \\ -\overline{\mathbf{T}}_2 \cdot \mathbf{b} \end{bmatrix} \tag{3-51}$$

The model coordinate vector is then computed as

$$\mathbf{p}_m = s_1 \overline{\mathbf{T}}_1 + \frac{1}{2} \mathbf{p}$$

or, by substituting (3-46) for **p** and simplifying, we obtain

$$\mathbf{p}_m = \frac{\mathbf{b} + s_1 \overline{\mathbf{T}}_1 + s_2 \overline{\mathbf{T}}_2}{2} \tag{3-52}$$

Equation 3-52 is applied for every point whose image space coordinates are measured.

3.5.5.2 Model Space to Image Space Transformation

This is the inverse of the transformations discussed in section 3.5.5.1. It enables the recovery of the image space vectors that were transformed to the model space. This transformation is often required particularly in connection with image space map updating in which an existing object space feature is required to be superimposed onto the image. These formulations may be derived using ARDOVS relation 2 (eqn. 3-24b), or by manipulating the simplified collinearity equations (3-45c).

Manipulating (3-45c) (assuming the dependent pair relative orientation):

for the left image space we have:

$$k_x = -\frac{\mathbf{i} \cdot \mathbf{P}_m}{\mathbf{k} \cdot \mathbf{P}_m}$$

$$k_y = -\frac{\mathbf{j} \cdot \mathbf{P}_m}{\mathbf{k} \cdot \mathbf{P}_m}$$

$$\mathbf{P}_a = -c \begin{bmatrix} k_x \\ k_y \\ 1 \end{bmatrix} = \begin{bmatrix} x_a \\ y_a \\ -c \end{bmatrix}$$

and for the right image we have

$$k'_x = -\frac{\mathbf{C}_1\cdot(\mathbf{P}_m-\mathbf{b})}{\mathbf{C}_3\cdot(\mathbf{P}_m-\mathbf{b})}$$

$$k'_y = -\frac{\mathbf{C}_2\cdot(\mathbf{P}_m-\mathbf{b})}{\mathbf{C}_3\cdot(\mathbf{P}_m-\mathbf{b})}$$

$$\mathbf{P}'_a = c\begin{bmatrix} k'_x \\ k'_y \\ -1 \end{bmatrix} = \begin{bmatrix} x'_a \\ y'_a \\ -c \end{bmatrix}$$

3.5.5.3 Model Space to Object Space Transformation

The model to object space transformation is often required for real time reduction of selected model points into the object space, for example during digitizing. Since this is a **C**-space to an **R**-space transformation, application of ARDOVS relation 1 provides the object space vector data from the model space vector if the A.O. elements are known.

$$\mathbf{P}_A = \mathbf{P}_o + s\begin{bmatrix} \mathbf{R}_1\cdot\mathbf{P}_m \\ \mathbf{R}_2\cdot\mathbf{P}_m \\ \mathbf{R}_3\cdot\mathbf{P}_m \end{bmatrix} \qquad (3\text{-}53)$$

where $s = 1/s'$ and R_1, R_2, R_3 are the apparent direction vectors of the object space computed using the rotation angles of absolute orientation. Note that the absolute orientation parameters used in this equation may be obtained directly as described in section 3.5.4.2, or indirectly from the E.O. parameters obtained by resection (section 3.5.4.1). Remember in the latter case, the A.O. elements are the E.O. elements of the left hand photograph of the stereopair of photographs being used. The row vectors of the rotation matrix are computed using equation 3-27. Equation 3-53 is applied to each model point.

3.5.5.4 Object Space to Model SpaceTransformation

The object to model space transformation is required in real time data processing in the analytical plotter when an object space information is to be overlaid on the corresponding image either for checking of digitizing accuracy or for map updating. Since this is an **R**-space to a **C**-space transformation, application of ARDOVS relation 2 provides the model space vector data from the object space vector if the A.O. elements are known.

$$\mathbf{P}_m = \frac{1}{s}\begin{bmatrix} \mathbf{C}_1\cdot(\mathbf{P}_A-\mathbf{P}_o) \\ \mathbf{C}_2\cdot(\mathbf{P}_A-\mathbf{P}_o) \\ \mathbf{C}_3\cdot(\mathbf{P}_A-\mathbf{P}_o) \end{bmatrix} \tag{3-54}$$

where $\mathbf{P}_m$ is an element in the **C**-space and $\mathbf{p}_A$ an element of the **R**-space; $\mathbf{C}_1,\mathbf{C}_2,\mathbf{C}_3$ are the apparent direction vectors of the **C**-space computed using the rotation angles of absolute orientation.

3.6 Computational Algorithms for Photogrammetric Operations

This section presents the computational procedure by which the mathematical models described above are utilized to achieve the objectives for which they were established. These algorithms are some of those already implemented in the workstation designs which will be discussed in chapter 4.

3.6.1 Computational Algorithms for Applications of 2-D Collinearity Equations

Two computational algorithms are considered here for two applications of the 2-D collinearity equations described in section 3.5.1. The first algorithm concerns application for resection while the second is for object space determination by spatial intersection of bundles. The algorithm for application of this functional model to control extension by block adjustment is a generalization of the ones given here extended to multiple photographs, taking into consideration the special circumstances of working with few control points.

3.6.1.1 Single-Photo Resection Algorithm.

In this application, the parameters of transformation (E.O. elements) are to be determined. It is assumed that sufficient object space control points are available. The algorithm is based on the following mathematical model:

$$(k_xC_3 + C_1)\cdot\mathbf{P}_A - (k_xC_3 + C_1)\cdot\mathbf{P}_o = 0$$

$$(k_yC_3 + C_2)\cdot\mathbf{P}_A - (k_yC_3 + C_2)\cdot\mathbf{P}_o = 0$$

and performs the following steps:

1. obtain initial approximations for the camera parameters.
2. construct the column vectors of the rotation matrix and their derivatives using the expressions in eqns. 3-25, 3-26
3. for each image point measured, compute:

$$k_x = \frac{x_a}{c},\quad k_y = \frac{y_a}{c}$$

$$H_x = k_xC_3 + C_1$$

$$H_y = k_yC_3 + C_2$$

and their gradient vectors with respect to the orientation elements (note that all evaluation is made at the current values of the unknown parameters)

$$H_{x\omega} = k_xC_{3\omega} + C_{1\omega}$$

$$H_{x\phi} = k_xC_{3\phi} + C_{1\phi}$$

$$H_{xk} = k_xC_{3k} + C_{1k}$$

$$H_{y\omega} = k_yC_{3\omega} + C_{2\omega}$$

$$H_{y\phi} = k_yC_{3\phi} + C_{2\phi}$$

$$H_{yk} = k_yC_{3k} + C_{2k}$$

$$\mathbf{P}_A = \begin{bmatrix} X_A \\ Y_A \\ Z_A \end{bmatrix} \quad \text{and } \mathbf{P}_o = \begin{bmatrix} X_o \\ Y_o \\ Z_o \end{bmatrix}$$

$$\Delta = \mathbf{P}_A - \mathbf{P}_o$$

4. form the design matrices for this point.

   The first design matrix (adjusted for the parametric case) with respect to orientation elements (ω,ϕ,k)

$$A_1 = \frac{c}{C_3 \cdot \Delta} \begin{bmatrix} \mathbf{H}_{x\omega} \cdot \Delta & \mathbf{H}_{x\phi} \cdot \Delta & \mathbf{H}_{xk} \cdot \Delta \\ \mathbf{H}_{y\omega} \cdot \Delta & \mathbf{H}_{y\phi} \cdot \Delta & \mathbf{H}_{yk} \cdot \Delta \end{bmatrix}$$

   with respect to the camera position (X_o, Y_o, Z_o)

$$A_2 = \frac{c}{C_3 \cdot \Delta} \begin{bmatrix} -\mathbf{H}_x \cdot \mathbf{i} & -\mathbf{H}_x \cdot \mathbf{j} & -\mathbf{H}_x \cdot \mathbf{k} \\ -\mathbf{H}_y \cdot \mathbf{i} & -\mathbf{H}_y \cdot \mathbf{j} & -\mathbf{H}_y \cdot \mathbf{k} \end{bmatrix}$$

   and the constant vector **f**:

$$\mathbf{f} = -\frac{c}{C_3 \cdot \Delta} \begin{bmatrix} \mathbf{H}_x \cdot \Delta \\ \mathbf{H}_y \cdot \Delta \end{bmatrix}$$

5. Aggregate the contribution of this point as

$$A_i = [\, A_1 \quad A_2 \,] \ , \ \mathbf{f}_i = \mathbf{f}$$

6. repeat steps 3-5 for all points measured on this photograph whose object position is given.
7. apply least squares procedure and test for convergence.
8. repeat steps 2-7 until convergence.

3.6.1.2 Algorithm for Space intersection.

In this application, the parameters of transformation (camera information) for two cameras are given and the object space coordinates of a point are needed. The algorithm uses the mathematical model:

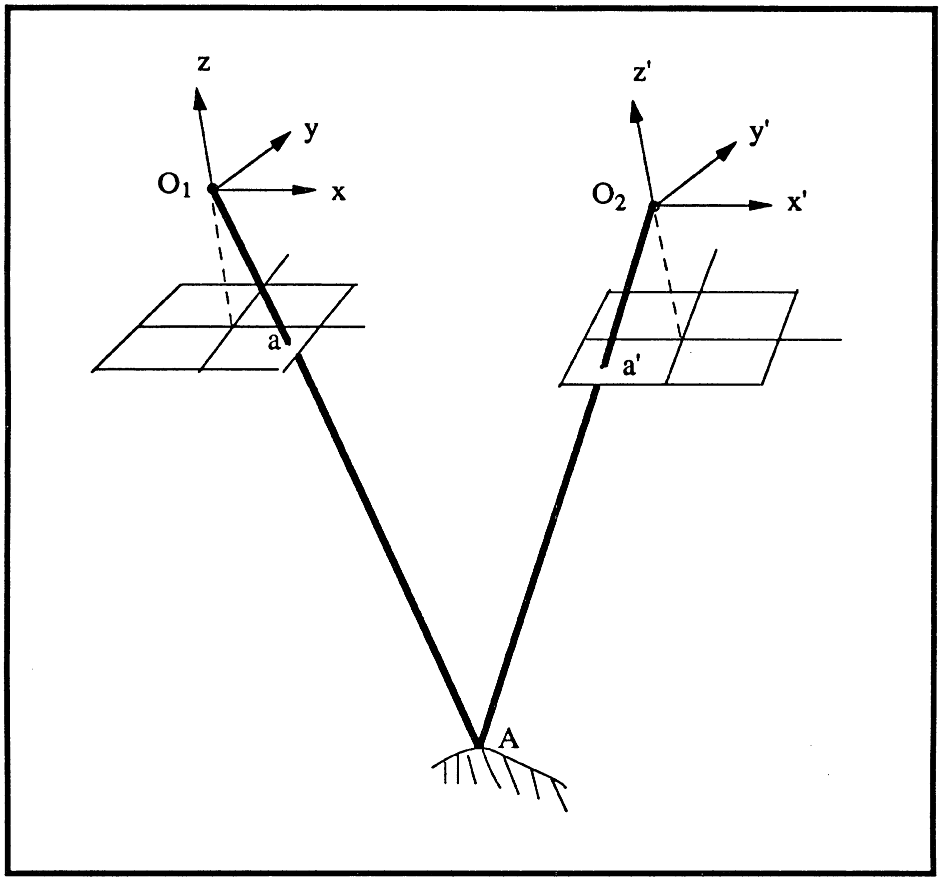


Figure 3-12: space intersection

$$(k_x C_3 + C_1) \cdot \mathbf{P}_A - (k_x C_3 + C_1) \cdot \mathbf{P}_o = 0$$

$$(k_y C_3 + C_2) \cdot \mathbf{P}_A - (k_y C_3 + C_2) \cdot \mathbf{P}_o = 0$$

and performs the following steps:

1. obtain approximate coordinates for this point.
2. construct the design matrix with respect to the object space coordinates (X_A, Y_A, Z_A) using the E.O. elements and the measurements for the left photograph, compute:

   (C_1, C_2, C_3), apparent direction of the left C-space

$k_x = \frac{x_a}{c}, \quad k_y = \frac{y_a}{c}$

$H_x = k_x C_3 + C_1$

$H_y = k_y C_3 + C_2$

$$P_A = \begin{bmatrix} X_A \\ Y_A \\ Z_A \end{bmatrix} \quad \text{and} \quad P_o = \begin{bmatrix} X_o \\ Y_o \\ Z_o \end{bmatrix}$$

$\Delta = P_A - P_o$

$$A_1 = \frac{c}{C_3 \cdot \Delta} \begin{bmatrix} H_x \cdot i & H_x \cdot j & H_x \cdot k \\ H_y \cdot i & H_y \cdot j & H_y \cdot k \end{bmatrix}$$

$$f_1 = -\frac{c}{C_3 \cdot \Delta} \begin{bmatrix} H_x \cdot \Delta \\ H_y \cdot \Delta \end{bmatrix}$$

and using the E.O. elements and the measurements for the right photograph, compute:

(C_1, C_2, C_3), apparent direction vectors of the right C-space

$k_x = \frac{x_a}{c}, \quad k_y = \frac{y_a}{c}$

$H_x = k_x C_3 + C_1$

$H_y = k_y C_3 + C_2$

$$A_2 = \frac{c}{C_3 \cdot \Delta} \begin{bmatrix} H_x \cdot i & H_x \cdot j & H_x \cdot k \\ H_y \cdot i & H_y \cdot j & H_y \cdot k \end{bmatrix}$$

$$f_2 = -\frac{c}{C_3 \cdot \Delta} \begin{bmatrix} H_x \cdot \Delta \\ H_y \cdot \Delta \end{bmatrix}$$

3. collect the normal equations together

$$A = \begin{bmatrix} A_1 \\ A_2 \end{bmatrix}, \quad f = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}$$

4. apply least squares procedure and test for convergence.
5. repeat steps 3-4 until convergence.

These algorithms have been used for various computational tasks in the software packages developed for this study.

3.6.2 Algorithms for Recovery of Relative and Absolute Orientation Elements

In this section, we list the algorithms for the direct recovery of the orientation elements as discussed in section 3.5.4. The first is the recovery of R.O. elements using the coplanarity condition and the second recovers the A.O. elements from the 3-D collinearity equations.

3.6.2.1 Algorithm for Recovery of R.O. by dependent pair relative orientation

The mathematical model (eqn. 3-46)

$$\mathbf{b} \cdot \mathbf{T}_1 \times \mathbf{T}_2 = 0$$

provides the basis for this algorithm, which performs the following steps:

1. obtain initial approximations to the relative orientation elements ($b_x, b_y, b_z, \omega, \phi, k$); b_x is usually fixed at a chosen value and determines the scale of the model space. For the first iteration, all five relative orientation parameters may be taken as zero.
2. construct the R-space direction vectors and their derivatives using equations 3-27, 3-28; then compute:

$$\mathbf{P}_a = \begin{bmatrix} x_a \\ y_a \\ -c \end{bmatrix}, \quad \mathbf{P'}_a = \begin{bmatrix} x'_a \\ y'_a \\ -c \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} b_x \\ b_y \\ b_z \end{bmatrix}$$

$$T_1 = \begin{bmatrix} \mathbf{i} \cdot \mathbf{P}_a \\ \mathbf{j} \cdot \mathbf{P}_a \\ \mathbf{k} \cdot \mathbf{P}_a \end{bmatrix} , \quad T_2 = \begin{bmatrix} \mathbf{R}_1 \cdot \mathbf{P}'_a \\ \mathbf{R}_2 \cdot \mathbf{P}'_a \\ \mathbf{R}_3 \cdot \mathbf{P}'_a \end{bmatrix}$$

$$\mathbf{b}_y = \mathbf{j} , \ \mathbf{b}_z = \mathbf{k} , \ \mathbf{T}_{1x} = \mathbf{i} , \ \mathbf{T}_{1y} = \mathbf{j}$$

$$\mathbf{T}_{2x} = \begin{bmatrix} \mathbf{R}_1 \cdot \mathbf{i} \\ 0 \\ 0 \end{bmatrix} , \ \mathbf{T}_{2y} = \begin{bmatrix} 0 \\ \mathbf{R}_2 \cdot \mathbf{j} \\ 0 \end{bmatrix}$$

$$\mathbf{T}_{2\omega} = \begin{bmatrix} \mathbf{R}_{1\omega} \cdot \mathbf{P}'_a \\ \mathbf{R}_{2\omega} \cdot \mathbf{P}'_a \\ \mathbf{R}_{3\omega} \cdot \mathbf{P}'_a \end{bmatrix} , \ \mathbf{T}_{2\phi} = \begin{bmatrix} \mathbf{R}_{1\phi} \cdot \mathbf{P}'_a \\ \mathbf{R}_{2\phi} \cdot \mathbf{P}'_a \\ \mathbf{R}_{3\phi} \cdot \mathbf{P}'_a \end{bmatrix} , \ \mathbf{T}_{2k} = \begin{bmatrix} \mathbf{R}_{1k} \cdot \mathbf{P}'_a \\ \mathbf{R}_{2k} \cdot \mathbf{P}'_a \\ \mathbf{R}_{3k} \cdot \mathbf{P}'_a \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} \mathbf{b} \cdot \mathbf{T}_{1x} \times \mathbf{T}_2 & \mathbf{b} \cdot \mathbf{T}_{1y} \times \mathbf{T}_2 & \mathbf{b} \cdot \mathbf{T}_1 \times \mathbf{T}_{2x} & \mathbf{b} \cdot \mathbf{T}_1 \times \mathbf{T}_{2y} \end{bmatrix}$$

is used to compute the transform factor:

$$w = (BB^T)^{-1}$$

For matrix A: the elements with respect to the base components (b_y, b_z) are

$$\mathbf{A}_1 = w \begin{bmatrix} \mathbf{b}_y \cdot \mathbf{T}_1 \times \mathbf{T}_2 & \mathbf{b}_z \cdot \mathbf{T}_1 \times \mathbf{T}_2 \end{bmatrix}$$

and those with respect to the rotation elemements (ω,ϕ,k) are

$$\mathbf{A}_2 = w \begin{bmatrix} \mathbf{b} \cdot \mathbf{T}_1 \times \mathbf{T}_{2\omega} & \mathbf{b} \cdot \mathbf{T}_1 \times \mathbf{T}_{2\phi} & \mathbf{b} \cdot \mathbf{T}_1 \times \mathbf{T}_{2k} \end{bmatrix}$$

while the constant vector f is obtained as

$$\mathbf{f} = - \, w \ \mathbf{b} \cdot \mathbf{T}_1 \times \mathbf{T}_2$$

The contribution for this point to the normal equation is:

$$\mathbf{A}_i = [\, \mathbf{A}_1 \quad \mathbf{A}_2 \,] \ , \ \mathbf{f}_i = \mathbf{f}$$

3. repeat steps 1 and 2 for all points and assemble equations
4. apply least squares procedure and test for convergence.

5. repeat steps 1-4 until convergence.

3.6.2.2 Algorithm for recovery of A.O. elements from 3-D collinearity equations

The computational algorithms may be designed according to the application to which the mathematical model is put, i.e whether the equation is being applied to single model absolute orientation, or to the adjustment of blocks of stereo models. Given here is the scheme for single model absolute orientation. In this application, the parameters of transformation (absolute orientation elements) are to be determined. The algorithm is based on the following mathematical model:

$$s'C_1 \cdot \mathbf{P}_A - s'C_1 \cdot \mathbf{P}_o - x_m = 0$$

$$s'C_2 \cdot \mathbf{P}_A - s'C_2 \cdot \mathbf{P}_o - y_m = 0$$

$$s'C_3 \cdot \mathbf{P}_A - s'C_3 \cdot \mathbf{P}_o - z_m = 0$$

and provides the following steps:

1. obtain initial approximations for the parameters.
2. the column vectors of the rotation matrix and their gradients are obtained by evaluating the expressions in (3-25, 3-26)
3. for each model point, compute:

   design matrix with respect to orientation elements and scale (ω,ϕ,k, s') in that order

$$\mathbf{P}_A = \begin{bmatrix} X_A \\ Y_A \\ Z_A \end{bmatrix} \quad \text{and } \mathbf{P}_o = \begin{bmatrix} X_o \\ Y_o \\ Z_o \end{bmatrix}$$

$$\Delta = \mathbf{P}_A - \mathbf{P}_o$$

$$A_1 = \begin{bmatrix} s'C_{1\omega} \cdot \Delta & s'C_{1\phi} \cdot \Delta & s'C_{1k} \cdot \Delta & C_1 \cdot \Delta \\ s'C_{2\omega} \cdot \Delta & s'C_{2\phi} \cdot \Delta & s'C_{2k} \cdot \Delta & C_2 \cdot \Delta \\ s'C_{3\omega} \cdot \Delta & s'C_{3\phi} \cdot \Delta & s'C_{3k} \cdot \Delta & C_3 \cdot \Delta \end{bmatrix}$$

and with respect to the object space position of the origin of the model system (X_o, Y_o, Z_o)

$$A_2 = \begin{bmatrix} -s'C_1 \cdot \mathbf{i} & -s'C_1 \cdot \mathbf{j} & -s'C_1 \cdot \mathbf{k} \\ -s'C_2 \cdot \mathbf{i} & -s'C_2 \cdot \mathbf{j} & -s'C_2 \cdot \mathbf{k} \\ -s'C_3 \cdot \mathbf{i} & -s'C_3 \cdot \mathbf{j} & -s'C_3 \cdot \mathbf{k} \end{bmatrix}$$

The constant vector **f** is given by

$$\mathbf{f} = \begin{bmatrix} -(s'C_1 \cdot \Delta - x_m) \\ -(s'C_2 \cdot \Delta - y_m) \\ -(s'C_3 \cdot \Delta - z_m) \end{bmatrix}$$

4. Aggregate the contribution of this point as

   $$\mathbf{A}_i = [\ \mathbf{A}_1 \quad \mathbf{A}_2\ ]\ ,\ \mathbf{f}_i = \mathbf{f}$$

5. repeat this for all points measured in this model whose object position is given.
6. apply least squares procedure and test for convergence.
7. repeat steps 1-6 until convergence.

3.6.3 Algorithms for image space to model space transformation

The algorithms given in this section are the functional operators which are used in real time mode to move position vectors from the image spaces to the model space as described in section 3.5.5.1.

3.6.3.1 Image space intersection

The computational algorithm is summarized as follows.

The four collinearity equations are (eqn. 3-45c):

$$k_x\, \mathbf{k} \cdot \mathbf{P}_m + \mathbf{i} \cdot \mathbf{P}_m = 0$$

$$k_y \mathbf{k} \cdot \mathbf{P}_m + \mathbf{j} \cdot \mathbf{P}_m = 0$$

$$k'_x \mathbf{C}_3 \cdot \mathbf{P}_m + \mathbf{C}_1 \cdot \mathbf{P}_m - k'_x \mathbf{C}_3 \cdot \mathbf{b} - \mathbf{C}_1 \cdot \mathbf{b} = 0$$

$$k'_y \mathbf{C}_3 \cdot \mathbf{P}_m + \mathbf{C}_2 \cdot \mathbf{P}_m - k'_y \mathbf{C}_3 \cdot \mathbf{b} - \mathbf{C}_2 \cdot \mathbf{b} = 0$$

Based on these, the following steps are carried out:

1. construct the apparent directions of the right C-space from the relative orientation parameters, i.e. ($\mathbf{C}_1$, $\mathbf{C}_2$, $\mathbf{C}_3$). The formulas are the same as used in section 3.6.1

$$k_x = \frac{x_a}{c}, \quad k_y = \frac{y_a}{c}$$

$$k'_x = \frac{x'_a}{c}, \quad k'_y = \frac{y'_a}{c}$$

2. compute the design matrices of (3-45c) with respect to the model coordinates (X_m,Y_m,Z_m) by applying vector differential operators and noting that

$$\frac{\partial \mathbf{P}_m}{\partial x_m} = \mathbf{i}, \quad \frac{\partial \mathbf{P}_m}{\partial y_m} = \mathbf{j}, \quad \frac{\partial \mathbf{P}_m}{\partial z_m} = \mathbf{k}$$

and

$$\mathbf{i} \cdot \mathbf{j} = \mathbf{i} \cdot \mathbf{k} = \mathbf{j} \cdot \mathbf{k} = 0$$

we have:

$$\mathbf{A}_1 = \frac{c}{\mathbf{k} \cdot \mathbf{P}_m} \begin{bmatrix} \mathbf{i} \cdot \mathbf{i} & 0 & k_x \mathbf{k} \cdot \mathbf{k} \\ 0 & \mathbf{j} \cdot \mathbf{j} & k_y \mathbf{k} \cdot \mathbf{k} \end{bmatrix}$$

$$\mathbf{f}_1 = - \frac{c}{\mathbf{k} \cdot \mathbf{P}_m} \begin{bmatrix} k_x \mathbf{k} \cdot \mathbf{P}_m + \mathbf{i} \cdot \mathbf{P}_m \\ k_y \mathbf{k} \cdot \mathbf{P}_m + \mathbf{j} \cdot \mathbf{P}_m \end{bmatrix}$$

$$\mathbf{H}'_x = k'_x \mathbf{C}_3 + \mathbf{C}_1, \quad \mathbf{H}'_y = k'_y \mathbf{C}_3 + \mathbf{C}_2$$

$$\mathbf{A}_2 = \frac{c}{\mathbf{C}_3 \cdot \Delta} \begin{bmatrix} \mathbf{H}'_x \cdot \mathbf{i} & \mathbf{H}'_x \cdot \mathbf{j} & \mathbf{H}'_x \cdot \mathbf{k} \\ \mathbf{H}'_y \cdot \mathbf{i} & \mathbf{H}'_y \cdot \mathbf{j} & \mathbf{H}'_y \cdot \mathbf{k} \end{bmatrix}$$

$$f_2 = -\frac{c}{C_3 \cdot \Delta}\begin{bmatrix} H'_x \cdot p_m - H'_x \cdot b \\ H'_y \cdot p_m - H'_y \cdot b \end{bmatrix}$$

so that for the two rays we have

$$A = \begin{bmatrix} A_1 \\ A_2 \end{bmatrix}, \quad f = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}$$

3. apply least squares procedure

$$P_m = P_m + (A^T A)^{-1} A^T f$$

4. repeat steps 1-3 if correction is large.

3.6.3.2 Bisection of Parallax Vector Algorithm.

Given the conjugate image space vectors of an image point, the task is to compute the equivalent model coordinates. The algorithm given is for the dependent pair relative orientation.

1. construct the apparent direction of the R-space using the parameters of relative orientation and compute the following:

$$p_a = \begin{bmatrix} x_a \\ y_a \\ -c \end{bmatrix}, \quad p'_a = \begin{bmatrix} x'_a \\ y'_a \\ -c \end{bmatrix}, \quad b = \begin{bmatrix} b_x \\ b_y \\ b_z \end{bmatrix}$$

$$T_1 = \begin{bmatrix} i \cdot P_a \\ j \cdot P_a \\ k \cdot P_a \end{bmatrix}, \quad T_2 = \begin{bmatrix} R_1 \cdot P'_a \\ R_2 \cdot P'_a \\ R_3 \cdot P'_a \end{bmatrix}$$

$$\overline{T}_1 = \frac{T_1}{|T_1|}$$

$$\overline{T}_2 = \frac{T_2}{|T_2|}$$

$$D = 1 - (\overline{\mathbf{T}}_1 \cdot \overline{\mathbf{T}}_2)^2$$

$$\begin{bmatrix} s_1 \\ s_2 \end{bmatrix} = \frac{1}{D} \begin{bmatrix} 1 & \overline{\mathbf{T}}_1 \cdot \overline{\mathbf{T}}_2 \\ \overline{\mathbf{T}}_2 \cdot \overline{\mathbf{T}}_1 & 1 \end{bmatrix} \begin{bmatrix} \overline{\mathbf{T}}_1 \cdot \mathbf{b} \\ -\overline{\mathbf{T}}_2 \cdot \mathbf{b} \end{bmatrix}$$

2. The model coordinate vector is then computed as

$$\mathbf{p}_m = \frac{\mathbf{b} + s_1\overline{\mathbf{T}}_1 + s_2\overline{\mathbf{T}}_2}{2}$$

3. repeat steps 1-2 for all points.

3.6.4 Model space to object space transformation

The computation of the object space position of a given model space vector is obtained by applying equation (3.53):

$$\mathbf{P}_A = \mathbf{P}_o + s \begin{bmatrix} \mathbf{R}_1 \cdot \mathbf{p}_m \\ \mathbf{R}_2 \cdot \mathbf{p}_m \\ \mathbf{R}_3 \cdot \mathbf{p}_m \end{bmatrix}$$

3.6.5 Object space to Model space transformation

The computation of the model space position of a given object space vector is obtained by applying equation (3.54):

$$\mathbf{P}_m = \frac{1}{s} \begin{bmatrix} \mathbf{C}_1 \cdot (\mathbf{P}_A - \mathbf{P}_o) \\ \mathbf{C}_2 \cdot (\mathbf{P}_A - \mathbf{P}_o) \\ \mathbf{C}_3 \cdot (\mathbf{P}_A - \mathbf{P}_o) \end{bmatrix}$$

3.7 Summary

This chapter has presented the various mathematical formulations for handling vector elements in an analytical photogrammetric system. The vector approach, in the author's experience, provides substantial savings in the thought process and an improvement in the understanding of routinely applied techniques. The use of the **C**-space, the **R**-space and the ARDOVS methodology ensures the precise application of appropriate formulations for specified tasks and removes ambiguities both in the interpretation and further applications of the results.

In all the presentations given in this chapter, references have been made to image space and model space vectors. It has been assumed that these quantities are always given or are measurable using some instrument and some procedure. In the practice of analytical photogrammetry, the acquisition of the image space vectors is achieved by using the analytical plotter in comparator mode, while the stereo plotter mode is used to acquire the model space vectors. Moreover, of significance to the process are the elements of interior orientation of the image space. As has been noted earlier, these elements are usually determined by calibration. Also, the departure of the photogrammetric ray from an ideal straight path due to refraction has to be taken into account. However, of particular interest to this study is the selection and combination of procedures for the basic tasks such as mensuration (process of obtaining the image space or the model space vectors), triangulation (for determination of both camera orientation elements and object space control points for a block of photographs covering large areas), stereomodel definition (separating a block of models into its constituent stereomodels), stereomodel restoration (reinstallation of a pair of photographs on the instrument stages ready for digital data collection), and the digitizing operation, in analytical systems. For each of these tasks many approaches or options are possible which leads to the many system configurations that may be implemented for any system. The selection of the optimum software design for a photogrammetric workstation is the main issue addressed in the next chapter.

To conclude this chapter, we present three vector data processors which are prototype implementations of the real time subsystem of an analytical photogrammetric workstation. These designs demonstrate the tremendous power of the vector space approach and the simplification introduced by the ARDOVS methodology. These real time vector processors are named according to the method of image-to-model transformation used by the system. In the three designs, a spatial data base and a graphic screen are shown as part of the system to represent the object space. Furthermore, the image space could be digital or analogue. In each system, a **C**-space (image) to an **R**-space (model) followed by a **C**-space (model) to an **R**-space (object) takes an image point to the data base. An **R**-space (object) to a **C**-space (model) followed by an **R**-space (model) to a **C**-space (image) takes a point from the data base to the image.

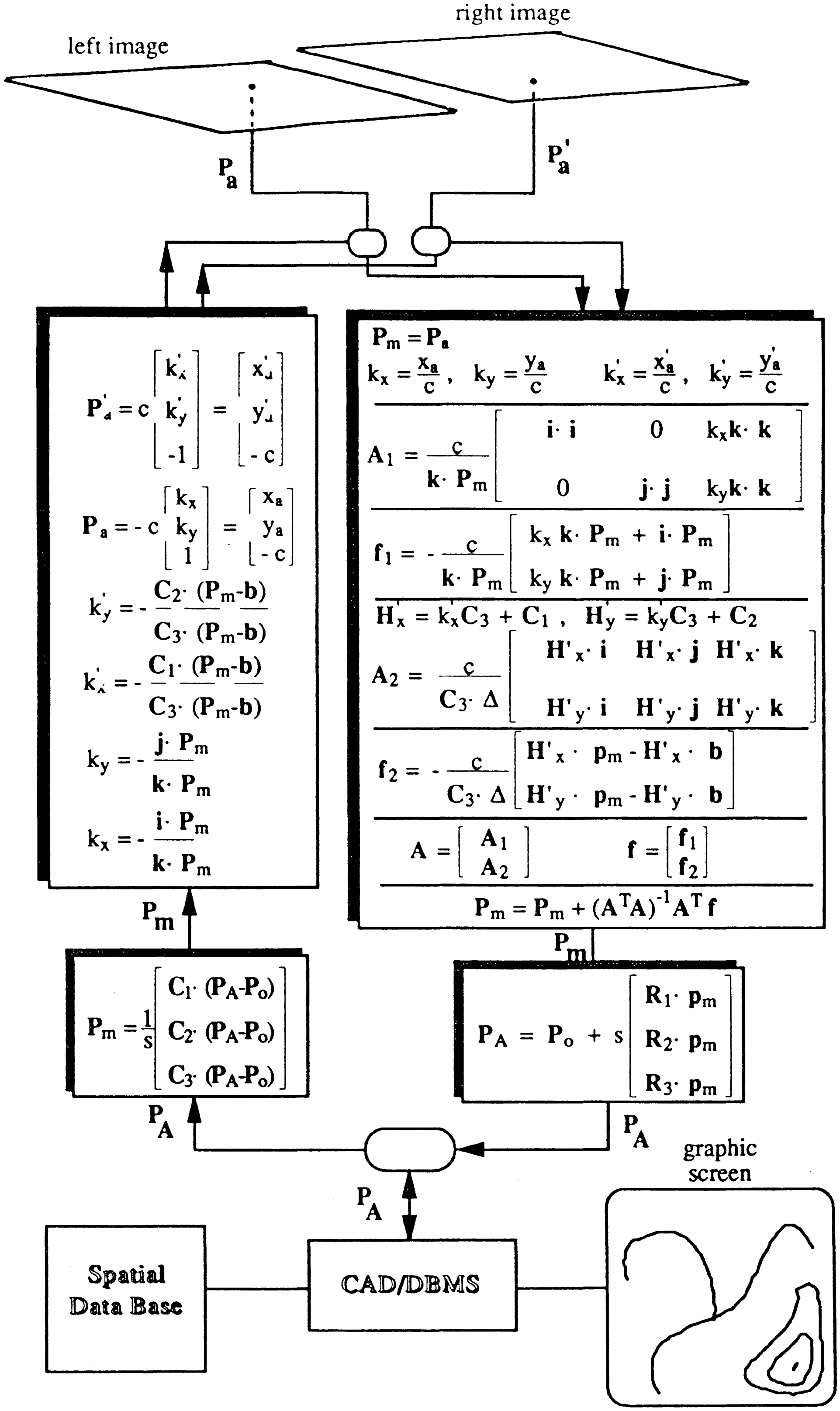


Figure 3-13: Real Time Data Processor Based on Image Space Intersection

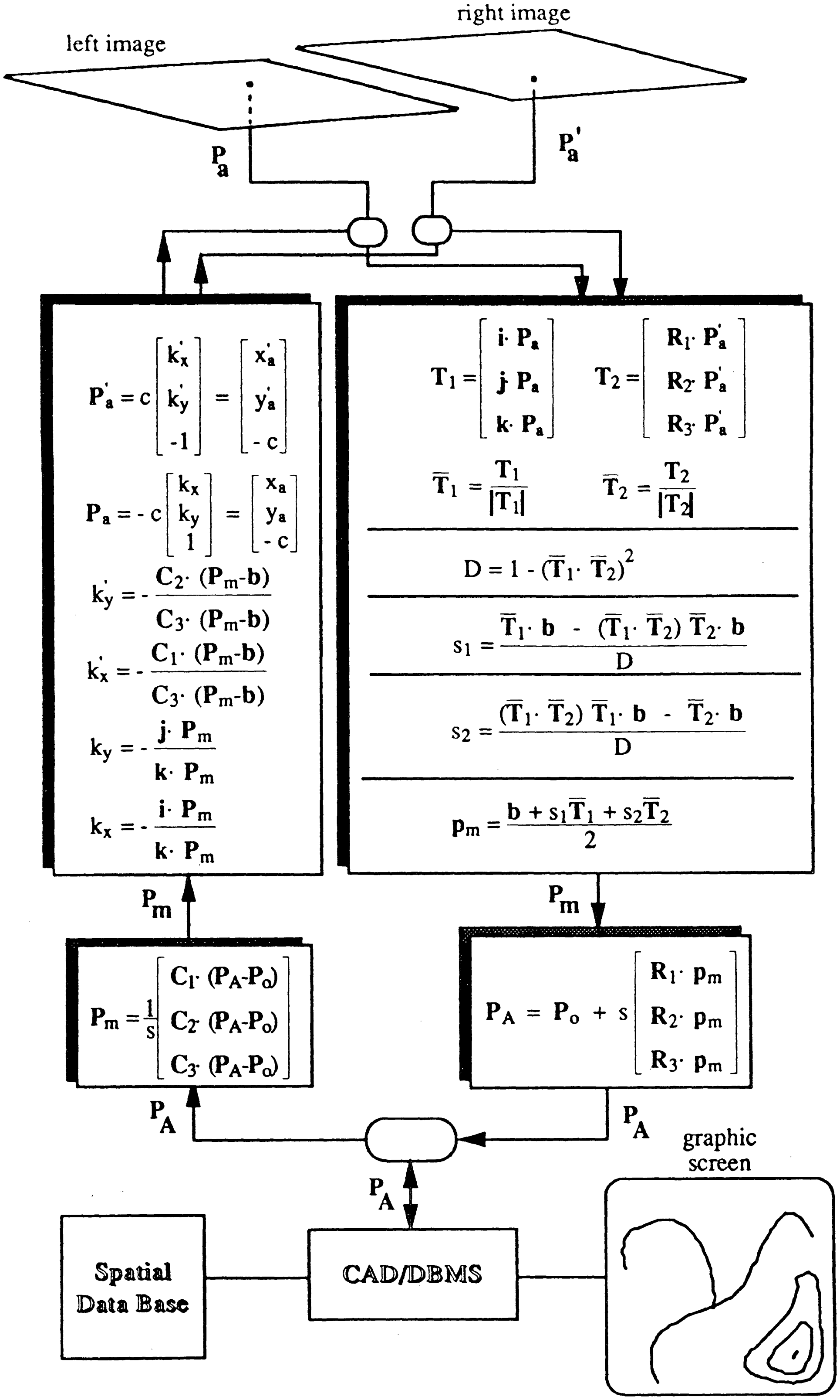


Figure 3-14: Real Time Data Processor Based on Parallax Vector Bisection

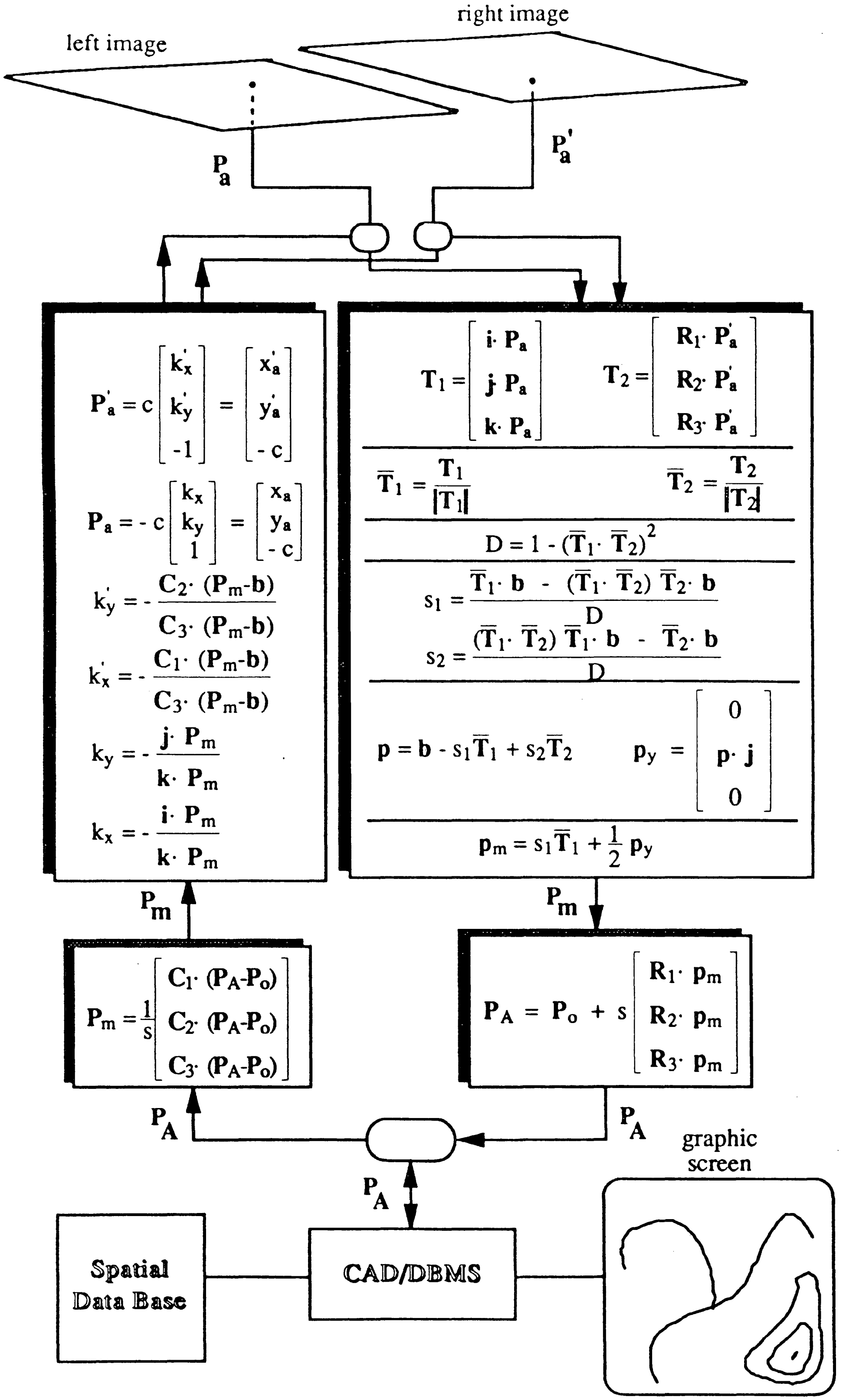


Figure 3-15: Real Time Data Processor Based on Y-Parallax Averaging

Chapter 4

The Optimum Software Configuration for a Photogrammetric Workstation

In this chapter, the various options available for the basic photogrammetric tasks are explored. Different operational configurations are developed through an interplay of the options for the basic tasks; and these make up software design alternatives for the workstation. Automation content, speed of operation, production cost, and achievable accuracy of the derived spatial information are the criteria used as constraints in a detailed analysis employing linear mathematical programming to select the optimum system design. The chapter starts off in section 4.1 with the discussion of the workstation concept and the available methodologies for accomplishing its basic tasks; section 4.2 explores the different system architectures resulting from an interplay of the options considered in section 4.1, while in section 4.3, a brief concept of optimization by mathematical programming is given together with the derivation of an empirical cost function and also the constraint functions needed for the selection of the optimum design. In section 4.4, the strategy for estimating the cost (hardware + software) of each configuration, as well as the human labour involved are discussed; section 4.5 describes the practical test conducted to evaluate the coefficients of the performance and the constraint functions derived in section 4.3, and section 4.6 presents the analysis of the data and concludes the chapter with the selection of the optimum design.

4.1 The Workstation Concept and Options for the Basic Tasks

A fully equipped analytical photogrammetric system may be considered as a workstation dedicated to the acquisition of vector data from photographic images (Figure 4.1a). In such a system two groups of components may be identified: (1) Real time modules and (2) Off-line (or non-real time) modules. The real time components are data processors which achieve the collection of the vector data and maintain a link with the digital Cartographic unit (see Figs. 3-13 - 3-15). These components include analogue to digital conversion by manual digitizing or raster to vector conversion using correlators. Also included are vector processors which convert image space measurements into object space vectors in real time. The off-line components on the other hand, perform tasks required to reduce a block of photographs to digitizable form and supply the necessary parameters to the real time operators (Figure 4.1a).

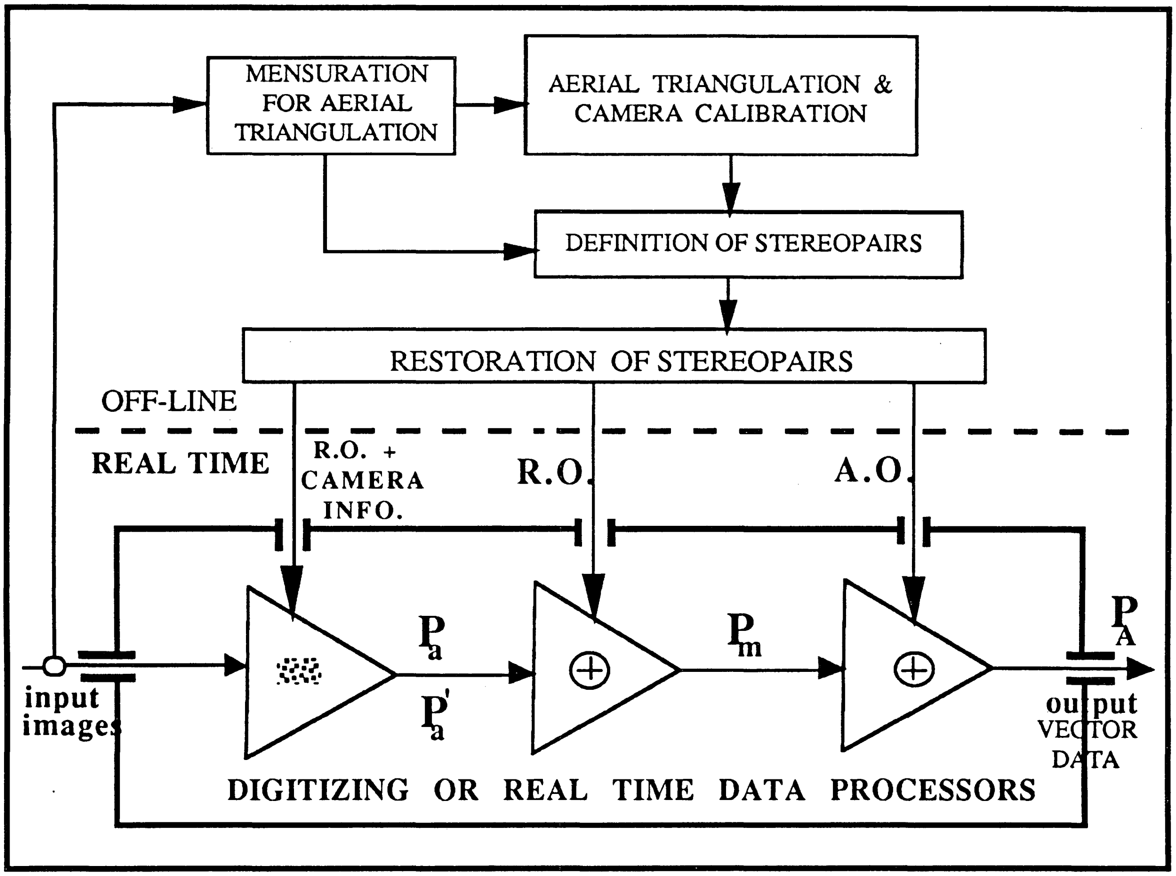


Figure 4-1a: The Workstation Concept of Analytical Photogrammetric Mapping

In order to bring the substance of this chapter into focus, the real time components will be considered as one subsystem or module in the workstation. Therefore, we generalize the ideal workstation representation in Figure 4.1a to the form shown in Figure 4.1b. From this figure we note that four main software modules are involved in the functioning of the workstation (system engine). These modules perform or facilitate the performance of some basic tasks such as mensuration, triangulation, stereomodel definition, and stereomodel restoration/digitizing which are necessary for the analog to digital data conversion processes within the system (see Fig 4-1b). These basic tasks and the various techniques by which each of them may be performed are summarized in what follows.

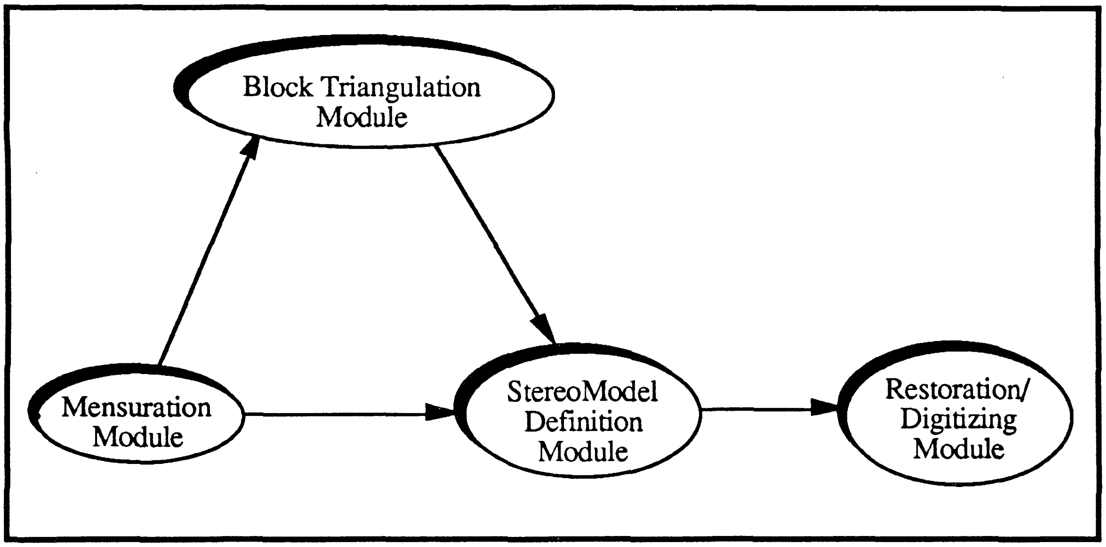


Figure 4-1b: The basic tasks and process flow at a photogrammetric workstation

4.1.1 Mensuration

This is the process of making initial measurements of either the photo coordinates or the model coordinates of selected points for use in a block adjustment. The human operator registers the photographs on the instrument stages, inputs all necessary information regarding the project into designated storage locations, and performs the measurements. Two options are available for this operation, namely: **comparator** mode, and **stereoplotter** mode. In the comparator mode, photographic coordinates of image points are measured on both the left and the right photographs usually in the fiducial coordinate system established by a process of plate registration during interior orientation.

The stereoplotter mode creates a three-dimensional model from two images from which coordinates are measured for each point in the model space system. Since the choice of one or the other option depends on the type of triangulation for which it is intended, these options are not given any identification name in the sequel.

4.1.2 Block triangulation

This links the block to the object space through a mathematical transformation process using measured photo or model coordinates together with a few object space control points. The results include the exterior orientation elements of the camera at each station (or of the stereomodels) and extended object space control points. In some cases the interior orientation elements of the camera may also be obtained from this process through self-calibration. Nevertheless, of all the operations involved in photogrammetric mapping, it is the most critical in terms of computational requirements. And because it provides the numerical framework for further detailed evaluation of the photographic images, it determines to a large extent the accuracy of the final product. Many approaches of varying sophistication may be employed. These range from the relatively inexpensive but less accurate approximation methods to the rigorous models technique and the high precision bundle method (Faig 1984, Heikkila & Kilpela 1984, Ghosh 1988). More often than not, the rigorous methods are preferred and are usually grouped into two, namely: the bundle approach which uses comparator measurements; and the independent model technique employing stereoplotter measurements. A third approach, which although, is mentioned in the literature but not commonly used in practice, is the so called hybrid mode which numerically converts measured photo coordinates to the equivalent model coordinates (Faig 1984, Ghosh 1988). The computed model coordinates are then used as input to the independent model block adjustment process in much the same way as the operator measured coordinates. In any event, irrespective of the approach used for the block adjustment, an object space numerical framework is provided by the triangulation module which may be used to control subsequent operations. For the bundle technique, this numerical structure is composed of the exterior orientation parameters of each photograph in the block and the object space coordinates of pass points; and for the model technique, it is made up of the absolute orientation parameters of each stereopair of photographs in the block and the object space coordinates of pass points. In this study, three options (i.e. bundle, hybrid and independent model techniques) are used; and for convenience of reference, they will be identified as follows: **BT** for bundle triangulation, **HT** for the hybrid, and **IMT** for the independent model triangulation.

4.1.3 Model Definition

The model definition task involves the aggregation of data from both the mensuration and triangulation units and the separation of this information into usable form for real time computations. Indeed, this process resolves the block into its component stereomodels and stores the data for each model in one data record. Using the results from the triangulation process, there are a number of ways by which this task could be achieved on the analytical plotter, especially in an integrated environment, in which the mensuration, triangulation, and definition modules communicate with each other (as represented in Fig. 4-1). These range from the operator-assisted definition to the fully automated type. To aid the understanding of these techniques the following naming convention is adopted. Each model definition technique is given a three-digit name. The first two digits (taken as individual digits) indicate the number of R.O. and A.O. or E.O.parameters (EO) to be determined by an adjustment procedure to define (or resolve) the pair of photographs involved. The last digit is a binary number which indicates whether or not the operator is to make measurements for the process. When this digit is one, the operator is making measurements and zero means otherwise. Now the following options are available:

1. **model definition by operator-assisted R.O. and A.O. (571)**:- This system recomputes the EO parameters of a stereopair by using the relative and absolute orientation procedures. The operator performs the interior orientation, the relative orientation, and measures model coordinates of pass points. The measured coordinates and their object space coordinates are used to compute the absolute orientation parameters for the pair of photographs. The relative orientation parameters (obtained when the operator re-established the model), and the computed absolute orientation parameters are then transformed into the required **image-model** and **model-object** space transformation operators (see section 3.5.5). This information, combined with the camera calibration data (needed for real time systematic error correction), are then stored for subsequent restoration and digitizing purposes. This process is repeated for all stereopairs in the block. It is important to note that the only information from the triangulation which is utilized in this process is the object space coordinates. Thus, this resolution technique may be used with any type of triangulation scheme. As a matter of fact, this method is common in situations where the control extension is performed separately from the rest of the process, such as when it is sub-contracted to a different organization. In such a situation, the operator often has to re-type the extended control point coordinates for use in the digitizing processes.
2. **model definition by operator-assisted 2-photo resection (661)**:- In this approach, the operator performs the interior orientation and measures the photo coordinates of the pass points, whose object space coordinates have been determined from block triangulation. A 2-photo resection computation is then performed for the EO elements. The required image-model and model-object transformation operators are then derived from the EO elements using the algorithm in sections 3.5.4.1 and 3.5.5. The derived data for each model is then stored for

further treatment of the stereopair. Other characteristics of this method are similar to those of (571) above.

3. **definition by automated R.O. and A.O. (570)**:- This takes advantage of an integrated environment to use the data from both the mensuration and the triangulation stages to compute the needed information. After the mensuration and the triangulation processes are concluded, both the image space and the object space coordinates of all points in each stereopair are available. These are then used in an automatic computational process to obtain the required data for subsequent restoration and digitizing of the model.
4. **definition by automated 2-photo resection (660)**:- This is the same as method 2 above except that, instead of the operator remeasuring the photocoordinates, advantage is taken of an integrated environment to use the data from the mensuration and the triangulation stages to achieve a determination of the required transformation matrices.
5. **model definition without re-orientation (000)**:- This method derives the transformation information from the mensuration and the triangulation processes without further rigorous parameter estimation. Different from the other options described above, this technique does not use the minor control points, but the camera or stereomodel exterior orientation data obtained in the triangulation process are employed. Depending on the type of triangulation procedure used, the amounts of arithmetic processing needed to extract the model information vary. For triangulation by models, the results of the triangulation and mensuration are simply mixed into the required form without extensive arithmetic calculations. The bundle triangulation technique however, requires that the relative and absolute orientation data for each stereopair be extracted from the exterior orientation information obtained from the triangulation. The algorithm given in section 3.5.4.1is used in the latter case.

4.1.4 Model restoration

The model restoration exercise involves the retrieval of the stored information for each stereopair, performance of the interior orientation (if the pair of photos involved have been removed from the instrument stages), downloading of the image to model spatial operators to the real time processor (processor P2 for the DSR-11) for the creation of the stereomodel, and downloading of the model to object spatial operators to the digitizing unit. For the restoration task, there is just one task and only one option. Technically, the module that performs this task could be just a part of the definition module or the digitizing module, or it could be a separate unit in its own right as shown in Fig. 4-1a. In this work however, it is associated with the digitizing module.

4.1.5 Digital data collection

This task involves the digitizing of a restored stereomodel. The operator selects points defining a feature of interest in the stereomodel, and such points are transformed to the object space system and vice versa, usually in real time. There are two tasks performed by this system unit: (1) real time spatial transformation, (2) communication with an editing station or a GIS. While it is possible to store digitized data into a file for off-line editing and processing, operating the photogrammetric system as a workstation within a spatial information environment has a number of advantages which include: immediate availability of data for GIS applications, elimination of intermediate data handling, flexibility of updating the database, use of the photogrammetric workstation to assist in GIS applications, and reduced production cost. For this purpose however, an interface is required between the photogrammetric system and the GIS. The details of one such interface especially developed in this study is covered in the next chapter. For the spatial transformation of measurements, two computational options are available in principle

(Reece & Kleinn 1980, Kratky 1988), namely: 3-D spatial transformation (3-D collinearity), and spatial intersection of bundles (using 2-D collinearity). In practice, the former approach is used by virtue of its computational advantages as we have seen in chapter 3.

It is obvious that through an interplay of the various options discussed above, a number of different production arrangements may be constructed, all offering an integrated solution to the mapping problem. Undoubtedly, the relationships between the block triangulation process and the mensuration process, between the block triangulation process and the model definition process, between the mensuration process and the model definition process, and between the model definition process and the digitizing process, determine the level of integration achievable for the particular configuration, and this in turn affects the labour consumption, time requirements and efficiency of the system. The configurations which have been implemented on the DSR-11 at UNB are described in the next section. It is important to state at this point, that in subsequent discussions, the terms "digitizing module" or "digitizing unit" will be used to mean the combination of the processes of model restoration and digitizing unless otherwise noted.

4.2 System Configurations

A number of different system configurations are easily identifiable through interconnections of the different mensuration, triangulation and model definition options described in the previous section; however, only eight of these are considered in this study. Again, for ease of identification in the sequel, the following naming convention is adopted: Each name is composed of two sets of characters separated by a dash. The first set indicates the type of block triangulation package used in the system while the second set indicates the type of model definition method employed by the system. As explained in section 4.1.3, the last digit of the name is a binary number whose value is 0 for an automated definition (e.g.

options 3, 4, 5 of section 4.1.3), and 1 for the operator assisted definition (options 1and 2, section 4.1.3). Thus BT_000 refers to a system employing bundle triangulation with direct model definition from the camera information without further adjustment. It is an automated definition type because the last digit is zero. Using this nomenclature, a few of these systems are identified as: IMT_000, IMT_571, HT_000, HT_570, HT_660, BT_000, BT_570, and BT_660. However, for reasons of easier reference and notational convenience in the optimization process, alternative names are given as follows: X_1 for IMT_000, X_2 for IMT_571, X_3 for HT_000, X_4 for HT_570, X_5 for HT_660, X_6 for BT_000, X_7 for BT_570, and X_8 for BT_660. These names will be used interchangeably in the rest of this chapter. On the whole, there are eight design options from which a selection may be made for constituting our workstation. These designs are described in the following subsections, more in a diagramatic fashion than textual. This is done for reason of clarity and brevity. Nevertheless, it is helpful, in interpreting the figures, to note that the model definition unit makes up an important component of each system, and is almost invariably, the distinguishing element between systems in which other system units are the same. Moreover, in the accompanying description, the communication between the digitizing station and the editing station (or a GIS environment) is neither stressed nor distinguished here since this is covered in chapter 5.

4.2.1 IMT_000 (X_1)

This is a configuration involving an Independent Model Triangulation without re-computation for the model definition. Figure 4-2 illustrates the software components of this system. The definition module accepts the camera calibration parameters and the relative orientation elements from the mensuration module (these are usually stored in files during the mensuration process), and the absolute orientation elements for all the models in the block from the triangulation module. The required image-model and model-object transformation operators are constructed and stored for use in the restoration and digitizing processes. Single or multiple models may then be restored without further computation except for the registration of the photographs if they had been removed.

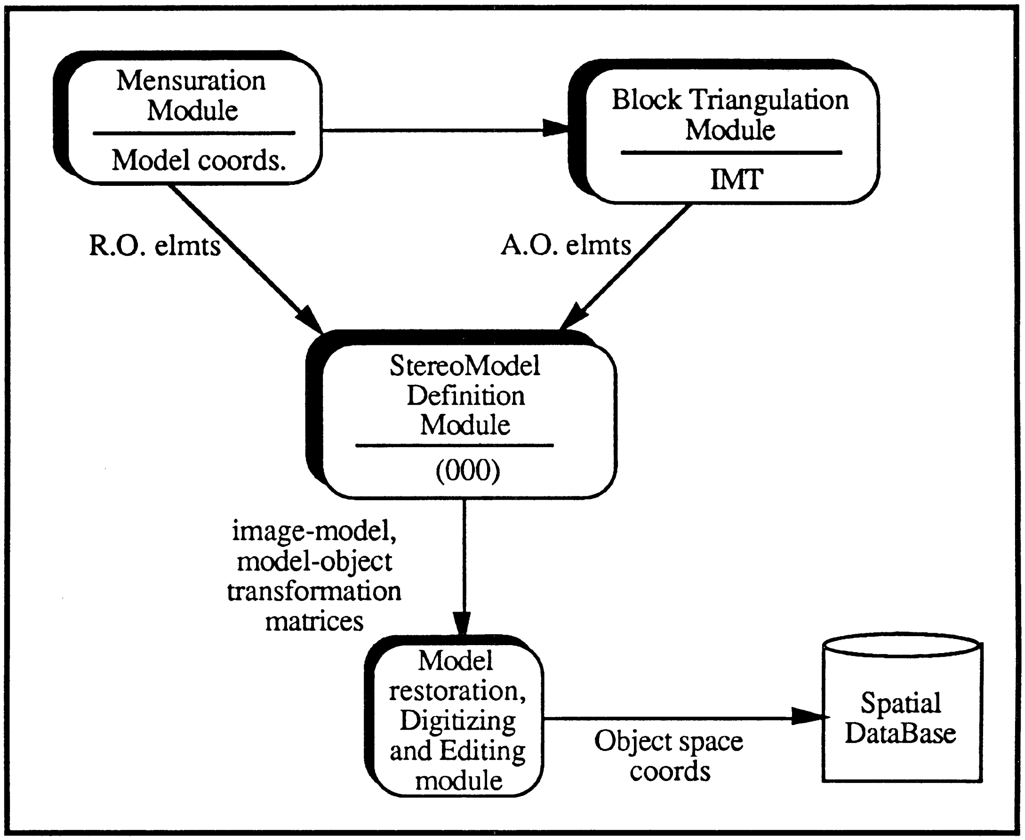


Figure 4-2: Software structure for system IMT_000 (X_1)

4.2.2 IMT_571 (X_2)

This involves an Independent Model Triangulation and an operator-aided model definition. Figure 4-3 shows the software layout for this system. It makes use of extended control points from the block triangulation in an operator-assisted single model reorientation. Integration between the mensuration and the restoration modules is not essential since the operator has to repeat all the orientations and measurements. In cases where the triangulation is done separately (outside the system) or when the data formats of the triangulation and model definition modules are not compatible, the operator serves as the link by entering the data through the terminal. For the model definition process, the operator performs the interior and the relative orientations, then measures the model coordinates of the minor control points and enters their identification numbers. The definition module then computes the absolute orientation data from which the required transformation parameters are extracted. In the case of the DSR-11, dedicated program modules for these orientations are chained together and run in sequence by a program called DSR1B. The main problem with this system is the duplication of efforts involved in the definition of the models. Its advantage lies mainly in the fact that users are free to purchase or develop their own triangulation packages or leave that aspect to another organization instead of being tied to a particular software type.

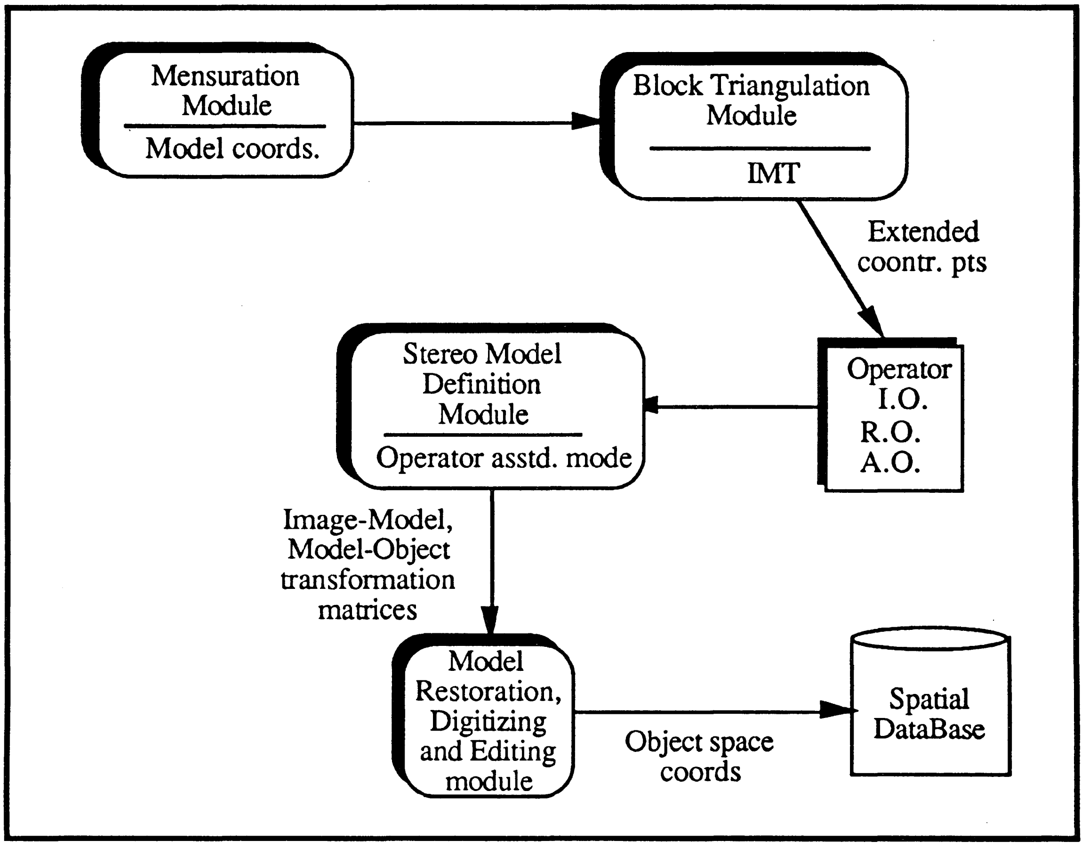


Figure 4-3: Software structure for system IMT_571 (X_2)

4.2.3 HT_000 (X_3)

This uses Analytical Independent Model Triangulation without re-computation to resolve the block into its model constituents. This method differs from the IMT_000 in that photocoordinates are measured instead of model coordinates (see Figure 4-4). The model coordinates are computed as part of the triangulation process. This saves operator time since stereomodels are not created in the process of making initial measurements. The restoration data is derived as described in option 5 of section 4.1.3.

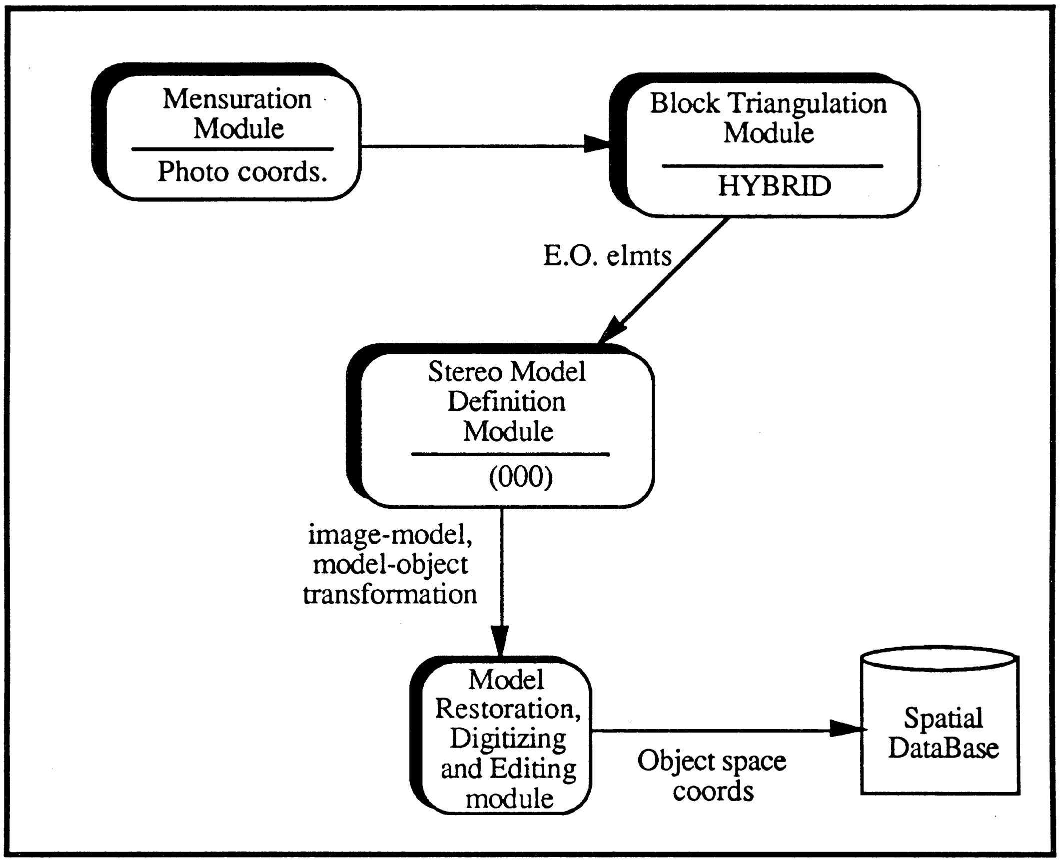


Figure 4-4: Software structure for system HT_000 (X_3)

4.2.4 HT_570 (X_4)

This uses Analytical Independent Model Triangulation and automated re-computation of R.O. and A.O. parameters. This design makes use, in the definition process, of the extended object space control points (including minor control or passpoints) from the triangulation module plus the photocoordinates from the mensuration unit.

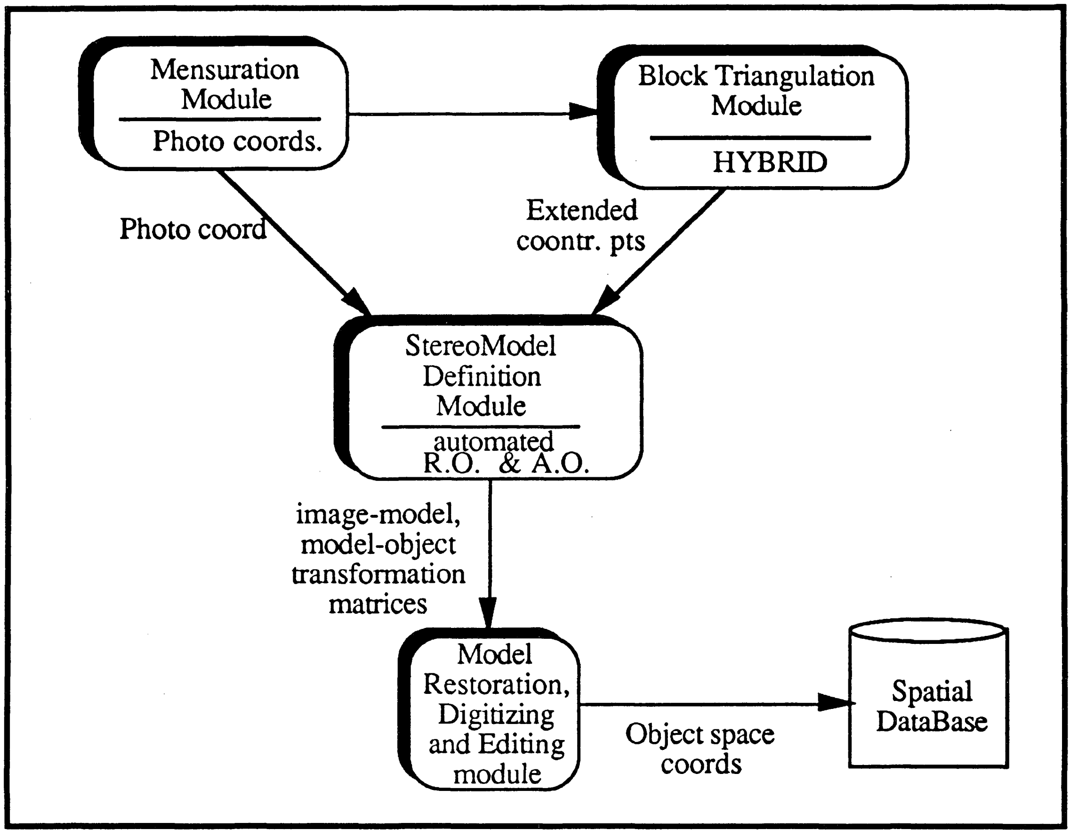


Figure 4-5: Software design for system HT_570 (X_4)

4.2.5 HT_660 (X_5)

This system computes the model definition data for all stereopairs by an automatic 2-photo resection. It makes use of the photo coordinates from the mensuration unit and the extended control points from the triangulation module to compute the exterior orientation elements for each photograph. The image-model and model-object space transformation matrices are then derived therefrom. In this procedure, the absolute orientation information for the stereogram is taken as the exterior orientation parameters of the left photograph. An arbitrary scale may be used for the model or an approximate one computed. Usually, a constant scale is selected for all models when using this technique.

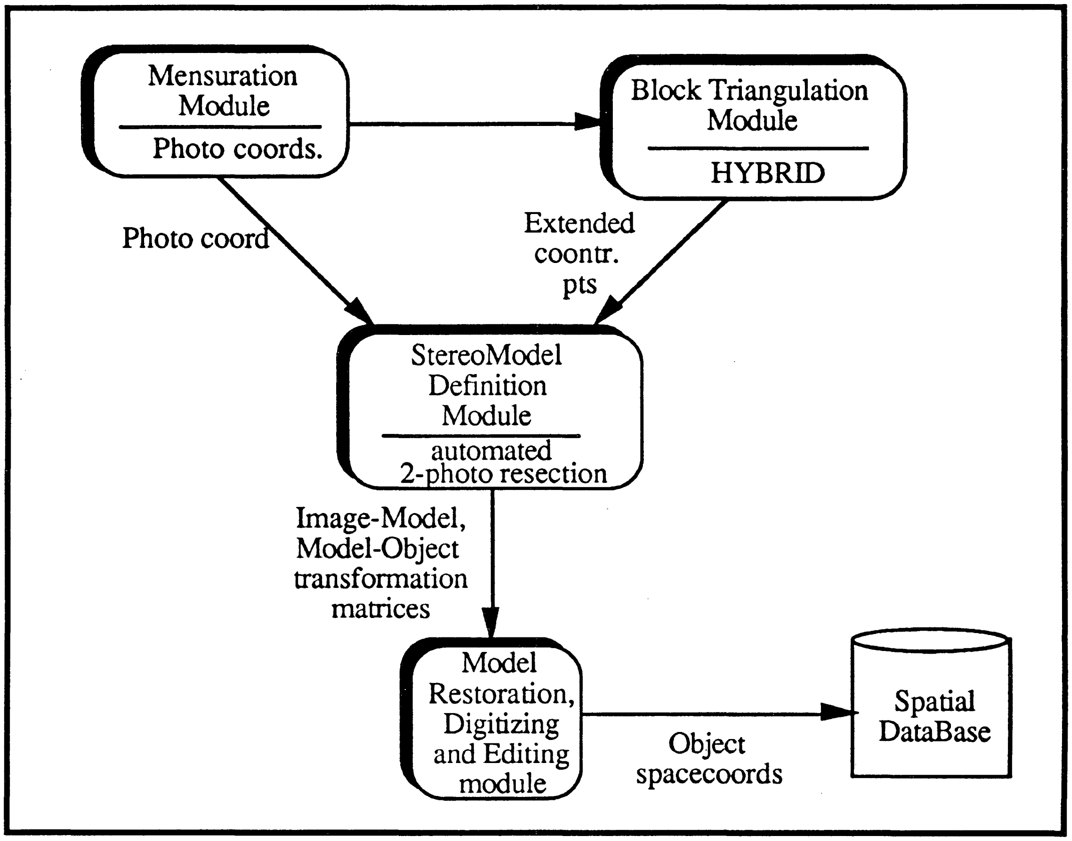


Figure 4-6: Software design for system HT_660 (X_5)

4.2.6 BT_000 (X_6)

This system uses Bundle Triangulation and an adjustment-free model definition (Figure 4-7). The only information needed from the mensuration unit is the camera calibration data and this is supplied along with the measured photocoordinates to the triangulation unit, the output of which is utilized for the model definition.

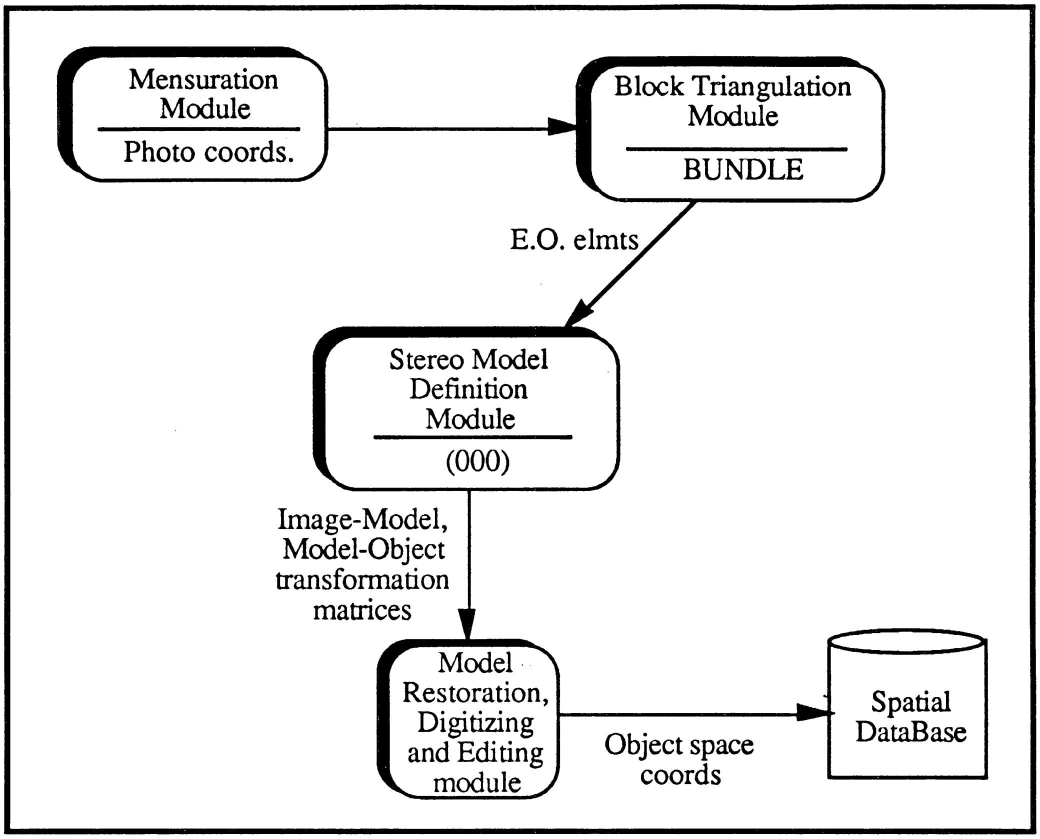


Figure 4-7: Software structure for system BT_000 (X_6)

4.2.7 BT_570 (X_7)

This uses the Bundle Triangulation with model definition by automated re-orientation. In this system (Figure 4-8), extended control points from the bundle triangulation are used together with the measured photo coordinates from the mensuration module. The re-orientation is done automatically. Except for the triangulation module, this configuration is similar to that for system HT_570 described earlier.

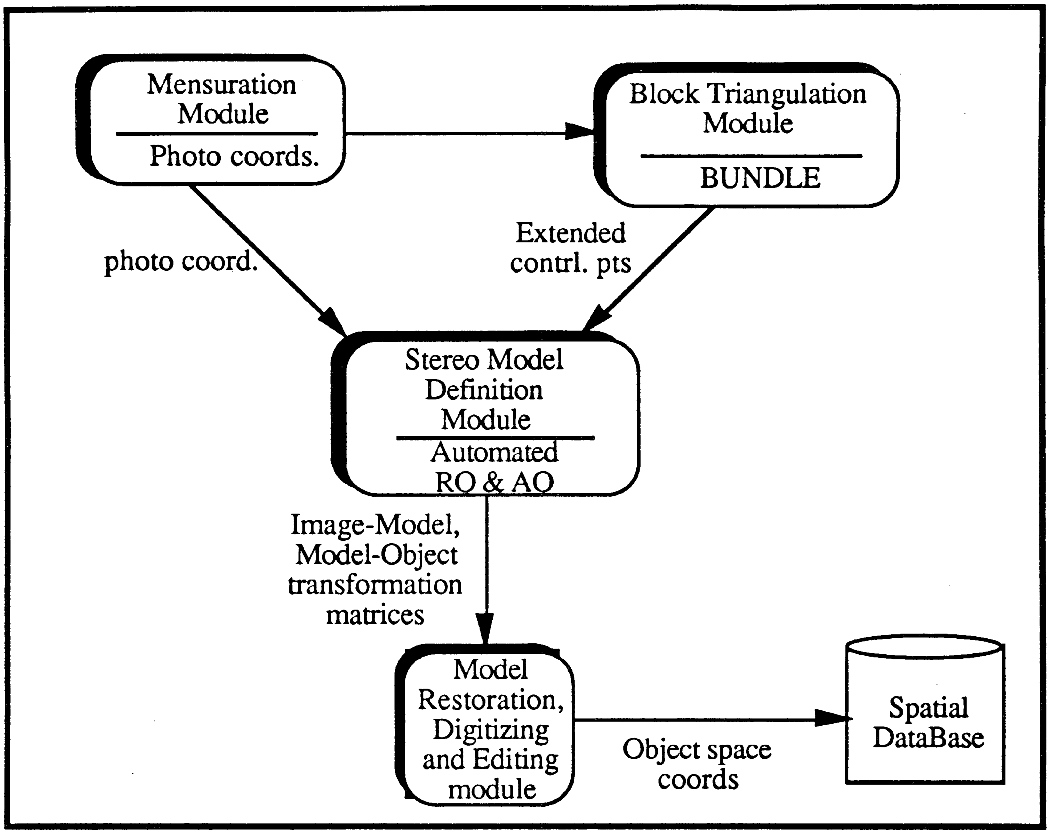


Figure 4-8: Software structure for system BT_570 (X_7)

4.2.8 BT_660 (X_8)

This design combines Bundle Triangulation with automated 2-photo resection computation to resolve a stereomodel (Figure 4-9). It is similar to the system HT_660 except for the triangulation module.

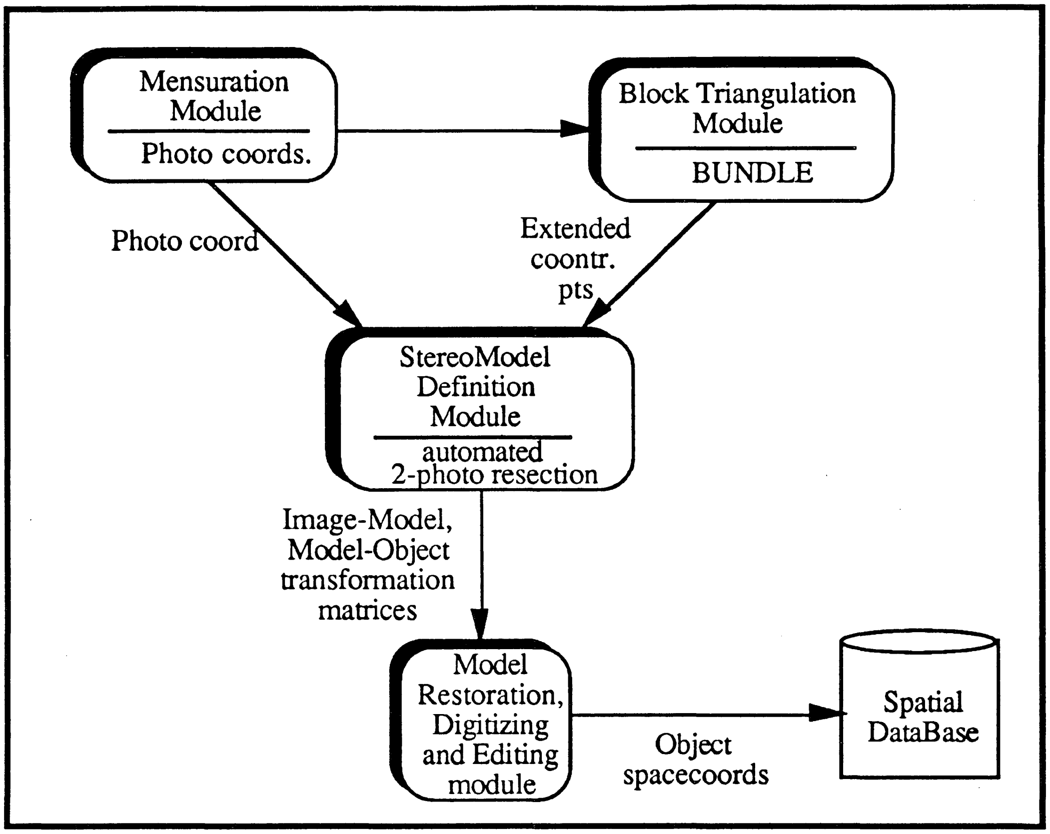


Figure 4-9: Software structure for system BT_660 (X_8).

The above systems have been implemented on the Kern DSR-11. Nevertheless, in order to avoid a cluttered menu system, provision is made for the engagement of only three at any one time. This is explained in chapter 6.

4.3 Selection of the Optimum Design by Mathematical Programming

One of the major challenges of this work is the attempt to evolve a precise, albeit empirical method of selecting the best system design in terms of the performance measures of system characteristics and behaviour. Tasks of this nature are usually performed using techniques of optimization by mathematical programming. For the purpose of this study, linear mathematical programming is employed; this approach has the virtues of simplicity, preciseness, elegance, and, in many cases, mathematical tractability. Yet, a cost model (or some performance function) is required, and this may be established either by theoretical formulation or through an empirical process. The latter approach has been used in this study. We begin with a brief description of the key terms in optimization theory as applied in the present work.

Optimization has been defined as an art of obtaining best policies (or decisions) to satisfy certain objectives, at the same time satisfying some fixed requirements (Luenberger 1969, Killen 1983). Put more simply, it means a process of selecting a best way to accomplish a task that involves fixed requirements. Applied to our case, it is a process of choosing the best of all the system designs considered in the previous sections in order to achieve mapping at the minimum possible cost. The objective is usually a single real quantity, which summarizes the performance or value of a decision (or selection). It is expected to reach a minimum or a maximum when the best selection among available alternatives has been made. Often, it is expressed as a function of the available alternatives. Constraints are limitations placed on the operations of the system, and are usually expressed as algebraic statements. In our case, these constraints are the limitations imposed by available resources for the execution of a particular mapping project. As Jaksic (1980) has noted, system optimization is dependent on user requirements and resources. For example, there is always a time limit for the completion of a project. Also, available human-hours are limited by the number of employees and how many hours they can each work per month. Furthermore, accuracy of the derived mapping is a major requirement which must be met. There are a host of other terms used in the general theory of optimization but the ones defined here suffice for the purpose of our analysis. Excellent treatments of optimization concepts and applications exist richly in the literature (Gottfried and Weisman 1973, Adby and Dempster 1974, Papalambros and Wilde 1988).

4.3.1 Derivation of the objective function

Considering each of the system designs discussed earlier as production lines in a mapping industry, our first task is to evolve an objective function which will express the cost of executing a mapping project as a function of certain characteristics of each individual system. For example, we may consider all the systems as parallel flow lines through which a certain substance must pass (see Figure 4-10 for illustration). The properties of such flow lines may be the amount of resistance each of them poses against the flow, thus the total amount of substance it allows through will directly depend on the resistance. A less resistive flow line will obviously allow more flow. Using a parallel argument, the number of stereomodels that will be processed through each system will depend on the efficiency of that particular system. Thus we may derive a functional relationship between the cost of a project and the efficiency of each system. Nevertheless, since there is no existing theoretical formulation for this phenomenon, an empirical mathematical model is developed in this study for the purpose.

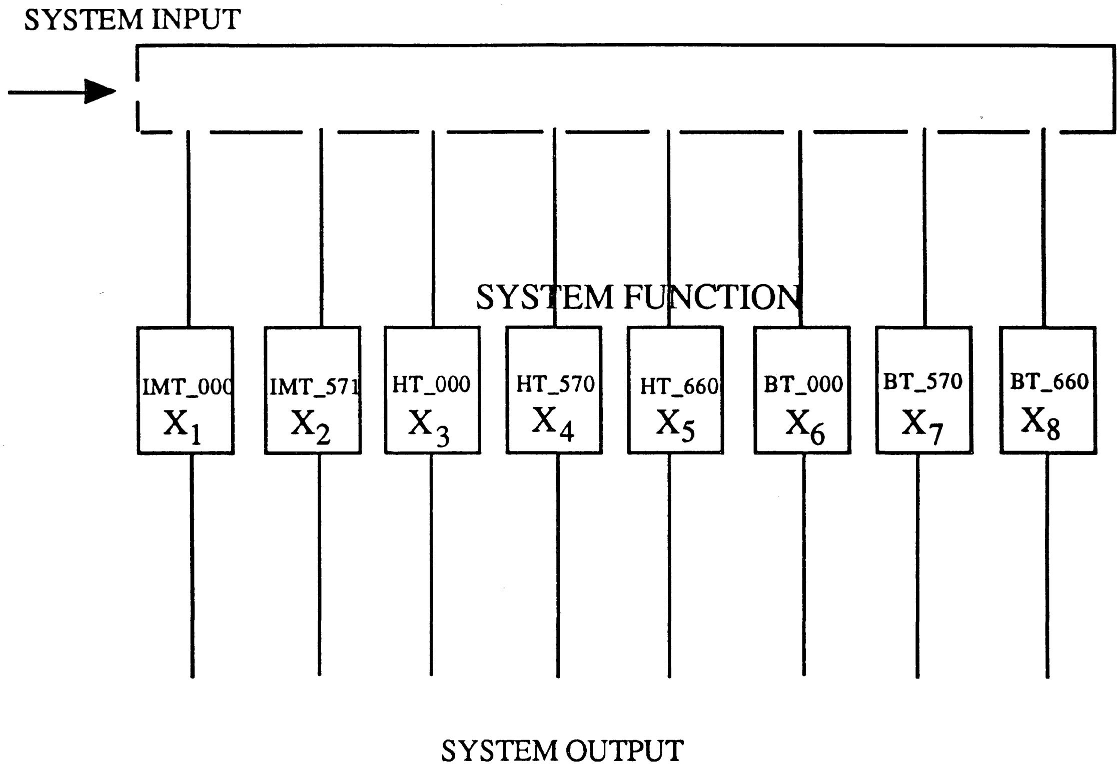


Figure 4-10: Combination of different systems as parallel production lines

Let the cost of a mapping project using the collection of systems being treated here be functionally represented as:

$$Z = f(X_1, X_2, X_3, \ldots, X_8) \tag{4-1}$$

where

Z = total cost (in dollars)

$X_1, X_2, \ldots, X_8$ are the number of units of production (No. of models) that is processed by the corresponding systems denoted as $X_1, X_2, \ldots, X_8$ respectively.

Linearizing equation 4-1 using Taylor's expansion to first order terms, we have

$$Z = Z_o + \frac{\partial Z}{\partial X_1} dX_1 + \frac{\partial Z}{\partial X_2} dX_2 + \text{---} + \frac{\partial Z}{\partial X_8} dX_8 \tag{4-2}$$

Z_o is the value of the function at some initial value of the variables.

Equation 4-2 may be interpreted as follows: the total production cost is the sum of a fixed amount (Z_o) and some small variations due to the actual production process. The contribution to the production cost by each production line is directly related to the number of units of production (stereomodels) that is actually produced on that system. It is obvious that the minimization of the cost function Z may be achieved by minimizing the contributions (i.e. the variations) from the different systems. In other words the cost function for the purpose of our optimization may be modified by evaluating equation 4.2 at the origin (i.e. when initial values of the variables are $X_1 = X_2 = ... = X_8 = 0$), and setting $Z_o = 0$, so that we may write the cost function as:

$$Z = \frac{\partial Z}{\partial X_1} X_1 + \frac{\partial Z}{\partial X_2} X_2 + \text{---} + \frac{\partial Z}{\partial X_8} X_8 \qquad (4\text{-}3)$$

This is the fundamental variation function we are seeking for our cost model. Denoting the coefficients of (4-3) by U_i we may write

$$Z = U_1 X_1 + U_2 X_2 + \text{---} + U_8 X_8 \qquad (4\text{-}4)$$

where U_i are the cost coefficients (or value coefficients) per production unit (stereomodel) produced through system X_i; and the X_is are the system variables which may be interpreted as the number of stereomodels processed through system X_i. In any case, the evaluation of the coefficients of equation 4-4 is a task which can be achieved only through practical studies or from past experience. This task is described later in this chapter.

4.3.2 Derivation of the constraint functions

It has been stated many times before, that the best design we seek must have a high level of integration among its functional components to meet the requirements of minimum operator involvement, high accuracy of digitized data, and minimum total project time. These requirements, often determined by the amount (or capacity) of resources available or by project specifications, may be imposed as constraints by expressing each requirement as a function of the system variables and requiring that the value of such function should not exceed or be less than available resources for the task at hand. For the sake of brevity it is stated here that through a parallel argument to that used for the empirical cost function, the constraint function for each of the afforementioned requirements may be written as:

$$Q = \frac{\partial Q}{\partial X_1} X_1 + \frac{\partial Q}{\partial X_2} X_2 + \text{---} + \frac{\partial Q}{\partial X_8} X_8 \qquad (4\text{-}5)$$

again, denoting the coefficients of (4-5) by C_i we may write

$$Q = C_1 X_1 + C_2 X_2 + \text{---} + C_8 X_8 \qquad (4\text{-}6)$$

where C_i are the constraint coefficients (or constraints) per production unit (stereomodel) One equation of the type (4-6) represents one type of constraint and is usually expressed as an inequality equation, i.e.

$$C_1 X_1 + C_2 X_2 + \text{---} + C_8 X_8 \leq Q \qquad (4\text{-}7)$$

This implies that the total amount of a certain resource to be employed cannot exceed (or must not be less than) the value assigned to the quantity Q. For example, suppose it is stated that the available operator time for a project is 10 hours; this translates into a constraint on the project as "the sum of the operator time used by all the systems involved must not exceed 10 hours". A similar argument holds for each of the other constraints. The evaluation of the constraint coefficients also requires experimental data to give definite functional representations of these constraints. This is further explained in section 4.6.

4.4 System cost per unit of time

A critical requirement for the evaluation of the coefficients of both the performance function (eqn. 4-4) and the constraint functions (eqn. 4-7) is the cost of each system involved related to some operational time unit. Thus, preparatory to the practical test leading to this process, the estimation of the cost of installing each of the system configurations (including hardware, software and the human operator using it) to full production capability is undertaken here.

4.4.1 Cost of Hardware + Software

The cost of the analytical plotter system may be expressed as

$$\text{Cost} = IC + SC \tag{4-8}$$

where

IC = initial purchase cost (hardware and operating system software only)

SC = application software cost e.g. for the software packages for each of the systems being considered.

Since this study is based on the Kern DSR-11, the initial cost of the analytical plotter system (hardware, computers, operating system software) will simply be assumed to be the amount paid for the DSR-11 by the University of New Brunswick. For the application software packages, estimating the cost would be easy if the software packages were purchased from a vendor. However, since these were developed in-house, estimating the cost is not a simple process because many factors are involved which require the expertise of a professional to quantify. Fortunately, a number of formulations exist in the literature for the costing of software packages (Kligman 1973, Clapp 1976). The simplest of these as given by Kligman is adopted here. Basically, it expresses the cost of a software package as

$$SC = \left[\frac{N * H}{L}\right] * \left[\frac{B}{100}\right] \tag{4-9}$$

where

SC is the cost in dollars.

N is the total number of lines of source code.

L is the number of lines of fully debugged and documented source code per hour.

H is the hourly rate for a programmer.

B is the overhead expressed as a percentage (includes computer usage time and materials for program checkout but does not include programmer management).

Thus the cost of the analytical system may be expressed as

$$\text{Cost} = \text{IC} + \left[\frac{N * H}{L}\right] * \left[\frac{B}{100}\right] \qquad (4\text{-}10)$$

Specifying values for the constants in eqn. 4-9, the initial purchase cost of the DSR-11 in 1987 was 161,000 dollars (CanLabINSPIRE documentation 1987). Furthermore, using the values H = 7.50 dollars/hour, B = 200, L = 4 as given by Kligman (1973, p.53) equation 4-10 may be written as:

$$\text{Cost} = 161{,}000 + \left[\frac{N * 7.50}{4}\right] * \left[\frac{200}{100}\right] \qquad (4\text{-}11)$$

The number of lines of software code for each system as developed in this study is given in table 4-1. Substituting these values for N in (4-11), and adding a 10% allowance for maintenance and downtime compensation over a 10 year amortization period (the amortization period is necessary in order to be able to estimate the cost per unit of time, and a 10 year period is chosen for convenience), the cost of each system (hardware + software) is shown in table 4-2. And using the amortization factor (based on 30 day month), the cost per minute for each system is shown in table 4-3. Knowing the cost per minute for each system, the cost for processing one stereomodel on any system may be estimated if the time taken to process the model is recorded. If one model is processed in 5 minutes on IMT_000 for example, the cost will be 21.62 cents, and if the same model is processed in 30 minutes on this system, the cost will be 1.30 dollars. This information will be used later in the practical test.

Table 4-1: Lines of Application software codes

	System							
	IMT_000	IMT_571	HT_000	HT_570	HT_660	BT_000	BT_570	BT_660
Lines of Code	11390	11323	12810	13543	13245	11958	12588	13235

Table 4-2: System cost in dollars (Hardware + Software)

	System							
	IMT_000	IMT_571	HT_000	HT_570	HT_660	BT_000	BT_570	BT_660
Cost	203712.50	203461.25	209037.50	211786.25	210668.75	205842.50	208205.00	210631.25
10% allowce.	20371.25	20346.13	20903.75	21178.63	21066.88	20584.25	20820.50	21063.13
Total	224083.75	223807.38	229941.25	232964.88	231735.63	226426.75	229025.50	231694.38

Table 4-3: System Amortization cost per minute in cents (Hardware + Software)

	System							
	IMT_000	IMT_571	HT_000	HT_570	HT_660	BT_000	BT_570	BT_660
Cost per min.	4.323	4.317	4.436	4.494	4.470	4.368	4.418	4.469

4.4.2 Operator rate per minute

Turning now to the evaluation of the cost of human labour involved in the process, and accepting the parameters of an experienced operator able to work 45 hours per week and earning a weekly rate of 750 dollars (these figures are based on advertised rates commonly given in the PE&RS Journal), then the per minute rate of this operator amounts to 27.78 cents. This rate is assumed the same irrespective of the system on which the operator works. In later discussions, this rate is simply referred to as OPR (for Operator Pay Rate: OPR = 27.78 cents).

4.5 Practical Tests

The primary objective of these tests was to collect data for the determination of the coefficients of the performance function (eqn. 4-4) and the constraint functions (eqn. 4-7). The tests were therefore designed to determine such quantities as (1) the time needed to process a stereomodel on each of the systems described, (2) the amount of operator's labour needed (measured in the unit of time) per stereomodel, and (3) the level of accuracy of the spatial data attainable with each system. However, an accurate determination of the characteristics of each system, as to these criteria, requires more extensive practical investigations than can possibly be accomplished within a short time frame. Fortunately, because most of the basic operations are common to some systems (some are in fact common to all systems), and their technical details and requirements are independent of the systems from which they are performed, it is possible to ascertain the characteristics of the systems through a "once for all" test of the basic tasks. Because these tasks, in various combinations, form the components of the systems being evaluated, aggregates of their characteristics provide close approximations to those of the systems. For this reason, this experiment has tried to establish the behaviour of each system through the testing of its components within the environment of the integrated system. This approach offers flexibility since both real and simulated data may be employed for the component tests. Specifically, processes such as interior orientation, relative orientation, photo and model coordinate measurements, block triangulation, stereomodel definition and stereomodel restoration were empirically evaluated for the resources required per unit model. The strategy being, in the first case, to evaluate, in the environment of one of the systems, operations such as interior orientation, relative orientation, photo and model coordinate measurements, and taking such estimates to be the same for all systems employing the process. Secondly, operations such as stereomodel definition, stereomodel restoration and systems spatial accuracy were evaluated within individual systems, and finally, the triangulation process was investigated within the environment of a system using a particular triangulation method.

Yet, different processes require different experimental set ups for their evaluation. On the one hand, because operations such as interior orientation, relative orientation, and stereomodel restoration involve mensuration procedures (indeed, some form of computation is involved), ascertaining their characteristics necessarily requires that actual photographs be set up on the instrument stages. Therefore, only a real block of photographs may be used for this kind of test. For the stereomodel definition and the triangulation operations, either a real block or a simulated block is suitable. However, simulation allows flexibility since the parameters of the block may be modified or extended at will, a process which is expensive or impossible to do with a real block. Consequently, both real and simulated blocks of photographs have been used in this experimentation. Before discussing the test procedures and results, the parameters of these blocks of photographs are given.

4.5.1 Edmunston Block

The real block used in this investigation is the Edmunston Block, a set of photographs which covers an area in north-western New Brunswick, and comprises 15 photographs in three strips (12 stereomodels), at a scale of 1:7000 with 60% and 30% forward and side overlaps respectively. Figure 4.11 shows the layout of the stereomodels and also the configurations of the pass points, check points and the control points. The check points and points appearing in several models were used for the accuracy assessment of the systems as will be explained later.

4.5.2 Simulated Block

The simulated block was procured using the fictitious block generating program "DATAGEN" developed by Woolnough (1973) and adapted for use in the VAX VMS environment by the author. Since the simulation was intended to extend the experiment beyond the size of the real block, it was generated in sizes of 21, 33, and 45 photographs in 3 strips corresponding to blocks of models of sizes 18, 30, and 42 respectively. The photograph scale was 1:7000, with 60% and 30% forward and side overlaps to maintain consistency with the real block. Nevertheless, the behaviour of the program for blocks comprising two or more strips is not without problems. Often, the program assigned the same set of coordinates to the same point on different photographs, and sometimes, two or more points had the same set of coordinates on some photographs. However, it was possible to filter out the bad points from those blocks used in the investigation.

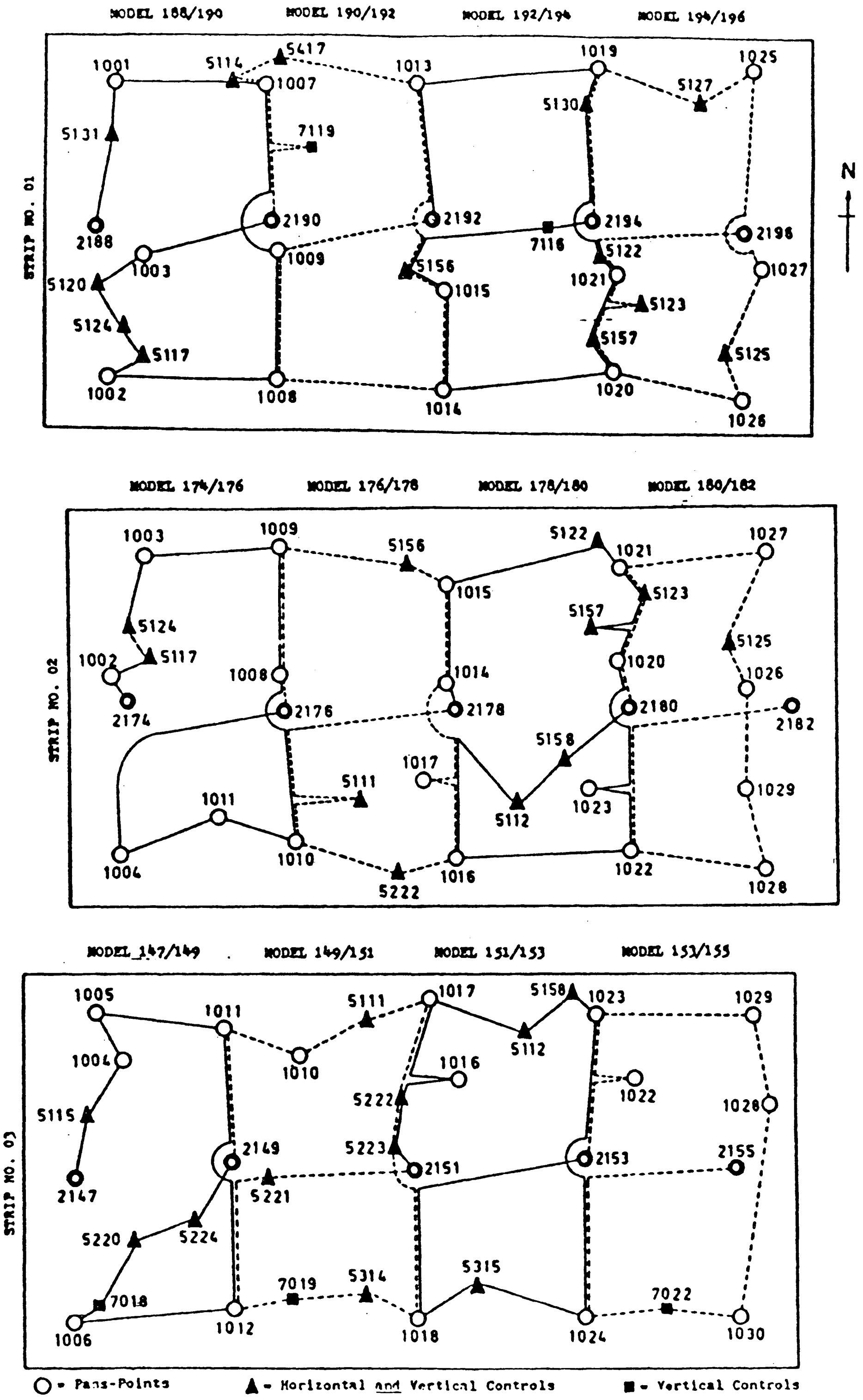


Figure 4-11: Emunston Block

4.5.3 Test Procedures and results

4.5.3.1 Interior orientation

For this operation, the standard process time and operator participation (measured in the unit of time) were determined by orienting the 15 photographs of the Edmunston block on the instrument. Time records were taken from the moment of initiating the process to its termination. Column 1 of Table 4-5 shows the time recordings from this experiment. The operator's participation in the I.O. process was measured by timing the periods in which he was actually physically involved in the process. Specifically, for plate registration, the operator had to locate the measuring mark on the fiducial point four times for each photograph (for a photograph equipped with four fiducial marks) and assess the residuals for goodness of fit through operator-program interaction. By timing these periods, the operator's time for plate registration was established. These records are shown in column 2 of Table 4-4. The average times were taken as the standard process time and operator's time for interior orientation of one photograph; and these would be the same for all systems employing the I.O. process. Although, using the base-in and base-out arrangement, the number of interior orientations could be reduced to the number of models plus one as against twice the number of models without this facility, nevertheless for simplicity, two interior orientations are assumed for one stereopair; and by this, the time per unit of production for the interior orientation process would be twice the average time shown in table 4-4.

Table 4-4: Time for repeated I.O. and operator time
(single measurements at each fiducial point)

Photo ID	Process time (min.)	Operator time (min.)
188	1.77	0.80
190	1.75	0.79
192	1.65	0.72
194	1.69	0.75
196	1.81	0.83
174	1.77	0.80
176	1.73	0.81
178	1.66	0.74
180	1.67	0.73
182	1.65	0.73
147	1.82	0.84
149	1.83	0.84
151	1.70	0.76
153	1.82	0.83
155	1.69	0.78
mean	1.73	0.77
per stereopair	**3.46**	**1.56**

4.5.3.2 Relative orientation

With the 15 photographs of the Edmunston block, 12 rounds of relative orientation were performed to create stereomodels. Following predefined patterns of 12 points chosen around the von Gruber locations, the instrument drove to each of these points and paused for the operator to place the measuring marks on conjugate image points. The time taken for the process was recorded for each model as shown in column 1, table 4-5. Furthermore, as the operator began to position the measuring marks, a stopwatch was activated, and upon completion the stopwatch was released. By summing these times during an R.O. session, the amount of operator time was obtained as shown in column 2 of table 4-5. Again, the average times were accepted as the standard time and operator's time required for the relative orientation of a pair of photographs, and would be the same for all the systems using this process.

Table 4-5: Process time and operator's time for R.O.
(12 points per model)

Model ID	Process Time (min.)	Operator Time (min.)
188/190	5.83	3.52
190/192	5.75	3.46
192/194	5.28	3.09
194/196	5.49	3.25
174/176	5.83	3.52
176/178	6.01	3.67
178/180	5.38	3.17
180/182	5.34	3.13
147/149	5.84	3.53
149/151	5.30	3.10
151/153	5.59	3.43
153/155	5.71	3.40
mean	**5.61**	**3.36**

4.5.3.3 Photo and Model coordinate measurements

For the measurement of coordinates, the 15 photographs were measured in 12 stereopairs both in the comparator mode and the stereoplotter mode. For this purpose, a total of 18 points were selected in the overlap area of each stereopair. These points included pass points, control points and check points. In practice, measurement of this type is often made for aerotriangulation purposes only. While the number of points that may be measured is not fixed, the choice of 18 was considered satisfactory to reduce the measurement load. The time records for both methods are shown in table 4-6. Note that these times do not include the preliminary operations of I.O and R.O. Perhaps it is necessary to remark that it was difficult to identify the times when the operator was not doing something during this process. The movement of the marks to the points, the placement of the marks on the selected points and the taking of the measurement, all involve the operator; the time records for operator's participation were therefore the same as the process times. The averages of these times would make up the standard times for one stereopair, and is the same for all systems using these processes.

Table 4-6: Time for coordinate measurements for 12 stereopairs
(18 points per pair)

Stereo pair	Comparator mode	Stereoplotter mode
188/190	7.82	6.33
190/192	7.11	6.25
192/194	6.97	5.84
194/196	7.45	6.02
174/176	7.51	6.48
176/178	7.03	6.32
178/180	6.99	6.33
180/182	7.02	5.89
147/149	7.09	5.93
149/151	7.13	5.85
151/153	7.05	6.51
153/155	7.67	6.56
mean	**7.24**	**6.19**

4.5.3.4 Block Triangulation

Triangulation times for five blocks of different sizes were determined. The Edmunston block (see section 4.5.1) was used in two blocks of 6 and 12 stereopairs, and three simulated blocks of 18, 30 and 42 stereopairs respectively as described in section 4.5.2. The simulated blocks were used in order to ascertain the behaviour of each triangulation method beyond the size of the Edmunston block used for other aspects of the investigation. Table 4-7 shows the total time records for the five blocks, while Table 4-8 shows the variation of the time per unit model with block size. This is obtained by dividing the entries in Table 4-7 by the number of models in each block. From Table 4-8, we see that the time per unit model for the triangulation process does not vary in a simple way as determined in this investigation. In general, it is observed that the time per model increases with the block size for the three triangulation techniques. It is also noted that for small block size, bundle adjustment takes almost twice as long as for the other two methods. The trend is however observed to be reversed at large block sizes for the bundle and the hybrid methods. Furthermore, there is a trend in the time per model, which is apparent for block sizes 30 and 42 stereopairs in Table 4-8, and the disparity between the times is higher at blocksize 42 than for blocksize 30 stereopairs. Therefore, to evaluate the contribution of each triangulation approach to the efficiency of the system of which it is a component, the time distribution for the block of 42 stereopairs has been adopted as the standard per model time for each of the triangulation techniques This time will be the same for systems employing the same triangulation method.

Table 4-7: Triangulation time for different block sizes (minutes)

Block size	Triangulation		
Model/photo	Bundle	Hybrid	IMT
6/9	2.10	1.01	0.99
12/15	10.69	7.92	7.89
18/21	29.37	26.67	25.01
30/33	118.56	123.45	113.85
42/45	331.20	338.90	288.68

Table 4-8: Triangulation time per stereopair for different block sizes (minutes)

Block size	Triangulation		
Model/photo	Bundle	Hybrid	IMT
6/9	0.35	0.17	0.17
12/15	0.89	0.66	0.66
18/21	1.63	1.48	1.39
30/33	3.95	4.12	3.80
42/45	7.89	8.07	6.87

4.5.3.5 Model Definition

This test determined the standard definition times for those systems using the automated stereomodel definition methods (i.e. those coded as "570" and "660"). Since the process does not require operator's participation, the time for resolving all the 12 stereopairs of the Edmunston block was determined through a batch computational procedure after the triangulation had been performed. Table 4-9 shows the per unit model definition times for all the systems excepting the operator-assisted system (IMT_571). In practice, systems employing the operator-assisted method combine the definition and restoration operations as one process, which is usually performed when digitizing of a stereomodel is requested. Thus, definition time for this system was not determined here but at the model restoration stage which is described next.

Table 4-9: Model resolution time per stereopair (minutes)

	System							
	IMT_000	IMT_571	HT_000	HT_570	HT_660	BT_000	BT_570	BT_660
minutes	0.03		0.03	0.09	0.11	0.04	0.09	0.11

4.5.3.6 Model restoration

Two situations exist here. For systems using the automated definition technique, the model restoration operation involves (1) the performance of interior orientation in case the pair of photographs have been removed from the instrument stages, (2) the retrieval of the model transformation matrices from the data file, (3) and downloading to the real time processor (P2) and to the digitizing unit. Since the model information at this stage is the same irrespective of the system involved, the time per model for this operation is composed of the time it takes the operator to perform the interior orientation plus the time it takes to transfer all information to the various units. And since the standard time for interior orientation had been determined earlier, the time per model for the data transfer operations was averaged from the time for the 12 stereopairs of the Edmunston block and added to the interior orientation time as shown in table 4-10. Note that the time for data transfer includes the time for fetching the data from memory, and transfering them to the various units including the real time processor P2 (which restores the model and moves the floating mark from one end of the model to the other to ascertain that the model is well restored and then returns an "okay" message to the host).

On the other hand is the system employing the operator-assisted restoration (i.e. system IMT_571). For this, the model restoration process involves a repetition of all the orientation and measurement operations to determine the image-model and model-object space transformation parameters. However, since the standard times for I.O, R.O, and measurement operations have been previously established, only the time required for the computation of the absolute orientation parameters and performing the restoration as described before, need to be determined. And since this is basically a computational operation, a batch processing of the 12 stereopairs was timed and averaged for the per model time for this aspect of the restoration. The average time was then added to the times established for the other operations involved to give the time per model for the operator-assisted restoration. This is also shown in table 4-10.

Table 4-10: Model restoration time per stereopair (minutes)

	System							
	IMT_000	IMT_571	HT_000	HT_570	HT_660	BT_000	BT_570	BT_660
Process time	3.72	15.42	3.72	3.72	3.72	3.72	3.72	3.72
Operator's time	1.56	11.11	1.56	1.56	1.56	1.56	1.56	1.56

4.5.3.7 Systems' spatial accuracy assessment

It is necessary to note at this point that the time for digitizing was not considered in this study since in practice, digitizing times depend on factors such as type of data being digitized, the level of detail, operator's skill and others, which are independent of the type of configuration being used and are therefore not relevant to the present study. Accuracy checks at the digitizing stage represent a way to monitor the overall accuracy of the entire system. The test was conducted computationally by applying the derived transformation matrix to the check points and the between model check points. This procedure eliminated the bias inherent in repeated operator measurements. The digitizing accuracy of each system was ascertained as the root mean square error (RMSE) of the absolute differences in coordinates at a number of check points and at those points common to two or more stereopairs. Since the check points have precisely known object space coordinates, they provide a good check on the absolute accuracy of each system. The values obtained at the absolute check points (plan and height) are shown in Table 4-11.

Table 4-11: RMSE of Checkpoint differences (in centimetres) for 12 stereomodels

	System							
	IMT_000	IMT_571	HT_000	HT_570	HT_660	BT_000	BT_570	BT_660
Plan	13.7	13.0	13.2	13.3	13.3	13.5	13.4	13.3
Height	19.6	20.1	20.1	20.4	20.3	20.9	19.4	19.9

Table 4-12: RMSE of Between-model Checkpoint differences (in centimetres)

	System							
	IMT_000	IMT_571	HT_000	HT_570	HT_660	BT_000	BT_570	BT_660
Plan	12.7	12.4	13.3	13.1	13.3	13.1	12.9	13.0
Height	23.9	24.3	24.7	24.5	23.9	21.9	21.2	21.4

Moreover, the coordinate differences for those points appearing in two or more stereomodels provide an indication of the amount of edge-matching needed for a feature that crosses model boundaries for each system. Table 4-12 shows the between-model RMSE values for both plan and height. In principle, both the absolute check point differences and the between model differences are criteria of reliability of the systems, and they may be used as constraints in a linear programming optimization strategy. However, to reduce the number of constraints, a dilution of both the absolute check point differences and the between- model check point differences was computed by taking the average of the L2 norms of the entries in tables 4-11 and 4-12. This is shown in table 4-13. The diluted differences will be used as the coefficients of a constraint function as will be discussed later in the analysis and selection of the optimum design.

Table 4-13: Diluted differences (cm)

	System							
	IMT_000	IMT_571	HT_000	HT_570	HT_660	BT_000	BT_570	BT_660
(cm)	25.49	25.61	26.05	26.07	25.81	25.20	24.20	24.49

4.5.3.8 Summary of Test Statistics

Table 4-14 shows the summary of the test statistics. The system time per model was derived by aggregating the various process times from the tables presented earlier; this is shown in row 1 of table 4.14. Multiplying these values by the corresponding values in table 4-3, we obtain the system cost (hardware + software) per model for each system as shown in row 2. In row 3, the aggregate of operator time per model is shown, and multiplying the entries in that row by the OPR (27.78 cents), the operator's cost per model in dollars is shown in row 4. Adding corresponding entries in rows 2 and 4, we obtain the total cost (system + operator) per model shown in row 5.

Table 4-14: Summary of test statistics

	System							
	IMT_000	IMT_571	HT_000	HT_570	HT_660	BT_000	BT_570	BT_660
System Time/ model (min.)	25.88	37.55	22.52	22.58	22.60	22.35	22.40	22.42
System Cost/ model (dollars)	1.119	1.621	0.999	1.015	1.010	0.976	0.990	1.002
Operat. Time/ model (min.)	12.67	22.22	10.36	10.36	10.36	10.36	10.36	10.36
Operat. Cost/ model (dollars)	3.520	6.173	2.878	2.878	2.878	2.878	2.878	2.878
Total Cost/ model (dollars)	4.639	7.794	3.877	3.893	3.888	3.854	3.868	3.880

4.6 Data analysis and optimal system selection

The data in row 5 of table 4-14 provides empirical approximate estimates of the coefficients of the decision variables in the cost function. Substituting these values for U_i in eqn. 4-4, we may write the approximate objective function (eqn. 4-4) as:

$$Z = 4.639X_1 + 7.794X_2 + 3.877X_3 + 3.893X_4 + 3.888X_5 + 3.854X_6 + 3.868X_7 + 3.880X_8 \qquad (4\text{-}12)$$

Using the data in rows 1 and 3 of table 4-14, and those in table 4-13 in that order, as values for C_i, the approximate constraint functions (eqn. 4-7) may be explicitly written as:

$$\begin{aligned}
25.88X_1 + 37.55X_2 + 22.52X_3 + 22.58X_4 + 22.60X_5 + 22.35X_6 + 22.40X_7 + 22.42X_8 &\le q_1 \\
12.67X_1 + 22.22X_2 + 10.36X_3 + 10.36X_4 + 10.36X_5 + 10.36X_6 + 10.36X_7 + 10.36X_8 &\le q_2 \\
25.49X_1 + 25.61X_2 + 26.05X_3 + 26.07X_4 + 25.81X_5 + 25.20X_6 + 24.20X_7 + 24.49X_8 &\le q_3 \\
X_1 + X_2 + X_3 + X_4 + X_5 + X_6 + X_7 + X_8 &= q_4 \\
X_i \ge 0,\ i = 1, ---, 8 &
\end{aligned} \qquad (4\text{-}13)$$

where the constant terms are the values of the constraints i..e.

q_1 is the total time available for the project

q_2 is the amount of operator time in minutes that can be provided

q_3 is the maximum size for the sum of all the misclosures

q_4 is the total number of models desired to be treated

The task remains to determine the values of the system or decision variables in the cost function such that the cost of executing a project, given limited resources, is the minimum possible. This is the optimization problem which, using our empirical model, may be mathematically stated as:

minimize:

$$Z = 4.639X_1 + 7.794X_2 + 3.877X_3 + 3.893X_4 + 3.888X_5 + 3.854X_6 + 3.868X_7 + 3.880X_8$$

subject to:

$$\begin{aligned}
25.88X_1 + 37.55X_2 + 22.52X_3 + 22.58X_4 + 22.60X_5 + 22.35X_6 + 22.40X_7 + 22.42X_8 &\leq q_1 \\
12.67X_1 + 22.22X_2 + 10.36X_3 + 10.36X_4 + 10.36X_5 + 10.36X_6 + 10.36X_7 + 10.36X_8 &\leq q_2 \\
25.49X_1 + 25.61X_2 + 26.05X_3 + 26.07X_4 + 25.81X_5 + 25.20X_6 + 24.20X_7 + 24.49X_8 &\leq q_3 \\
X_1 + X_2 + X_3 + X_4 + X_5 + X_6 + X_7 + X_8 &= q_4 \\
X_i \geq 0,\ i = 1, ---, 8
\end{aligned}$$

Using the simplex algorithm (Pierre 1969) values may be obtained for the system variables if the constant terms (the resources and requirements i.e. q_1, q_2, q_3, q_4) are specified. In this study, the values for total project time and operator time constraints are fixed at an integral multiple of the biggest coefficient in the respective constraint function in order to ensure feasible solution of the optimization problem. Thus, to specify resources sufficient for a project comprising 40 stereomodels (i.e. $q_4 = 40$), we must have the numerical values of the constraints as:

$$q_1 = 37.55 * 40 = 1502 \text{ minutes}$$

$$q_2 = 22.22 * 40 = 889 \text{ man-minutes;}$$

For the accuracy constraint, we initially select an integer multiple of an accuracy value near the average and then explore the effect of relaxing the accuracy requirement on the solution.

Thus, choosing a value of 25 cm (from the coefficients of the third constraint function, we have:

$$q_3 = 25 * 40 = 1000 \text{ cm}$$

These constraints will be referred to as the basic constraints, to distinguish them from the external constraints which will be imposed in order to ascertain the behaviour of the system under certain circumstances.

The task of identifying the optimum system design is accomplished in four phases as follows:

First phase: we successively constrain all systems except one to zero (external constraint); in other words, all the 40 stereomodels are directed to be processed by only one system at a time. The minimum cost for executing the project on each system is noted. Figure 4-12 shows the distribution of the minimum project cost. Note that with the imposition of this type of external constraint, the basic constraints become irrelevant and in fact ineffective since in reality, only one system is operational at one time as specified by the external constraints.

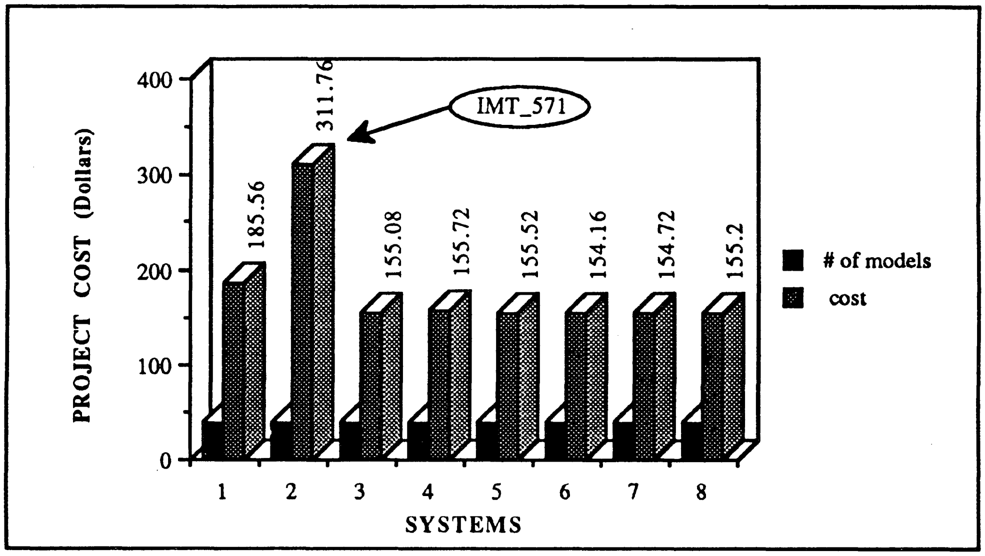


Figure 4-12: Variation of cost value with all 40 stereomodels processed on one system

It is obvious from Figure 4-12 that software designs X_1 and X_2 incured the highest project cost. Therefore they may be rejected as not being optimal. However, they are not eliminated outright from the process for reason of convenience for subsequent analysis. Furthermore, systems X_3 to X_8 are seen to incur about the same minimum project cost, implying that the optimal software architecture is contained in this group; thus they have to be subjected to further tests.

Second phase: we then respecify the external constraint by requesting that each system handles at least one stereomodel while they all operate simultaneously. This implies that the optimization process can assign any number of models to any of the systems in a way to achieve minimum project cost; however, each system must have at least one model assigned to it. This external constraint is designed to show which system will have the largest allocation as indicated by the values of X_1, X_2, ..., X_8 respectively from the solution. Figure 4-13 shows the distribution of these decision variables, marked as the optimality score in the figure. Note that in this case the basic constraints are now in effect, though restricted by the external constraint. From Figure 4-13, it is seen that only systems BT_000 and BT_570 have allocations above the minimum dictated by the external constraint. This seems to suggest that the optimal system solution resides within the borders of bundle technology. Furthermore, we see that out of the total of 40 models available for processing, 23.68 are allocated to system BT_000 and 10.32 to BT_570. Are we then to conclude that we have found the optimum system ? Certainly not, because the external constraints are still in place, and we are yet to explore the effect of relaxing or tightening the accuracy constraint on the solution.

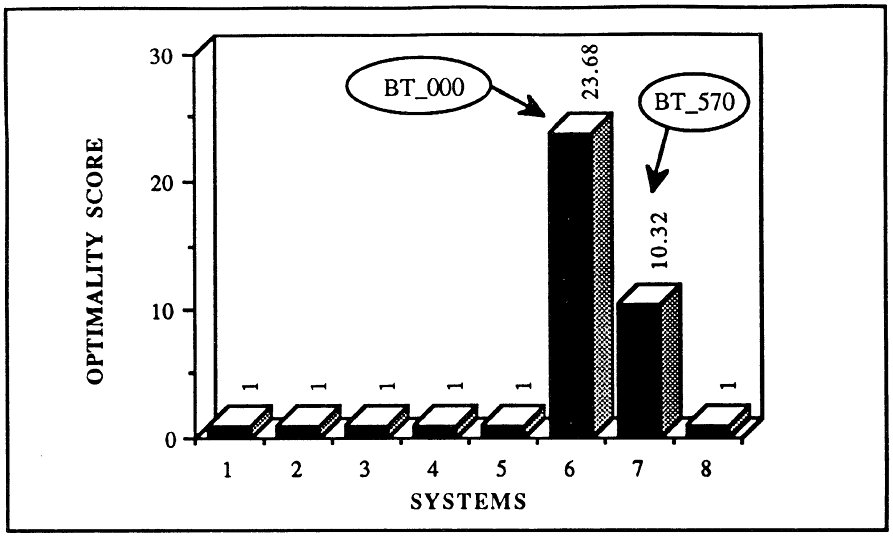


Figure 4-13: Optimality distribution when all systems are constrained to not less than 1 unit each

Third phase: we remove the external constraints from the process so that each system may be allocated a proportion of the project (in terms of number of stereomodels) only according to its characteristics. Figure 4-14 shows the distribution of the number of stereomodels assigned to each system for processing.

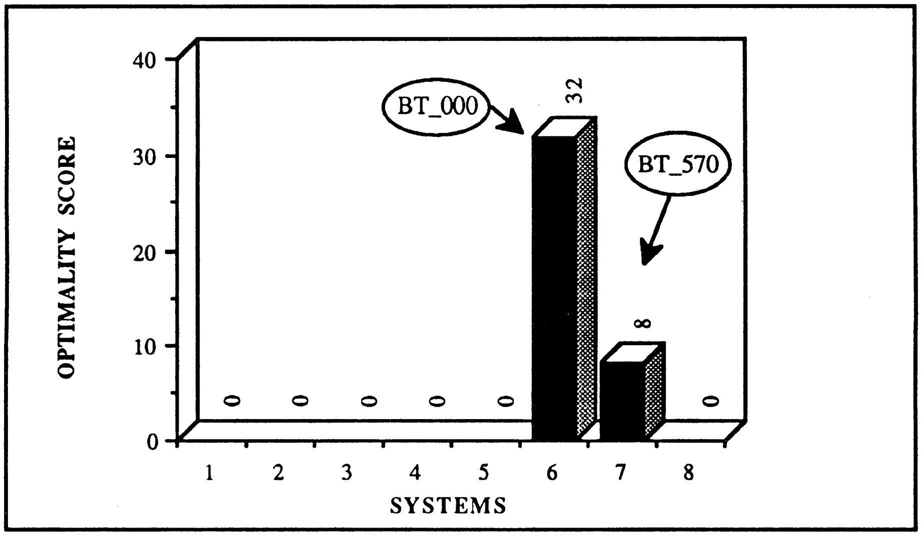


Figure 4-14: System optimality score with total time, operator time and accuracy constraints.

From here, it is seen that all systems except X_6 and X_7 have no job assigned to them. Therefore, we have enough reason to reject these designs as being unoptimal. Also, we observe that out of 40 stereomodels, 32 are assigned to X_6 (BT_000) and 8 to X_7 (BT_570). Because systems BT_000 and BT_570 have survived all the elimination tests conducted, we may conclude that these two designs are optimal and that if we were to select a design based on the initially specified basic constraints, system BT_000 would be a better choice. However, we have to move the test point away from the intial accuracy specification in other to ascertain the behaviour of each system under varying accuracy requirements.

Fourth phase: We vary the accuracy constraint away from the initial value and note the optimality score of each system. The responses of the two systems are shown in Figure 4-15. It is interesting to note that as the accuracy requirement becomes less stringent (Figure 4-15), system BT_000 becomes the predominantly optimal design; however as the requirement gets stricter, its optimality score drops off in favour of system BT_570.

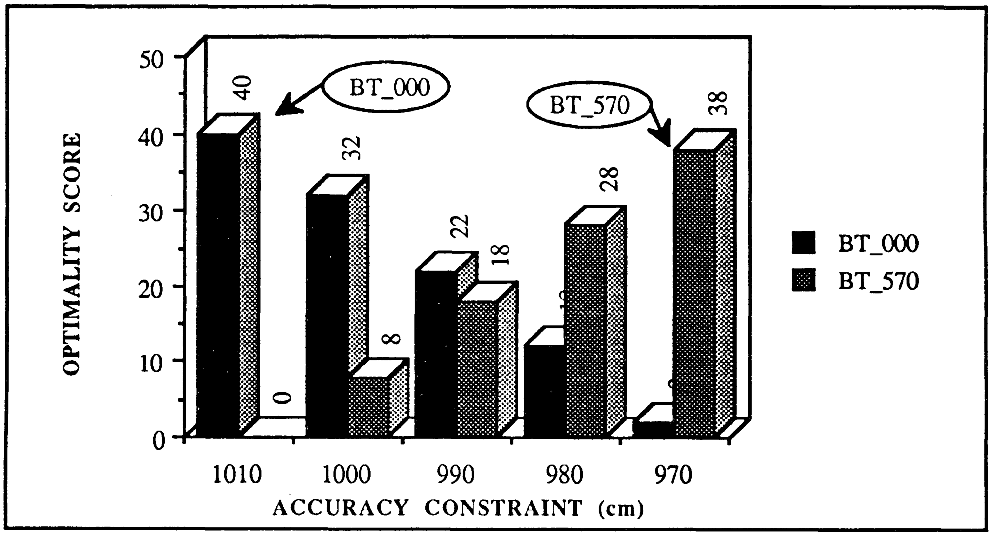


Figure 4-15: Variation of the Optimality scores for systems BT_000 and BT_570 with variation in accuracy requirement

4.7 Summary and Conclusions

From figures 4-12, 4-13, 4-14 and 4-15, we state conclusively that there are differences in the performance of the various software configurations of a photogrammetric workstation and that, depending on the chosen procedure, significant savings in both system time and operator involvement can be realized by using either system BT_000 or BT_570. Furthermore, when high accuracy of spatial information is a critical requirement, software design BT_570 (Figure 4-16) will be the most appropriate architecture for the workstation. However, when lower accuracy is sufficient, the design BT_000 (Figure 4-17) will be the optimum configuration for the workstation.

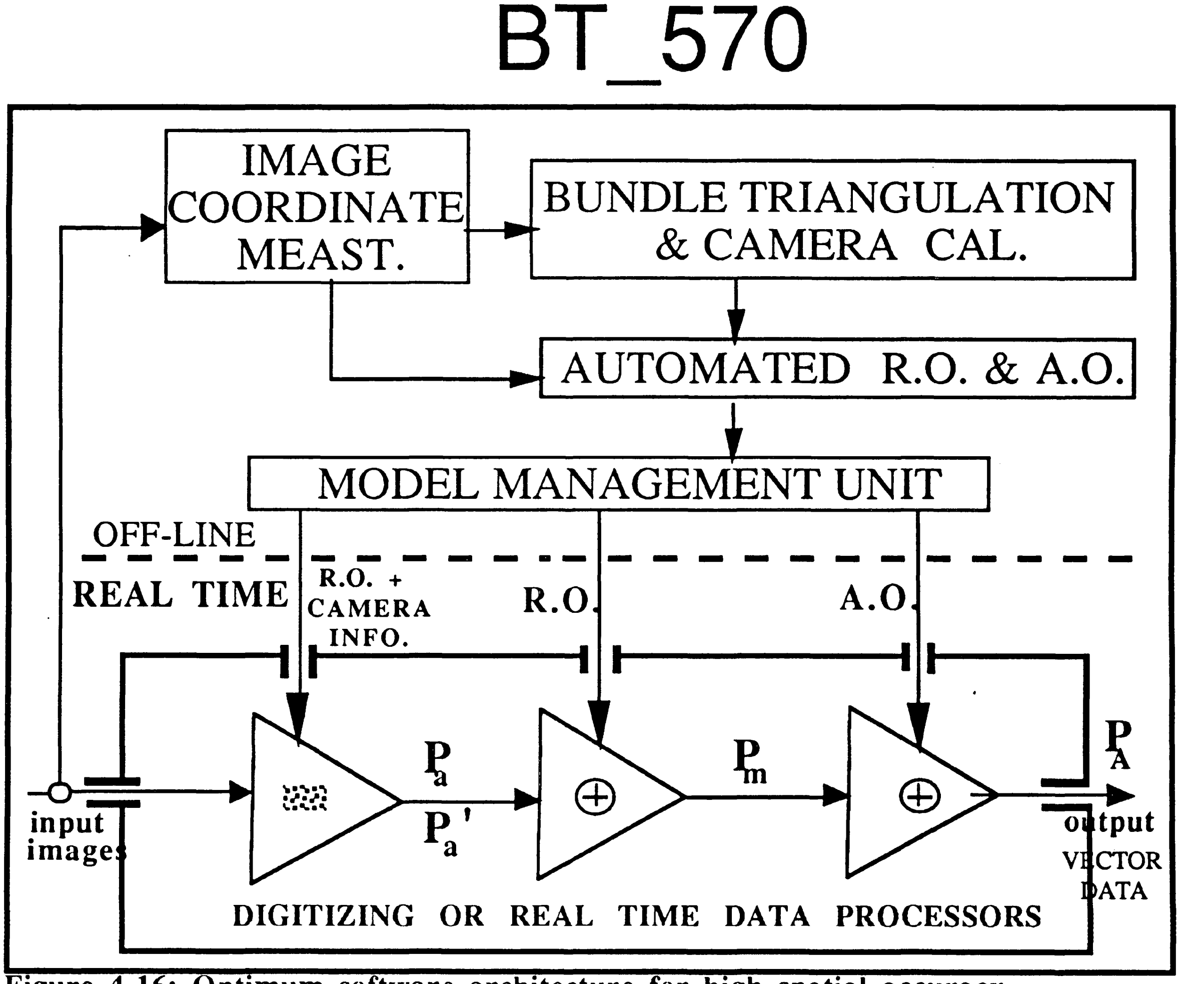


Figure 4-16: Optimum software architecture for high spatial accuracy

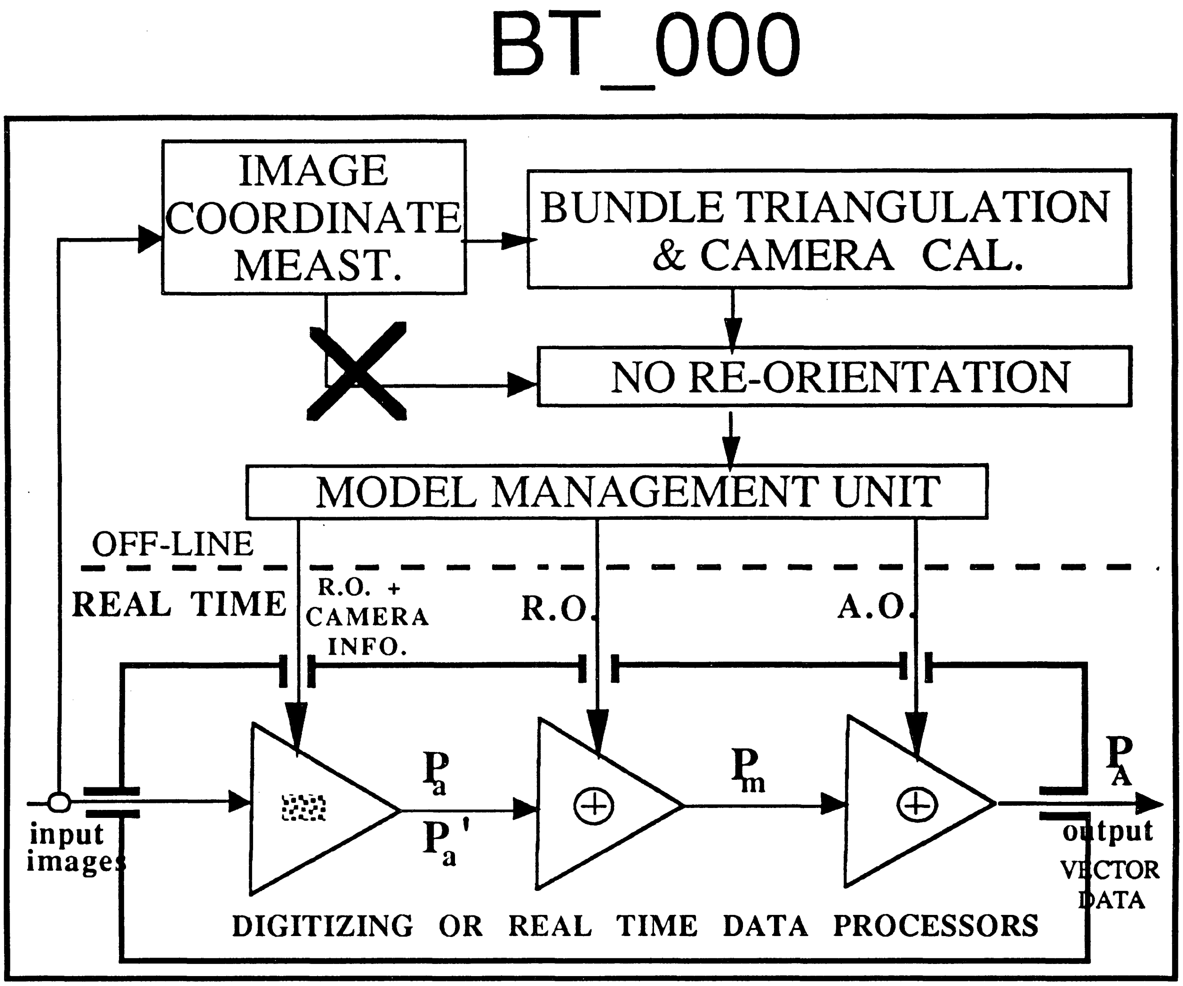


Figure 4-17: Optimum software architecture with relaxation of accuracy

These conclusions are valid for the stated criteria and within the limitations of the experiment conducted here. Naturally, as demonstrated in Figure 4.15, changes in user requirements which are translated to changes in the constraints on the problem may lead to a different conclusion. In this study, the system design BT_570 meets the requirements for minimum project time, high automation content and as well high spatial accuracy of the vector data. Consequently, it has been chosen to configure the Kern DSR-11 analytical plotter to a fully equiped workstation, capable of producing vector data for GIS applications.

Yet, a second major objective of the present study is to achieve the integration of the photogrammetric workstation into the CARIS GIS for real time operational communication. This matter is addressed in the next chapter. The developmental processes and operational mechanisms of a passive interface between the workstation and the CARIS GIS are described. It is noted at this point that the photogrammetric system is from now on referred to as the workstation.

Chapter 5

Interfacing the Photogrammetric Workstation to the CARIS GIS

This chapter describes the problems involved and the facilities and techniques which have been used in this study to achieve an interfacing of the photogrammetric workstation (the source) with the CARIS GIS. To begin with, section 5.1 focuses on the effect of incompatibility of data formats from one system to another on the complexity of the interfaces required to achieve integrated operation. Section 5.2 describes the server module which is available in the CARIS GIS for connection to a data source. In section 5.3, a data collection module developed in this study and the methodology of its partitioning into parent and child driver modules are described, while section 5.4 presents the theory of the parent and the child process creation in the VMS environment. Section 5.5 highlights the concepts of the MailBox (a buffer unit) which serves as a data channel between the parent and the child digitizing modules. The chapter concludes in section 5.6 with the description of the menu-driven user interfaces by which the user interacts with the system during a digitizing application.

5.1 Effects of Inter-Component Data Disparity on the Development of Integrated Systems

Practically, the objective of systems integration is to establish an inter-process communication channel for sharing and exchange of information either between systems from the same vendor or from different vendors; more often it is the latter. Because of the disparity in data structures, a greater per centage of the integration effort is spent on developing data interpreters between such components [Hodgson et al 1989, Ramirez 1989].

Generally, the task of systems integration may involve one of two situations depending on the type of integration desired - serial operation or parallel operation. For a given style of integration, the type and complexity of the required interfaces are determined by the data format circumstances between the units involved as shown in Figure 5.1. The effects of data incompatibility on the integration for both serial and parallel operations are examined in the following two subsections.

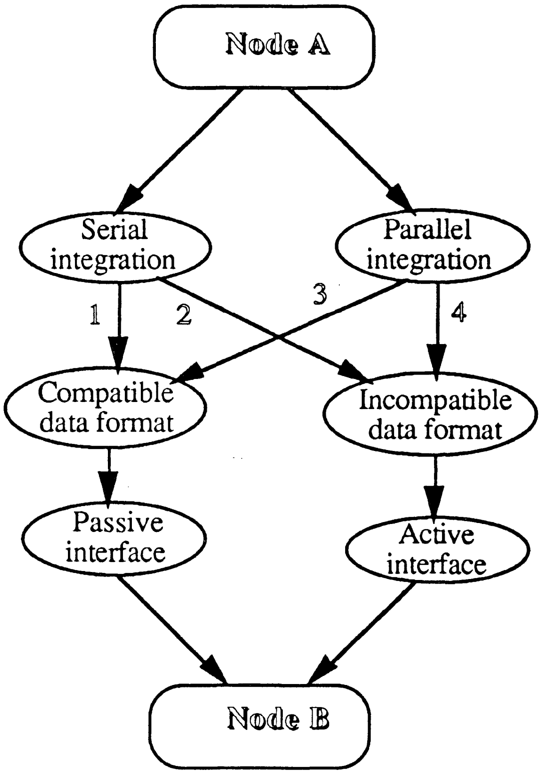


Figure 5-1: Optional paths for system integration

5.1.1 Effect on integration for serial operation

From a technical viewpoint, integration for serial operation is a relatively simple process since the operations of the units involved are separated in time; therefore, turnaround time is often not a critical factor in the communication process (i.e. may be hours, days or even months). Consequently, the required communication interfaces may be simple or complex depending on the data conformity between system nodes (see Figure 5-1). If the system units have concordant data formats (case 1, Fig. 5-1), a passive interface is all that is needed in the communcation facility to effect the required integration (Figure 5-

2a). The word "passive" is used here in the sense that the interface serves only as an information channel which is not required to perform any translation. For example, the interface could simply be a datafile into which one system writes all the information to be communicated using the mutually understood format. Upon its termination, the cooperating system unit reads the file and vice versa.

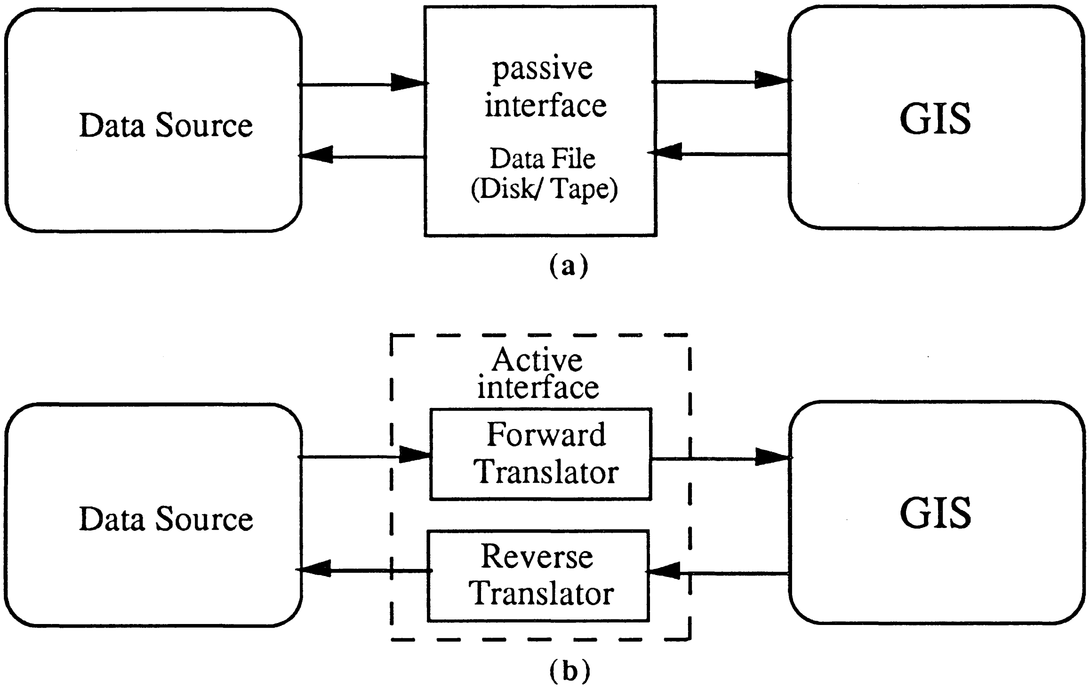


Figure 5-2: Interfacing of two system components for in-line operation. (a) compatible data structure requires passive interface, (b) incompatible data structure requires active interface (translators)

When the data structures of the system components are incompatible (case 2, Figure 5-1), integration for serial operation requires a more complex interface which has the knowledge and capability to translate the information from the format of the source unit to that of the GIS for example. In this case, it must be an active interface because its role is not only to allow the passage of information but also to decode, interpret and then transform the information into the format acceptable to either system. For bidirectional communication, the active interface will have two such translators, one for each direction of communication (Figure 5-2b). However, since turnaround time is not a critical factor in such systems, the translation stages may be as many as desired and any suitable communication means may be employed as long as the information is ported from unit to unit.

To elaborate on the ideas in the previous paragraphs, imagine a hypothetical organization concerned with digitizing spatial data from stereomodels. This company supposedly, has adopted the "blind" digitizing mode in order to speed up the digitizing process - i.e. the digitizing and editing operations are done at separate workstations, presumably utilizing third party digitizing and editing software packages. If the company wants to integrate its activities such that the editing commences when the digitizing operation terminates, such integration is serial and the task is to get the collected data from the digitizing workstation to the editing workstation. Obviously, if the two stations have the same data format, then the interface that this company needs is a datafile into which the digitizing software writes all the collected data. This file may then be transported by a number of communication means to the editing station for processing by the editing program. The communication required may be as simple as swapping of data discs across computer terminals or merely inputing of the datafile name when the editing program is started, or it may be as elaborate as sending the data over a telecommunication route, or the data disc through a postal agency etc. Yet, in the event that the data structures used at the two stations are incompatible, then this company needs a translator to be placed somewhere along the communication route. Certainly, another software package which understands the data formats of the operating packages has to be acquired to do the translation. Since this company is not pressed for time, the translation can take as long as it requires. Furthermore, if this company wants to return the data disc for some correction or updating for example, then it must integrate for bidirectional communication, and this means two translators are required. However, as fragmented as this company's integrated system seemingly is, it nevertheless shows that data structure differences lead to complex and more expensive system interfaces. Moreover, this example, though hypothetical, truely portrays the realities of the data transfer situation between most of the modern mapping workstations using CAD/DBMS packages and the GIS application environments (Hodgson et al 1989, Parker 1990).

5.1.2 Effect on integration for parallel operation

The task of integration for parallel operation involves synchronizing the operations of the system units so that they share information in real time (cases 3 & 4, Figure 5-1). In this case, the situation becomes compounded by the fact that response time is now a critical factor in the process. The rate of information transfer must match the pace of operation, particularly in an interactive operation involving the human operator. If the two units involved in the parallel integration possess compatible data structures (case 3, Fig. 5-1), then similar to the serial connection process, the integration is free from data format problems, and a passive interface which is consistent with real time operation provides the logical choice. For example, for systems operating in the same environment such as digitizing with real time editing, such a passive interface could well be a data record (or a datablock for a large volume of data) which is accessible to both system units for writing into and reading from (Figure 5-3a). Yet, it is important that the writing and reading operations by the cooperating programs be coordinated to avoid collisions. On the other hand, if the components have data structures which are mutually discordant (case 4, Fig.5-1), then the problem of parallel integration becomes convoluted in that the interface module must perform two translations to achieve the desired link. Firstly, it must understand and translate the data source into the form understandable to the GIS. Secondly, it must understand and be able to translate data and commands from the GIS to the source unit (Figure. 5-3b). And still, the translation has to be done in real time, implying that an active interface consisting of two translators, and able to operate within a short time cycle is inevitable. Of course, the translators must know the two opposing data structures in their entirety, otherwise they cannot achieve an error-free translation.

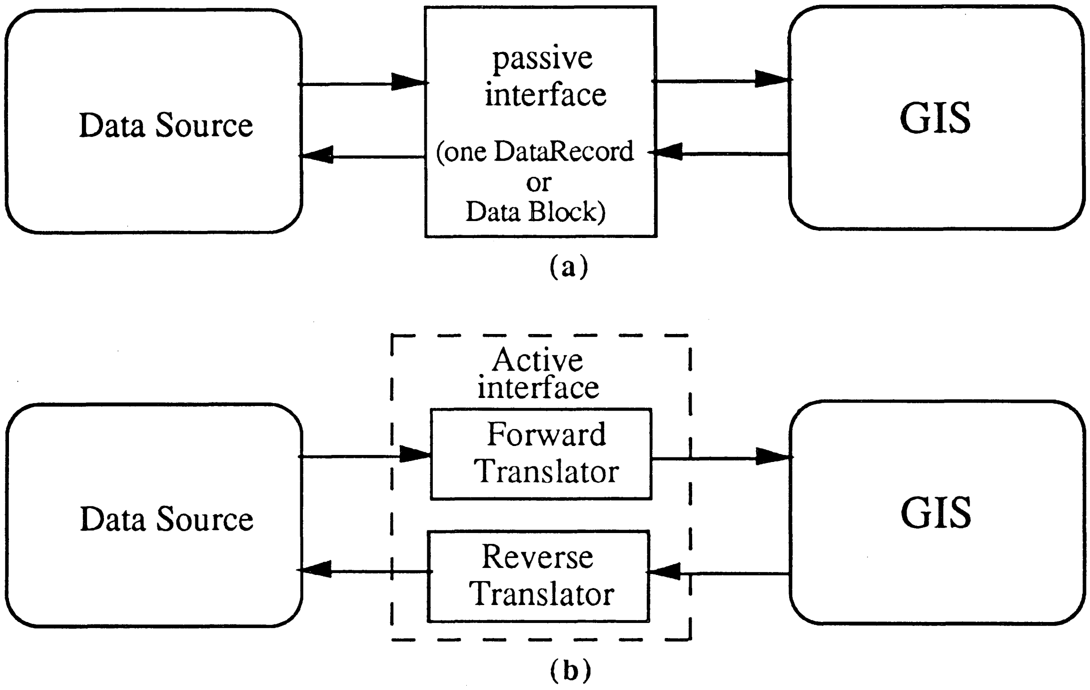


Figure 5-3: Interfacing of two system components for parallel operation. (a) compatible data structure requires passive interface, (b) incompatible data structure requires active interface (real time translators)

Evidently, data format incompatiblity makes integrated systems complex and expensive. Again, let us return to the hypothetical example in which the company is considering its options for solving the data format problem in its set up. Were it possible for this company to open its software packages and change their data formats so they become compatible, it would logically prefer to do this to eliminate the use of translators which make its system complex, costly, and predictably slow. However, breaking into the software is not a practicable option since software vendors do not generally provide source codes of their products. Furthermore, discarding existing software packages and purchasing new compatible ones is an attractve but expensive proposition which the company probably cannot afford. Neither is the option of re-organizing to develop its packages in-house a feasible one. Obviously, the only practicable way to get information across programs with differing data formats is to employ translators, not withstanding the associated imperfections and inconveniences for real time applications. This episode demonstrates that data structure incompatibility is a big obstacle to the implementation of integrated systems particularly when parallel operation is desired.

It has been tacitly assumed in the foregoing discussion that none of the system units involved in the parallel integration denies access to its datafiles during operation, or is incapable of communicating with an external program in real time. However, in order to provide enough background for understanding all the associated problems encountered during this study, consider yet another dimension to the company's problem, a situation where one or both of the system units to be integrated do not accept communication with an external program during execution or deny real time access to their data files. This may happen for instance when a digitizing program at a photogrammetric workstation opens a data file which cannot be accessed by an external program while digitizing is in progress, or when a GIS denies access to an external program that attempts to write or read its data file. In such situations, integration for real time operation is not possible, and if it must be implemented, new digitizing modules and translators have to be purchased or developed in-house. The point being made here is that, apart from data incompatibilitty, "closed" operation of individual programs is also a problem when assembling systems for simultaneous operation.

5.1.3 A substitute digitizing package for MAPS200

Against the background provided by the foregoing discussions, the problems encountered in the effort to integrate the DSR-11 into the CARIS GIS environment, and the technique used to solve these problems are now briefly highlighted. Remember however, that the photogrammetric system implementation discussed in the previous chapter only transforms the analog photographs into digitizable stereomodels. The actual digital data collection is usually performed using a digitizing program. Therefore, in the discussions that follow, this digitizing program will often be referred to as the source unit or the host program.

The digitizing program supplied with DSR-11 is called MAPS200, which is similar to a computer-aided drafting (CAD) package, and its purpose is for the collection and editing of digital data from stereomodels on most of Kern's photogrammetric stereoplotters. By its design, this package can output to the Kern GP1 plotter, and optionally to a user defined file (Kern DSR-11 documentation 1987). The file format is the Kern CAM format which consists of a listing of digitized coordinates, interspersed with a host of "pen-up" and "pen-down" strings. Furthermore, the colour, line type and size selections are written into the file, also in the Kern coding formats. On the other hand, the CARIS GIS format is such that a spatial object is recognized either as a point, a line, or an areal element (CARIS documentation 1990). An element's feature code and its list of coordinates are stored as a unit in the CARIS data structure. As discussed in the previous sections, given these different data formats and the need to coordinate the user-interfaces of the two modules, the integration of these two systems for real time operation involves the use of bi-directional translators for data and command transfer. Figure 5-4 shows the configuration of such a system.

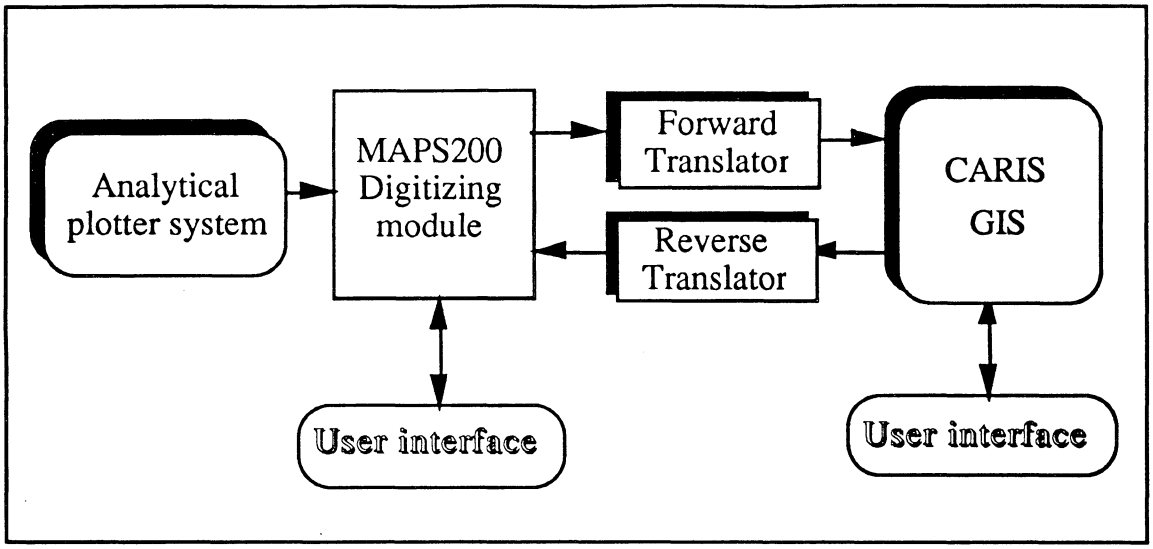


Figure 5-4: Integration of MAPS200 into the CARIS GIS

Nevertheless, the configuration shown in Figure 5-4 is impractical for real time operation for the following reasons: (1) the MAPS200 package does not allow an external program access to its data file while digitizing is in progress, and (2) it does not accept any data or commands from an external program (i.e. it is a closed program), (3) the translators needed to implement the configuration do not exist. Obviously, interfacing the MAPS200 package as it is to the CARIS for simultaneous operation is impossible. If translators were available, the best we could achieve was a serial integration. However, since the aim of this study has been to interface for real time operation, a solution to these problems was required. The methodology adopted here was to avoid the use of MAPS200 altogether, and to develop a more responsive digitizing module which could be initiated from the CARIS GIS environment. This approach takes advantage of the fact that the CARIS GIS possesses all the editing and CAD functions of MAPS200 and also provides a server interface to which an external program may be attached for real time operation and transfer of data, thereby eliminating the need for an active interface involving translators. This has been facilitated by the multitasking capability of the VAX operating system (a feature of the modern minicomputer systems),which allows a number of separately compiled program modules to operate in parallel (VAX documentation 1984, Hoppe 1991). One possible configuration of this system is shown in Figure 5-5.

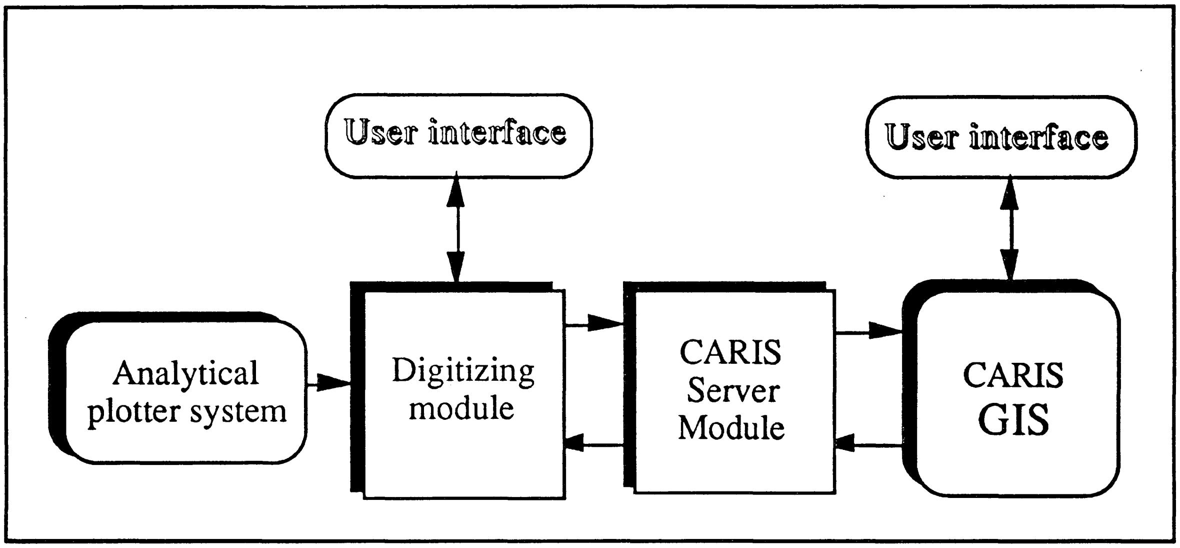


Figure 5-5: A configuration for the Integration of the DSR-11 into the CARIS GIS

Although, the configuration shown in Figure 5-5 is simple on the design board, its practical realization is hampered by a number of operational problems such as the limited stack memory made available to a single program by the VAX operating system, the limited number of allowable concurrent system processes that may be initiated from a single program, and the synchronization of the many user interfaces involved in the various operations. To circumvent these problems, the new digitizing module (discussed later in section 5.3) has been partitioned into **parent** and **child** submodules linked together by a buffer unit (the mailbox), whose design and implementation are explained in section 5.5. While the parent cooperates with the CARIS GIS, the child and its associated subprocesses interact with the analytical plotter system. Thus, introducing this concept into Figure 5-5, the resulting configuration which has been implemented is shown in Figure 5-6. The concept of process creation for either synchronous or asynchronous operation, which has been instrumental to the practical realization of this system is described in section 5.4. Prior to that, the major components of the new system, namely the server interface and the digitizing unit are discussed. Note that the description of the server interface given in the next section is adapted from a USL's publication on the server interface as described by Reeler (1990).

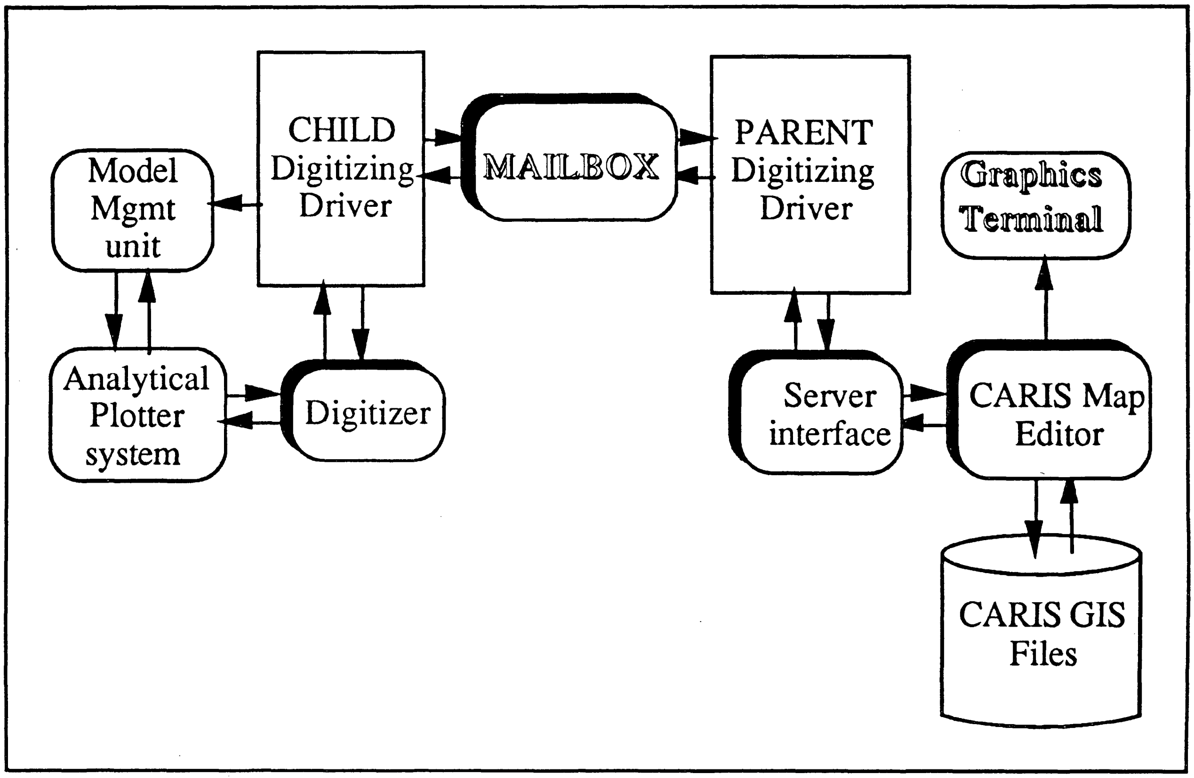


Figure 5-6: Implemented system configuration for the Integration of the DSR-11 to the CARIS GIS

5.2 The CARIS Server Interface

The CARIS server interface has been developed by Universal Systems Limited (USL) to enhance the capability of interfacing the CARIS GIS to external software packages (Reeler 1990). This facility enables users to interactively interface other program modules in order to have access to the CARIS data files for real time graphics display, editing, storing and retrieving of data. The CARIS server module provides software developers with many powerful capabilities which include:

1. storage and retrieval of CARIS GIS data directly using simple subroutine calls.
2. access to all of the graphics functions of both CARED and CARMAN modules, allowing users to manipulate and display their graphics
3. access to database management systems, such as INGRES
4. access to continuous database functions
5. ability to incorporate topology into user applications
6. ability to customize the GIS applications to satisfy user requirements

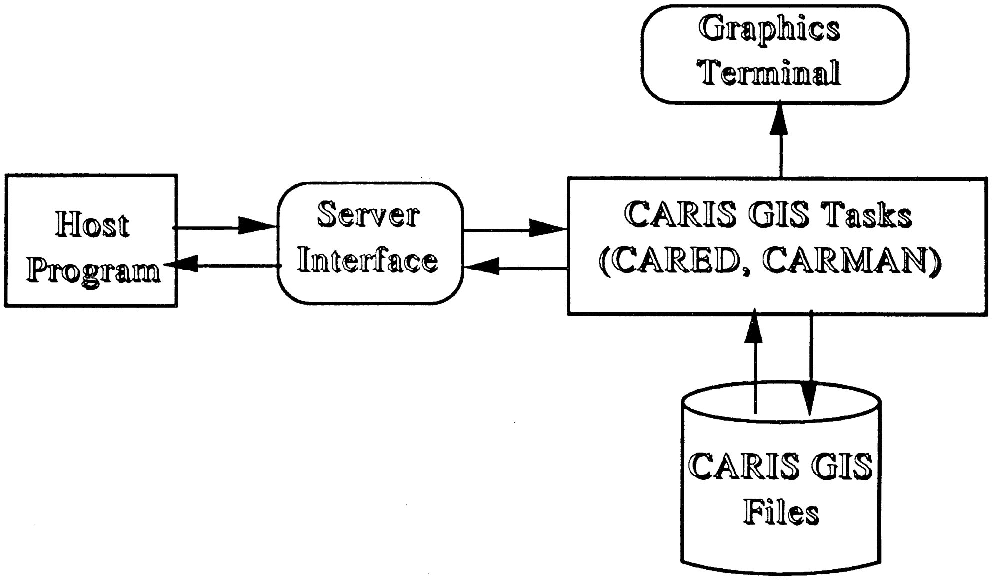


Figure 5-7: The Server Interface configuration.

Technically, the server interface is designed such that its operation is transparent to the host task (see Figure 5-7). It communicates with the host program through a number of functions and subroutines which must be initialized by the host program. The host software package can send commands to and read data back from a CARIS GIS module such as CARED (CARIS EDitor). For example, to store symbols in a CARIS GIS file, the host program sends a **SYAD** command to CARED through the server module, followed by the responses to the CARED prompts, usually displayed when a symbol is added (feature code, size, font, key and position). This procedure can be performed to add other types of data (lines, texts, etc.). The host program can also retrieve information about a particular graphical item in the database. Typically, **CARED** commands such as **LSPT**, **LSSE** and **MHED** can obtain information about the data stored in a CARIS GIS file. For example, **LSPT** allows the user to point at a displayed graphic element, and then makes available to the host program the key, descriptor, super-descriptor (if applicable) and the **X**, **Y**, **Z** coordinates (ground, screen, and latitude, longitude).

Basically, the operation of the interface is centered around calls to the following routine:

Routine name:	**Hstnit**
Purpose:	initializes the server interface. It is used by the host program at the start of operation and before any task is requested.
Usage:	**call Hstnit (Rcode)** if **Hstnit** executed without errors, Rcode will return with a value of 0; if an error occured, Rcode will return with a value of 1.
Routine name:	**Hstwrt**
Purpose:	sends information to the CARIS GIS server task from host program. Once the server has been initialized, this routine may be used to download all data and instructions to the CARIS GIS task.
Usage:	**call Hstwrt (Messtg, CtrlZ, Response, Rcode)** **Messtg** contains the information which the host program sends to the CARIS GIS task, e.g. LSPT, X, Y, Z coordinates etc. Messtg is a character string variable up to 132 characters long. **CtrlZ** is a logical variable which may be used to indicate the end of file. **Response** is a variable which contains a message from the server task which may be useful to the host program. The response may contain the string "OK" indicating that the server is expecting a certain character string. **Rcode** is an error monitoring variable which the host program tests to detect error conditions.

The host program can read data from the CARIS GIS files through calls to a number of other routines. These include:

Routine call	Return item
call GtsKey (Key, Rcode)	Key
call Gtsgxy (Xg, Yg, Zg, Rcode)	X, Y, Z ground coordinates
call Gtsllh (Lat, Long, Ht, Rcode)	geographic coordinates and elevation
call Gtssxy (Xs, Ys, Zs, Rcode)	screen or internal coordinates
call Gtsdes (Desc, Deslen, Rcode)	descriptor of the graphic element

The server interface, provides a powerful means for an external program to drive the CARIS GIS task modules. However, from the author's experience, this package needs to provide some kind of "software switch" which will enable the user to change from the host program's user interface to the CARIS GIS user interface by pressing a function key or entering some character code which will hold the host program while the user performs some quick editing via the graphics terminal's keyboard. This facility is useful, for example, during a data capture and editing session when the operator may want to perform some interactive editing (e.g. adding a name, deleting a feature or symbol, etc.) some of which require pointing at already drawn features. Using the CARIS GIS interface, the operator can make more precise pointings with the graphic cross-hairs than in the stereomodel. Furthermore, if the user wants to add a name, this can be more efficiently performed if the user can switch to the graphics terminal and type in the name directly than having the user type the name through the host program. This will not only make the host program smaller; but by not intervening in such editing operations as adding names and deleting symbols etc. other than to initiate the process by sending the commands to CARED, the host program also becomes more flexible and intimately integrated to CARIS. This feature may be incorporated using program segments in both the host program and the CARIS GIS which will log requests for a particular key or command to each other and to the user. This facility if provided will make the server interface more powerful and thereby further enhance the development of smaller driver programs. Nonetheless, it has been of tremendous utility in the successful implementation of the system devloped in this study.

5.3 The Photogrammetric Digitizing Module

In general, the digitizing unit is an assemblage of software tools which, enable the collection of digital data from a stereomodel. In addition to its other functions, such as providing an interface for the user, reducing user selected points to the object space system, and transfering measurements to the editing and data storage modules, this unit also has to initiate the analytical plotter system each time the operator chooses to reset or change a stereomodel, thus allowing the operator to perform measurements or pointings. This implies that, functionally, the digitizing unit's operations partly overlap the basic photogrammetric data reduction processes while also serving some cartographic function.

In the process of implementing a system for integrated operation, involving the purely photogrammetric unit, the data collection unit and the purely cartographic unit it is often difficult to achieve a clear separation of operations between the digitizing unit and the other two units. This is to be accomplished such that the entire system, though modularized, achieves coherence and continuity of operation on a time sharing minicomputer system with limited capacity without sacrificing flexibility. The CARIS GIS and the modules (discussed in chapter 4), which control the basic operations and applications of the analytical plotter system demand high computing power. Furthermore, both systems create a number of subprocesses (see next section) which either run serially or in parallel with the parent process, and which in turn may initiate subprocesses of their own, thus imposing an additional burden on the computing device. As explained in the previous section, the server interface of the CARIS GIS is linked to the host program (the digitizing program, which also initiates its own subprocesses), such that the CARIS GIS initiates the execution of the host program to run concurrently with CARIS in oder to share data in real time. Therefore, given the possible limitations on the number of such concurrent processes within a program unit, the host program has to be designed so that it does not initiate many subprocesses to enable it to run simultaneously with the CARIS GIS. This means that it cannot include all the basic photogrammetric operations of the plotter system. Yet, it is important to provide for a flexible operation in the digitizing process. A good digitizing module must at least present the user with a number of options and allow the operator to change or reset a stereomodel. It must be able to control such routine operations as the measurements for interior orientation, the retrieval of relevant data for the selected model from the database, and the recreation of the model for digitizing.

Experience has shown that even with only these seemingly elementary functions, the digitizing program will initiate too many subprocesses and cannot operate simultaneously with the CARIS GIS on the MicroVax-II minicomputer. The digitizing module therefore, has to be freed from some of the analytical plotter's driving mechanisms without compromising the flexibility of the system. The technique which has been devised to overcome these problems is to partition the digitizing module into two (see Figure 5-8). One part on the photogrammetric end of the system is the active or the child part (also called the Child Digitizing driver Module, CDM) which contains a large number of the basic photogrammetric functions. The second part, placed on the cartographic end of the system, is the dummy part (or Parent Digitizing Module, PDM). The parent part is called dummy because it does not contain the actual codes for the digitizing tasks but small code segments which merely communicate with the server module and are light enough to be initiated by the CARIS GIS for concurrent operation. A buffer module links the two parts so that the effect of the analytical plotter operations and its applications is hidden from the cartographic operations while they share data in real time. Figure 5-8 shows the partitioned configuration.

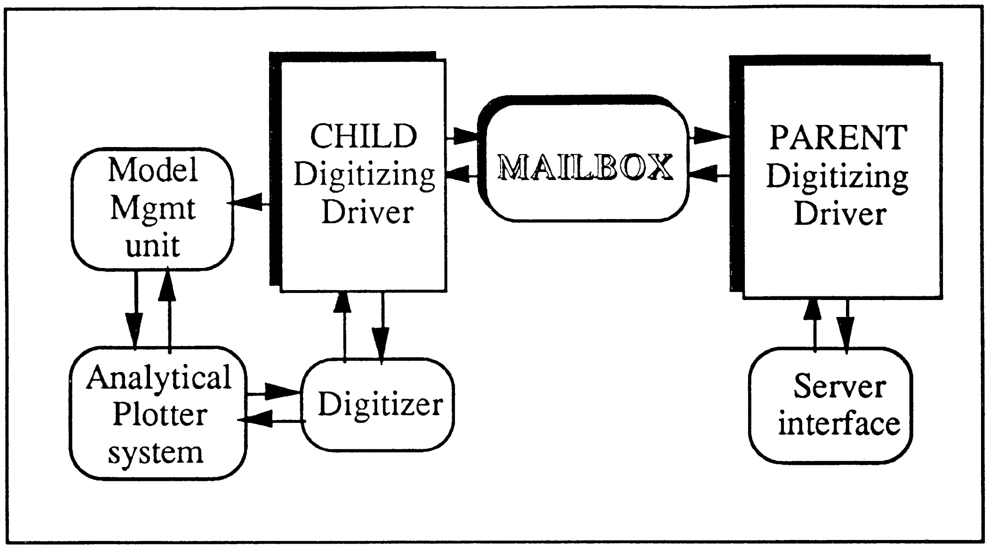


Figure 5-8: Partitioned configuration of the digitizing unit

One advantage of this approach is that it separates the computation intensive photogrammetric operations in such a way that they may be started separately and independently of the cartographic unit. Provided that enough computer power is available, this may also be initiated by the user from the cartographic interface. The approach also allows the child digitizing module to perform a variety of functions as necessary for operational flexibility while its parent (the dummy module) communicates and operates in parallel with the CARIS GIS. As mentioned earlier, the creation of the parent and the child driver modules is facilitated by the VAX operating system. Furthermore, the communication between such independent program units is achieved through the use of the Mailbox.

5.4 Creation of Parent and Child Processes in the VMS Environment

A process denotes an executable program (also called an executable image, applications system or subsystem) which consists of program units that are compiled and linked together. Such program units are often dedicated to perform specified functions where usually one program unit serves as the main program while the others are subprograms. Within the VAX VMS operating system environment, a process may be invoked from the operating system level or from another program. When the process is invoked from a program, it is referred to as a subprocess (a child) to the program which invokes it (the parent). Therefore, a process may invoke another process, which in turn may invoke another process and so on. The number of levels of such processes that may be cascaded is determined by the type of authorization and privileges given to the user.

Conceptually, the technique of process and subprocess usage is analogous to the well known concept of main program and subroutine structures in a higher level computer language such as Fortran. However, while the Fortran main program and its subroutines communicate using argument lists or common blocks, processes and subprocesses communicate using logical name tables, mailboxes, and installed common blocks. Furthermore, a subprocess may be invoked for either synchronous or asynchronous operation with the parent process, and may inherit the parent's properties (such as symbols, logical names, directories, etc.) or it may have its own distinct properties.

Technically, a subprocess is created by spawning calls from another program to the operating system service routine LIB$SPAWN (VAX unpublished documentation 1984). When this service routine is invoked, four arguments are specified. These include (1) the command line statement to be executed (such as "RUN DIGITIZING.EXE"), (2) the subprocess equivalence names for input/output devices (i.e. if the process is an interactive program, what input/output terminals are to be used; usually, the default is the parent's equivalence names), (3) the context and execution specification which indicates whether the subprocess inherits symbols, logical names, and keyboard definitions from the parent, and (4) whether the subprocess executes at the same time as the parent process or while the parent hibernates. Usually, only the first and the fourth arguments need to be specified explicitly as the defaults assumed by the system for the other arguments are applicable in most cases.

A subprocess can execute either while the parent process hibernates (in line) or while the parent process continues to execute (concurrent). If the fourth argument of the system service routine is not specified in the call for the creation of the subprocess, the subprocess will execute in line. However, to have the subprocess execute concurrently, the mask argument (the fourth argument to the system service routine) must be declared as "CLI$M_NOWAIT" when the system routine LIB$SPAWN is invoked. When the command is executed by the system, the subprocess is created and can then perform its dedicated function.

As already alluded to in the previous sections, the major advantages which make subprocess creation so useful in a systems concept include:

1. the ability to perform parallel processing i.e. create a process that executes one part of an application while the parent process continues executing a different part. This enables designation of tasks to separate program modules or the modularization of programs to be developed by separate program groups.
2. the ability to implement multiuser applications i.e. create a process for each application user. The parent process can coordinate the inputs from the created processes.
3. the ability to isolate codes i.e. create a process that executes privileged or sensitive code.

These advantages have formed the basis of the present system. Basically, the CARIS GIS creates the parent digitizing driver module as a subprocess, linked for real time communication by the server interface; the parent digitizing process in turn, creates the child digitizing module as a subprocess, and they both share information through the mailbox; yet, the child digitizing module creates a number of other processes which either run serially or parallel with it. At each stage of the chain, when the request to create a subprocess for synchronous operation is made to the operating system by the parent process invocation of LIB$SPAWN, the child process is started off with the mask set to

CLI$M_NOWAIT and then control is returned to the parent process leaving the child process to continue to execute and perform the data collection task.

To facilitate the real time communication involved in this arrangement, the VAX operating system provides a number of communication options. In general, symbols, logical names, mailboxes, installed common blocks, and global sections allow the passage of information between images executing in different processes. In particular, logical names are used to pass brief messages from one image to another. Mailboxes, installed common blocks, and global sections are used to carry on a dialog between images. The longer the messages in the dialog, the more reasonable it is to use installed common blocks or global sections. However, the communication requirements between the parent and the child digitizing processes are such that a record of information at a time is all that is passed, so the mailbox option is appropriate in the circumstances of the system discussed here.

5.5 The Mailbox Concept

Common knowledge suggests that the word "mailbox" denotes a means of communication between two entities whatever they might be. In the context of computer data processing however, it is a buffer unit which contains a single record into which information is written by one program for another program to pick up. In practice, a mailbox is created through a call to the VAX system service routine "SYS$CREMBX" from within a program. SYS$CREMBX creates the requested mailbox and returns the number of the I/O channel assigned to the mailbox and also its logical name. All the cooperating programs may then open the mailbox for read and write operations.

In general, there are three ways by which the mailbox may be read from or written into. **Synchronous** mode enables one program to perform a read or write operation to a mailbox and then wait for the cooperating image to perform the opposite operation. **Immediate** mode enables a program to perform a read or write operation to the mailbox and then continue to execute after the read/write operation is completed. **Asynchronous** mode enables a program to queue a read or write request to a mailbox and continue program execution while the request is being processed; and when the request is completed, a signal is given and the requested information is available to the process. Synchronous mode is nevertheless the easiest and often the recommended method of addressing a mailbox when real time communication between two processes is intended (VAX documentation 1984). In a digital data collection and editing situation which involves different processes, real time data and command transfer is a requirement; and for efficient operation, such communication has to be synchronized; therefore synchronous I/O is the appropriate choice when one program reads or writes information to another image and cannot continue until that image responds.

In practice, the mailbox is operated like any other file in the program. Once it is opened, then Fortran formatted, sequential read and write statements may be used for the I/O operations. The VAX system automatically synchronizes the I/O by not allowing an image to complete an I/O operation until a cooperating image has performed the opposite operation. For example, if an image performs a mailbox read operation, control is not returned to that image until a cooperating image performs a write operation to the same mailbox. Figure 5-9 shows the communication network of the integrated system.

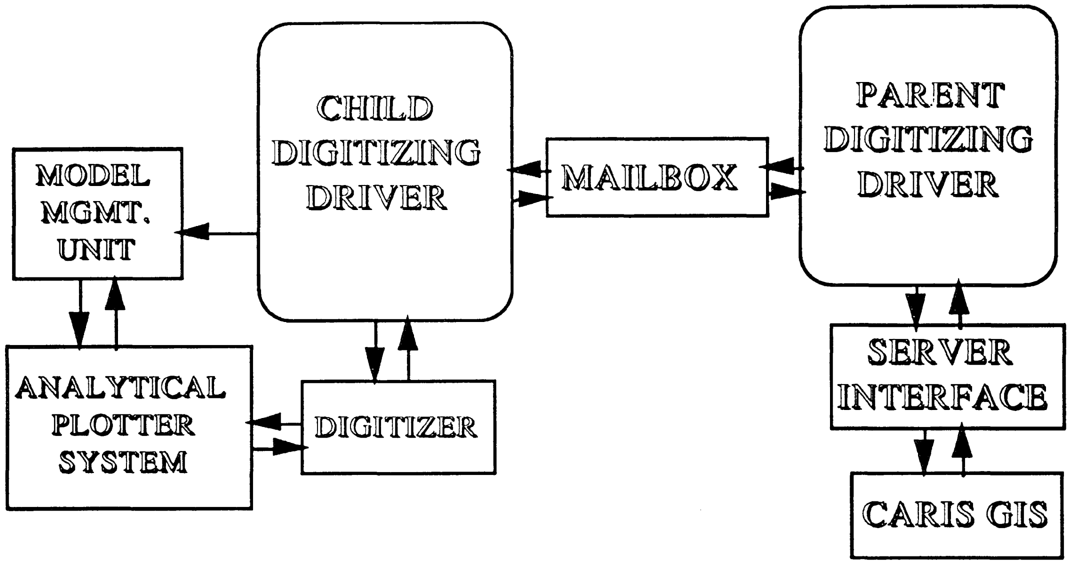


Figure 5-9: The communication network of the Integrated system

To illustrate the use of this concept in the context of the integrated system developed in this study, two example uses are given which demonstrate the synchronized operation between the two parts of the digitizing module discussed earlier. The first example shows the communication mechanism between these two units when a user requests to digitize a line in a point-to-point mode by issuing the CARIS GIS command called **LIAP**. At the parent digitizing process end, the user selects the menu option corresponding to this command; the parent process then writes this command to the mailbox and waits for a response from the child process before continuing with the execution. The child process reads this command and then calls the appropriate subroutine that initializes the digitizing process. The digitizing subroutine displays a message to the user requesting the user to specify the feature code, and other parameters of the graphic element to be digitized in the order in which such information is normally requested in the CARIS GIS environment. This information is then written to the mailbox, and control returns to the parent process which then reads the mailbox and transfers the information to the CARIS GIS through the server interface described in section 5-2. Control is then returned to the child process in which the LIAP subroutine initiates the collection of the stream of X,Y, (Z) coordiantes from the stereo digitizer. These data are then written to the mailbox and read by the parent program, one record at a time until the operator terminates the process. Normally, when the user wants to change a process or terminate the execution of a command or delete a digitized point, this is indicated by pressing the footpedal of the analytical plotter, after which a menu is displayed giving the user a number of possible options for the particular command currently in progress. If the user intends to terminate the command for example, the **Quit** option is selected and the child process sends the appropriate message to the mailbox and control returns to the parent process which then reads this information from the mailbox. Other commands may be similarly selected and processed. One unique beauty of this process is that as the messages are transferred from the child to the parent process, the message is immediately analysed by the CARIS GIS, and the appropriate action is taken such as drawing the element on the graphics screen in real time. In this case, the operator has a chance to check the feature for correctness, and to modify it as necessary to ensure spatial fidelity.

The second example demonstrates the situation between the child and the parent digitizing processes when the operator chooses to change to another stereomodel. From the parent process interface the operator selects the menu option corresponding to the "change model" command. This command is then written to the mailbox for the child process. Reading the change model command, the child process initiates the procedure for model restoration by calling the model management and restoration program into action. As has been mentioned earlier, the model restoration process involves a complex network of interrelated tasks which may include re-initiating the analytical plotter, allowing the operator to perform a whole range of operations such plate registration, image space mensuration, relative and absolute orientations, each of which is controlled by separate dedicated program modules. Therefore, the model management module has to invoke other processes to carry out these functions at the request of the operator. In the meantime, since nothing can be done until a new model is restored, the child process has to wait for these other subprocesses to conclude the model restoration task and return a message to it before it can continue. Therefore, since the child process and these other subprocesses operate sequentially, real time synchronous communication is not required between them. In any case, after the model is successfully changed, the child process writes a job-conpleted message to the mailbox and control returns to the parent process for subsequent operations.

These two examples demonstrate the communcation mechanisms that exists between the photogrammetric digitizing system and the CARIS GIS using process partitioning and the concept of mailboxes. Central to the successful use of the system are the various user interfaces and menus, through which the operator communicates with the system to control the operation. The logical consistency and user friendliness of these interfaces are vital to the productivity of the operator.

5.6 The digitizing user-interfaces

User interfaces are an essential component of any integrated user-oriented system. Being the medium of communication between the user and the computer, its impact on the efficiency of operation of the entire system is tremendous. The human factor is still an important requirement in today's spatial data acquisition, management and application systems; the primary role being to supply the high level intelligence needed in certain critical operations and to direct and control the system towards achieving the intended goal. Obviously, the productivity of the operator and thereby the extent to which system objectives are realized are dependent on the ease and flexibility provided by the user interface. It is therefore logical that a discussion of an integrated system should include aspects of its user interfaces. It is noted however, that the interfaces that are invoked when operations are commenced from the analytical plotter terminal are not included in the present description. Since these are more related to the basic photogrammetric data reduction tasks, they are appropriately dealt with in the description of the photogrammetric software packages and their applications, which is the subject of the next chapter.

The process of acquiring digital data from an analytical plotter involves a number of complex and interrelated operations. Using an integrated system, the operator has to initiate the execution of many programs either for in-line operation or for concurrent operation, and each program presents him with its own interface of optional menu items from which a choice must be made in order to achieve the desired result. The way in which these various menus are organized into logically related classes to facilitate easy use by the operator is extremely important. In this study, hierarchical decompositions into comprehensible classes, based on user work-flow analysis of the appropriate CARIS GIS commands, have been adopted. The menu layers have been arranged in a tree-structured pattern, employing familiar terminologies which progress from the more general class names at the root to the more specific, task-based menus at the leaves.

Yet, a critical variable that determines the attractiveness of a menu selection system is the speed at which the user can move through the menus. The philosophy of minimal keystrokes for menu selection has formed the foundation for the design and implementation of these user interfaces. The aim is to reduce the number of key entries required to select a menu item. To this end, each menu item in a class is assigned a one digit integer number from 0 to 9, with 0 reserved for the Quit option at all levels. For example, if the operator wants to select the option for point-to-point line digitizing (i.e. LIAPZ command), he or she types in the single digit number corresponding to that option. This implies that instead of typing many characters of the CARIS GIS command, only one digit needs to be typed.

Apart from the usual CARIS GIS application interface, there are basically three levels of human-computer interactions involved in the digitizing modules under consideration (see Figure 5-10). The interface of the CARIS GIS is primarily language or command driven; communication is achieved through a set of commands keyed in by the user on the computer keyboard. In response, the computer displays messages on the graphics screen. Because each command entered is processed immediately and the response from the CARIS program is displayed on the screen with a provision for the user to enter another command thereafter, this interface is interactive. On the other hand, the design of the menu system of the digitizing modules is such that the user does not have to type or memorize these commands in the process of acquiring data using the integrated system. All the commands are sent to the CARIS GIS through the server interface by the program units with which the user interacts.

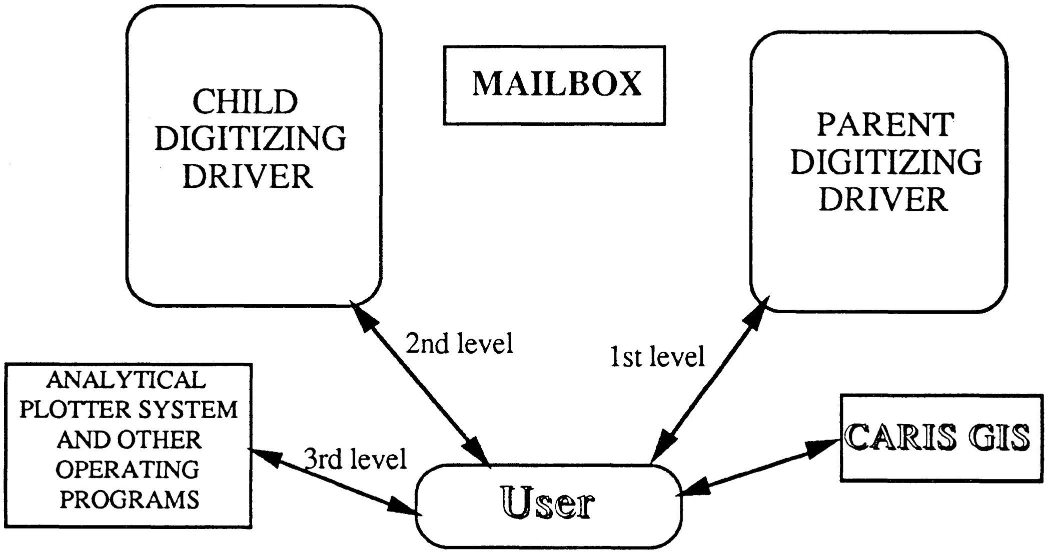


Figure 5-10: Computer-user interfaces for the Digitizing system

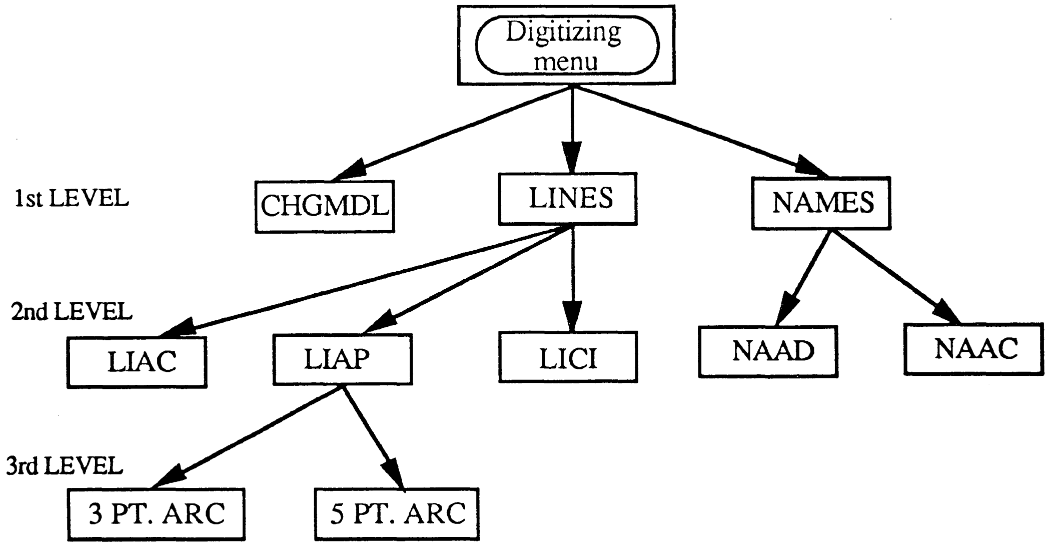


Figure 5-11: Tree-structured Digitizing menu system

5.6.1 First level User interface

The first level (or root level) user-interface is offered by the parent digitizing driver module (see Figs. 5-10, 5-11). In this interface, communication is achieved by the selection of menu items from the display on the graphics screen or some alphanumeric terminal. Included in the menu for this interface are options such as LINES for the digitization of line features, NAMES for the addition or deletion of feature names, SYMBOLS for the manipulation of symbols, and options relating to the use of the analytical plotter system for the restoration and interchange of stereomodels. When the user selects an option, the selection is immediately sent to the child digitizing unit from where other lower-level menus are displayed. In some applications, the selected option is completely processed by the parent driver module. For example, when the user selects the option for changing the scale of the graphics display, the CARIS GIS command "MAd" is immediately sent to the CARIS editor via the server interface and then both the old scale and the prompt for the new scale are displayed. Upon the user entering the new scale specification, it is immediately sent to the CARIS editor to effect the desired change. In general, most of the options at this level are meant to initiate the child process to perform a specified task.

5.6.2 Second level User interface

The second user-interface is offered by the child digitizing driver module. In this case also, communication is largely menu driven; the user selects from a list, usually by keying in the tally number. However, the menu items at this level are more specific than at the parent's level. Specific CARIS GIS commands are itemized and numbered for easy selection. The menus are arranged in groups, and the particular group of items displayed at any time depends on the option selected by the user at the parent's menu level (Figure 5-11). For example, if the user selected option LINES from the parent's menu, then the menu group which contains all relevant CARIS GIS line digitizing commands would be dispalyed at the child driver menu. Practically, the purpose of this menu level is to coordinate the activities of the various subroutines and subprocesses which actually execute the designated tasks. This interface, in essence, brings the user close to the functional level where tasks and options are more specifically defined. Indeed, the selection of a menu option at this level has the effect of activating the particular subroutine or subprocess that is designed to handle the option selected.

5.6.3 Third level User interface

The third user-interface is the collection of menus and sub-menus associated with the subroutines and subprocesses whose operations are controlled by the child digitizing process. These include those offered by the actual digitizing subprograms and those of the photogrammetric sub-processes.Usually, these menus, being at the operational level, are displayed by the software tools designated to perform the particular task the option for which the user has selected at the child digitizing module (CDM) menu level. These options are mainly used to specify how the task is to be performed or to override certain actions of the operating program, or to change the type of data to be collected etc. For example, when the point-to-point digitizing MACRO is activated, the options at the operational level will include (1) deletion of last digitized point, (2) digitizing a 3-point arc, (3) quit this command etc. Other tasks have their own menu options as well. In general, when an option is selected, it is immediately sent to the CARIS editor through the mailbox and the server interface. The CARIS GIS task module then performs the specified operation immediately. On the other hand, the purely photogrammetric menus often lead to a number of other lower-level interfaces, offering menu options related to the photogrammetric tasks (discussed in chapter 6). In all situations however, the user is returned to the CDM driver interface after each task is concluded.

5.6.4 Advantages and Disadvantages of menu selection systems

It is perhaps necessary to note that menu selection systems have their good and bad sides. The good side is that they can eliminate or at least reduce training and memorization of complex command sequences. Moreover, when the menu items are written using familiar terminology, users can select an item easily and indicate their choice with one or two keypresses, or use a pointing device when provided. This simplified interaction style reduces the possibility of keying errors and structures the task to guide the novice user. The bad side however, is that menu driven programs are more difficult to implement and more expensive than language or command driven programs. On the one hand, the planning of the menu structure and its coding is very demanding on the programmer, on the other hand, the code required to support a good user interface is sometimes up to 60 per cent of the entire code (Shneiderman 1987). This is particularly true when such interfaces are required to be flexible, easy to use and at the same time robust. Modular design often simplifies the implementation and synchronization of such user interfaces.

In summary, data format incompatibility poses a great obstacle to successful implementation of an integrated system for real time operation. Not only does it make such a system inefficient and expensive to run, but also makes its realization almost impossible.

These problems have been overcome in this study by making use of the editing facilities of the CARIS software. Moreover, using the mailbox concept it has been possible to partition the supporting digitizing module into the parent and the child, and to establish real time communication between them. Furthermore, it is seen that the user interacts with a number of program units in the utilization of the system, and his productivity has been shown to depend on the ease of such human-program interaction. Well structured and friendly user interfaces have been shown to provide the convenience and flexibility which improve the usefulness of the system. Hierarchical decompositions of functions and tree-structured menu arrangement have been utilized in this system. In the next chapter, the description of the main photogrammetric software packages, including the basic functional programs (the MACROS) and the task manager modules (the CONTROLERS) is given. The application menu structure is also explained for a typical application task.

Chapter 6

Basic Operators, Controllers and Photogrammetric Application Menu Structure

This chapter presents a description of each of the main software packages developed in this study. To simplify the presentation, the programs are grouped into:

(1) basic operator modules (MACROS), which execute the various tasks,

(2) the task manager modules (CONTROLERS), whose duty is to schedule the tasks involved in an application and to invoke the basic operators.

For reason of brevity, emphasis is placed on the purpose and operational charateristics of the basic operators; and for the same reason, only the functional structure of each of the controler modules is outlined. The chapter concludes with a discription of the menu layout followed in an application. Figure 6-1 shows the entire software network.

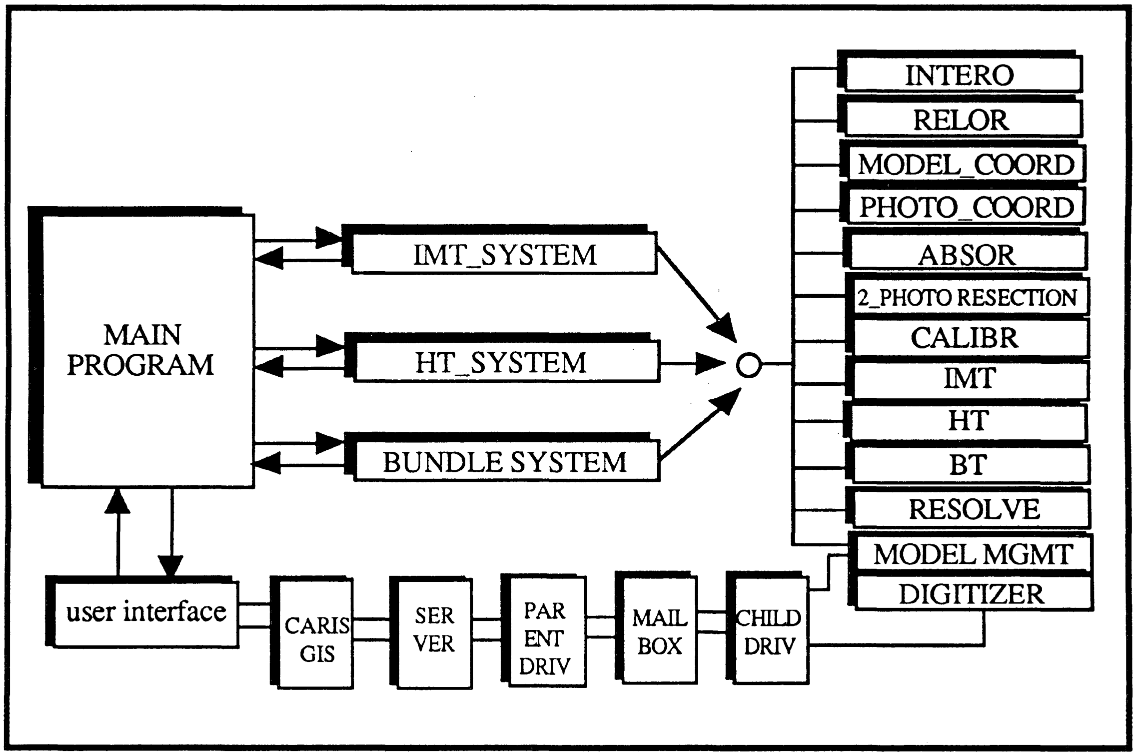


Figure 6-1: Layout of the main software modules.

6.1 The basic operational software tools (MACROS)

The basic operator modules are the software tools which perform the fundamental tasks necessary for the reduction of the photographic images to the object space. They include modules for interior orientation (QDSR20, INTERO), relative orientation (QDSR30, RELOR), absolute orientation (QDSR40, ABSOR), photo coordinate mensuration (PHOTO_COORD), model coordinate mensuration (MODEL_COORD), block triangulation (IMT, HT, BT), camera calibration (CALIBR), model definition (RESOLVE), model management and restoration program (SETBLOCK), parent digitizing driver program (PARENT_D), child digitizing driver program (CHILD_D), and the digitizing head (DIGITIZER). These modules are usually initiated into action by the task managers to perform designated functions. The following provides a discription of each module.

6.1.1 Interior orientation program (QDSR20, INTERO)

This module is designed to enable the registration of photographic diapositives on the stages of the analytical plotter, and also allows the updating of the camera information file. The program QDSR20 was supplied by Kern for this purpose, and was originally designed to operate only with other Kern software packages. Nevertheless, because of its good user interface and the possibility of using different types of mathematical models for plate registration, it has been augmented with a support module which reorganizes and translates its data file to conform with the requirements of the new workstation. On the other hand, INTERO is an experimental program developed to investigate an algorithm which reduces interior orientation time for photographs equipped with crossing but non-intersecting fiducial marks [Olaleye and Faig 1992]. This program currently uses a linear affine mathematical model. In practice, the interior orientation program is initiated as a subprocess by either the mensuration module while performing the initial mensuration task, or by the model management program during digitizing when the operator calls for a new model to be restored.

6.1.2 Relative orientation package (QDSR30, RELOR)

The relative orientation module facilitates the measurements and computations necessary for the determination of the relative orientation parameters of overlapping photographs. Following a predefined pattern, the module moves the measuring marks to a number of locations on the photographic images and allows the operator to pinpoint and measure identifiable image points on both photographs. These measurements are then used to compute the relative orientation information needed for the creation of the stereomodel. The QDSR30 module, provided by Kern, has been augmented with translators to enable it to work within the new environment. The program RELOR was developed in this study as a non-interactive program, and uses a rigorous solution to ensure acceptable results in most cases (including incomplete models), making use of the vector space formulations discussed in section 3.5.3 of chapter 3. The relative orientation package is invoked from the mensuration program or from the model resolution unit.

6.1.3 Absolute orientation package (QDSR40, ABSOR)

The absolute orientation module serves to relate a stereomodel to the object space using a number of common points known in both coordinate systems. The module QDSR40, supplied by Kern, may be used either as standalone or else chained to the other Kern software packages. ABSOR is a program developed to perform the absolute orientation computations based on the vector algorithms discussed in section 3.5.2, chapter 3. Operationally, the absolute orientation program is initiated by either the model resolution or the model management unit.

6.1.4 Photographic Coordinate Mensuration (PHOTO_COORD)

This is one of the more advanced operators which are capable of initiating a number of lower level subprocesses. The module enables the measurement of photographic coordinates of any number of photographs for use in the triangulation and model resolution processes. Since it primarily provides data for tasks requiring the photographic coordinates, it is invoked by the task managers to carry out the required mensuration. Upon being invoked, it initiates other modules such as for interior orientation and project definition. It also includes the subroutines that actually collect the photocoordinates interactively, and those for optional automatic point numbering. The program uses a software switching system to enable the user to track conjugate points starting either with the left image or the right image, thus making the process more flexible. In the Kern software for relative orientation where photo coordinates are measured, provision is made only for using the left image for this purpose. The program also employs a dynamic weighting technique [see Brown 1983 p181, Gelb 1986 p5], to facilitate repeated measurements, thereby providing the most probable value for each measurement. Furthermore, it provides an optional semi-automated post-measurement checking facility which drives to each of the measured points automatically, pauses for the operator to check for positional correctness of the measuring marks, and enables re-positioning and remeasurement to replace or reinforce previous measurements. This package also allows the user to edit the measurement file and to input the control information into the data file. It has the facility for suspended operation and also offers a full interactive environment. The package may be invoked by any task manager requiring image space coordinates.

6.1.5 Model coordinate mensuration (MODEL_COORD)

This package is similar to that for photo coordinate mensuration except that it measures model space coordinates. It is initiated from any task management program which requires the acquisition of model coordinates. The package itself initiates other subprocess such as project definition, interior orientation, and relative orientation tasks. It includes program segments which interactively perform the measurement. An optional automatic numbering facility is also provided, which assigns consecutive integer numbers to all measured points by default. A post measurement re-visit facility is provided which enables measured points to be re-checked and possibly remeasured. Interactive interfaces are provided for editing and printing of the measurement file and the object space control information file. In order to ensure uniformity of the data format, this module stores all the camera related parameters such as lens distortion values, principal point coordinates, camera constant, projection centre coordinates, relative orientation parameters etc. into a well structured data file for use by other programs. It also has facilities for suspended operation in which the user may suspend the measurement process to continue at a later date. When the program terminates, it returns to the environment of the task server from which it was initiated.

6.1.6 Independent Model Block Triangulation Package (IMT)

This is a computational MACRO which transforms a block of stereomodels into the object space. Given sufficient object space control and model space coordinate measurements, the program makes use of the vector space transformation algorithms discussed in section 3.6.2.2, chapter 3 to compute the absolute orientation parameters of an entire block of models. As a by-product, object space coordinates of passpoints, tiepoints and other unknown points are computed.

When initiated, this program checks that all the necessary files are available and that the information they contain is consistent with the requirements for a solution. If any file is missing or corrupted, appropriate error diagnostic messages are issued. The program checks that there is redundancy for the adjustment process. If the degree of freedom is zero or negative, the program gives related error messages and returns to the task manager. Moreover, corrections are applied for earth curvature and refraction. It also checks for possible divergence in the adjustment process, updates the user on the number of iterations and provides statistical information regarding the adjustment. If desired, the residuals for all points may be printed or the largest residual may be viewed. The program keeps all orientation data in a designated file and in a format suitable for the resolution/restoration program. The object space coordinates are also stored, and any section of the data may be routed to the online printer. At the termination of the task, the program returns to the initiating task manager.

6.1.7 Analytical independent model triangulation package (HT)

This package makes use of image space coordinates to form stereomodels analytically and then reduces such models to the object space. It employs the algorithms presented in sections 3.5.3 and 3.6.2.2 of chapter 3 for the transformation. All systematic errors are corrected for at the image space; the model coordinates are then computed and used in the computation of the model to object transformation parameters. All the facilities described for the IMT program are also available in this module.

6.1.8 Bundle Triangulation Package (BT)

This is a computational program which uses image space measurements to compute the exterior orientation parameters of an entire block of photographs using appropriate object space information. The vector algorithms discussed in section 3.2 of chapter 3 are used in a rigorous least squares adjustment. The facilities provided by this MACRO are similar to those described for the IMT module. At the conclusion of the process, the program exits to the main menu where the user can then interactively edit the block triangulation results and modify the data or even elect to remeasure some or all of the points for a possible re-run of the triangulation application.

6.1.9 Camera Calibration Package (CALIBR)

This program is designed to supply the camera calibration information in situations in which such information is not known at all or not accurately known by the user. The module makes use of a subset of the block of photographs and some object space information to compute values for the calibration data. It also provides data regarding average flying and terrain heights. The computed information is written into the camera calibration file in a format suitable for use in the programs which need such information. The block-invariant case of camera calibration (Moniwa 1977) is used by this program.

6.1.10 Model Definition Package (RESOLVE)

This package has the duty of resolving the output of the block triangulation process into its constituent stereomodels. The information needed to restore each stereomodel, i.e. image to model and model to object space transformation matrices, are extracted from the block output data, sometimes aided with data from the mensuration unit or supported with fresh measurements by the operator. Indeed, this program calls a number of MACROS to perform the computational aspects of the operation. Specifically, program modules for relative orientation, absolute orientation and 2-photo resection are invoked depending on the system type in use as explained in chapter 4. This program gathers the relevant information regarding each model from all the working files and collates them; it then writes the data for each model on one record with the proper identification of the photographs from which the model is formed. This information is then stored in a random access file which is readily accessible to the model management package. The five types of these module are named **MR1, MR2, MR3, MR4 and MR5.**

6.1.11 Model management and restoration program (SETBLOCK)

This program facilitates the restoration of individual stereomodels back to the instrument, usually for digitizing to commence. The menu option to restore a model is selected at the parent digitizing driver menu level, after which this program is activated by the child driver module. Upon being initiated, this program asks the operator for the identification number of the model to be restored; it then reads the data file and checks whether a valid model number has been indicated. If the model requested exists in the data file, the relevant information about the model, such as left and right photograph numbers, the principal distances, the project name, whether the model is base-in or base-out, etc. are displayed for use by the operator in the interior orientation. Note that interior orientation is necessary only when the particular stereopair has been removed after the initial mensuration. After the interior orientation is established, this program automatically restores the model and downloads the transformation matrices to designated locations for use by the digitizing head module, and also to the real time processor P2.

6.1.12 Parent digitizing driver program (PARENT_D)

This program serves as a clearing house for operator's commands and data between the photogrammetric workstation and the cartographic unit. It is usually initiated by the CARIS GIS. Its main features have been discussed in chapter 5.

6.1.13 Child digitizing driver program (CHILD_D)

This program coordinates the digitizing activities within the photogrammetric workstation. It receives commands from the parent driver module through the mailbox and assigns the task to the relevant program unit. As has been noted in chapter 5, this program is the active digitizing driver module since it initiates the execution of the desired operation and allows for the transfer of data to the cartographic unit.

6.1.14 The Digitizing head (DIGITIZER)

This program facilitates the actual collection of digital data from the stereomodel. Its function is similar to that of a digitizing tablet or a graphics cross-hair. It enables the movement of the floating mark to desired points in the model and at the press of the footpedal, the coordinates are taken and immediately transfered to the child digitizing driver and to the cartographic unit via the mailbox .

6.2 The Task management modules

These are modules which schedule a job into a number of tasks and serve these tasks to the task handlers described above. These modules include IMT_SYSTEM, HT_SYSTEM and the BUNDLE_SYSTEM. The outline of each module is given in the following.

6.2.1 IMT SYSTEM

This is an implementation of the system IMT_000 discussed in section 4.3 of chapter 4. The program implementation coordinates all the activities necessary for the reduction of photographic images to digitizable stereomodels using the independent models technique without re-orientation computations. Thus, it is termed a controller.
The progression of operational steps for this system is as follows:

1. Data preparation

Mensuration

- interior orientation for each stereopair; QDSR20, INTERO macros are invoked.
- relative orientation (initiates QDSR30, RELOR)
- measurements of model coordinates of all pass-, tie- and control points in each model (MODEL_COORD)
- storage of I.O. and R.O. elements and measured model coordinates
- repeat process for all stereopairs in the block

Triangulation

- compute the A.O. for all models (IMT triangulation)

Model Definition (RESOLVE)

- extract information about each model, construct the scaled rotation matrix and translation vector for each model and store in one data record

2 Digitizing

model restoration (SETBLOCK)

- check the validity of each model and retrieve its data from the data file
- interior orientation (QDSR20, INTERO)
- download the I.O. and R.O. information to processor P2 to restore the model
- download the scaled rotation matrix and translation vector for the model to the stereodigitizer module.

Digital data collection (stereo DIGITIZER)

- real time conversion of each operator selected model space vector (P_m) into corresponding object space vector P_A

- transfer of object space coordinates through the MAILBOX and the SERVER to the CARIS GIS (for drawing and storage)

6.2.2 HYBRID SYSTEM

This is an implementation of the system HT_000 discussed in section 4.3 of chapter 4. This program implementation performs a similar coordinative role as described for the IMT system. It employs the analytical independent model technique without re-orientation computations.

The progression of operational steps for the hybrid system is as follows:

1. Data preparation

Mensuration

- interior orientation for each stereopair; QDSR20, INTERO macros are invoked.
- measurements of photo coordinates of all pass-, tie- and control points on each stereopair (PHOTO_COORD)
- storage of I.O. elements and measured coordinates
- repeat process for all stereopairs in the block

Triangulation

- compute the E.O. for all photographs (R.O. + IMT programs)

Model Definition (RESOLVE)

- extract information about each model, construct the scaled rotation matrix and translation vector for each model and store in one data record

2 Digitizing

model restoration (SETBLOCK)

- check the validity of each model and retrieve its data from the data file
- interior orientation (QDSR20, INTERO)
- download the I.O. and R.O. information to processor P2 to create the model
- download the scaled rotation matrix and translation vector for the model to the stereodigitizer module.

Digital data collection (stereo DIGITIZER)

- real time conversion of each operator selected model space vector (P_m) into the corresponding object space vector P_A
- transfer of object space coordinates through the MAILBOX and the SERVER to the CARIS GIS (for drawing and storage)

6.2.4 BUNDLE SYSTEM

This is an implementation of the system BT_570 discussed in section 4.3 of chapter 4. This program implementation performs a similar coordinative role as described for the IMT system. It employs the rigorous bundle block adjustment method of triangulation with automated re-orientation computations.

The progression of operational steps for the bundle system is as follows:

1. Data preparation

Mensuration

- interior orientation for each photograph (QDSR20, INTERO macros are invoked)
- measurements of photo coordinates of all pass-, tie- and control points on each photograph (PHOTO_COORD)
- storage of I.O. elements and measured coordinates

- repeat process for all photos in the block

Triangulation

- compute the E.O. for all photographs and object space coordinates of all points (BUNDLE triangulation macro).

Definition (RESOLVE)

- automatically compute R.O. and A.O. elements for all stereopairs (using the results from mensuration and triangulation)
- construct the scaled rotation matrix and translation vector for each model and store in one data record

2 Digitizing

model restoration (SETBLOCK)

- check the validity of each model and retrieve its data from the data file
- interior orientation (QDSR20, INTERO)
- download the I.O. and R.O. information to processor P2 to create the model
- download the scaled rotation matrix and translation vector for the model to the stereodigitizer module.

Digital data collection (stereo DIGITIZER)

- real time conversion of each operator selected model space vector (P_m) into the corresponding object space vector P_A.
- transfer of object space coordinates through the MAILBOX and the SERVER to the CARIS GIS (for drawing and storage)

6.3 Photogrammetric application menu structure

With the integration of the DSR-11 analytical system to the CARIS GIS environment, the necessary tool for direct acquisition and editing of spatial data has been put in place. In this section, the procedure for using the system is discussed by way of describing the menu structure through which the user is taken when perfoming a specified mapping task. However, because the entire system as integrated comprises many processes, each of which is dedicated to a somewhat complex set of operations, the detailed description of all the steps involved in its utilization will be long and difficult. Therefore, only the key steps are summarized. Figure 6-2 shows the layout of the menu system, and it is followed in this description starting from the top to the bottom.

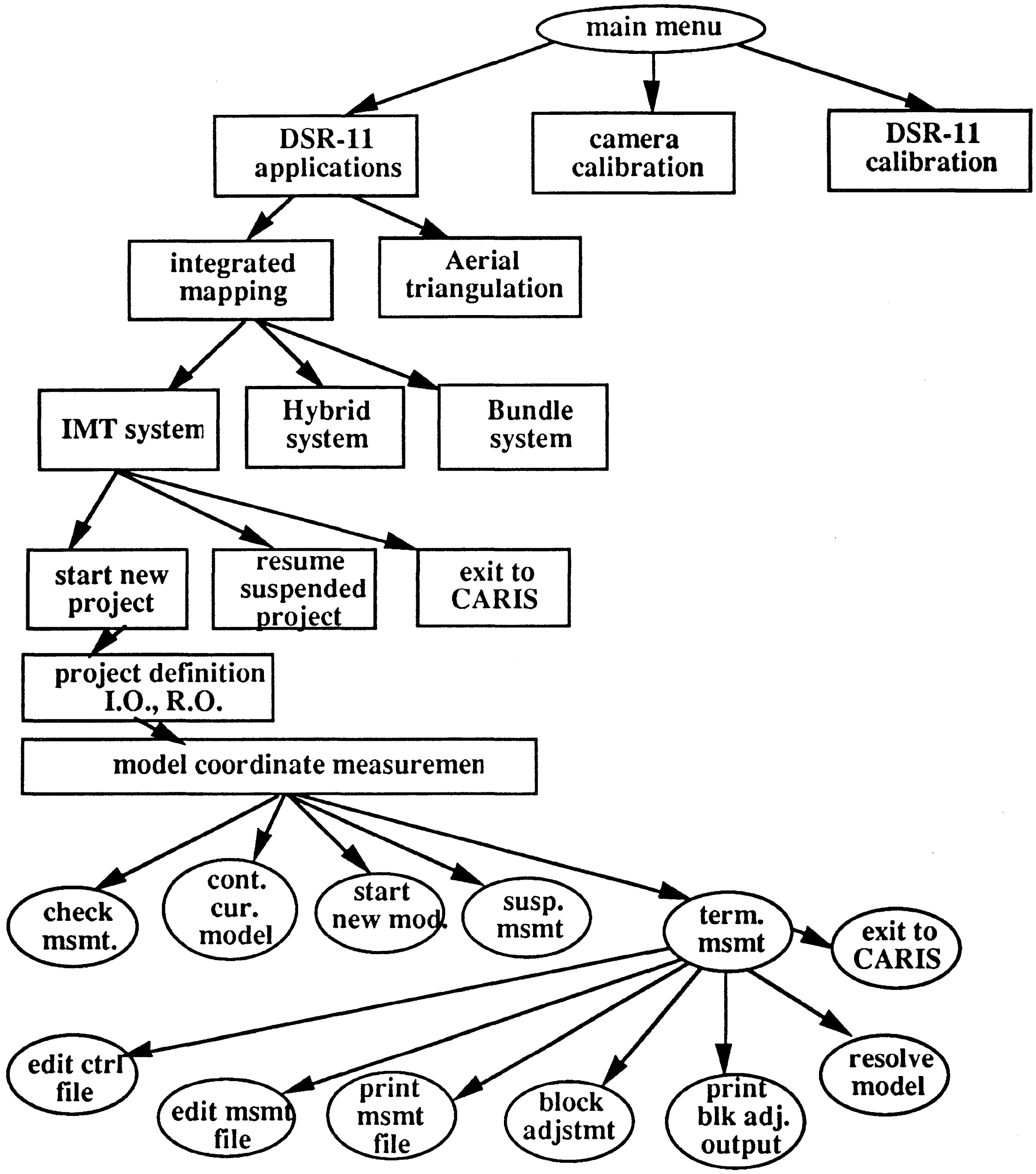


Figure 6-2: Photogrammetric application menu structure

At the main menu level there are three options to choose from, namely:

(1) DSR-11 applications

(2) camera calibration

(3) DSR-11 calibration.

The DSR-11 application option opens the door to the various mapping systems discussed in chapters 4 and 5. The camera calibration option leads to an environment in which the data relating to the operating characteristic of the camera may be derived through measurements and computations. The DSR-11 calibration option enables an empirical determination of the operating characteristics of the analytical plotter.

For a mapping project, the application option will usually be selected at the main menu, and this opens an application menu which consists of two options namely:

(1) integrated mapping
(2) aerotriangulation (Figure 6.1).

Choosing the aerotriangulation option, the user is lead to an application menu in which any of three techniques may be used for the determination of object space points over large areas. This option is useful for applications in which the immediate objective is not the acquisition of detailed maps. Selecting the integrated mapping option however, the user is presented with the three mapping controlers discussed in the previous section. These include:

(1) the IMT SYSTEM
(2) the HYBRID SYSTEM
(3) the BUNDLE SYSTEM.

As noted earlier, these three systems achieve the same goal but use different techniques with the bundle system being the optimal design. However, for future research purposes and to accommodate user preferences for a particular system, all three have been implemented. Regardless of this fact, let us objectively, follow the way of the first option, i.e. IMT SYSTEM. This leads us to the operational environment in which the choice of a menu option results in some action related to the mapping task. There are three choices we can make at this level (see Figure 6-2). These include:

(1) start a new project
(2) resume a suspended project

(3) exit to CARIS GIS.

While the first two options are self-explanatory, the third, indeed, is one of many outlets to the command procedure from which the CARIS GIS program may be manually initiated for the actual digital data collection. This option checks to see that all the necessary data files are available before returning to the command level procedure. However, choosing the "start new project" option, the user is taken to the activity area where the initial operations of project definition, interior and relative orientations, and model coordinate measurements are performed by program modules providing interactive interfaces.

As mentioned earlier, project definition involves the specification of the relevant information about the project, such as project name, camera type and characteristics, average terrain and flying heights, etc. If the project information was specified when using the camera calibration program, this stage may be skipped. The interior and relative orientation tasks are performed in sequence using the MACRO programs described in section 6.1.

The mensuration operation is handled (in this case) by the MODEL_COORD MACRO of section 6.1.5. At the mensuration stage, five menu options are provided (see Figure 6.1). These include options to:

(1) check measurements
(2) continue current model
(3) start a new strip
(4) suspend measurement
(5) exit measurement.

The check measurements option allows the verification of all measured points and a replication of measurements for higher mensural quality. The continue current model option enables the resumption of measurements of the current model after a temporary termination, while options 3 and 4 provide avenues for changing to another model and suspending the measurement process respectively. Option 5 is the exit route which has to be selected in order to get out of the measurement environment.

When this option is chosen, the user is taken to the editing and adjustment menu where seven tasks may be performed. These include:

(1) editing of control information file
(2) editing of measurement file
(3) printing of measurement file
(4) adjustment of the block of models
(5) printing of the block adjustment output file
(6) model definition
(7) exit to CARIS GIS.

The control file editing option allows the inputing and manipulation of the object space information which is needed for the adjustment process. Any number of control point coordinates may be included together with their weights. The block adjustment package correlates the control points with their measurements using the identification numbers. Any control point that cannot be so correlated is simply ignored. The editing of the measurement file grants the user an access to the measured model coordinates. The user may want to look through the file to make changes or deletions. This option provides full editing capability for this purpose. The printing facility allows the user to obtain a printed copy of the measurement file from inside the program.

Of particular importance is the option for the adjustment of the block to the object space. This enables the simultaneous determination of the absolute orientation parameters for all the models in the block using the given control information and the measurements. The adjustment module is basically non-interactive, but provides diagnostic messages relating to the status of the application as said before. At the conclusion of the process, the program exits to the same menu where the user can then interactively edit the block triangulation results and modify the data or elect to remeasure some or all of the points for possible re-run of the triangulation procedure. When the user is satisfied with the results of the triangulation, he may select the option for the model resolution application. The model resolution MACRO collects all relevant information from the project definition file, the camera information file, the measurements file, and the block adjustment output file. It then sorts out the essential information regarding each model, for the subsequent restoration and digitization of such a model. The information is then written to the model information file, one record per model. The model information file is an unformatted random access file in which the model serial number in the block is the record number containing its essential parameters. The model management program can thus easily pick up any model during digitizing.

The last option at this operational menu level is the one that exits to the command level procedure from where the CARIS GIS is initiated for the commencement of digitization. When the CARIS GIS has been initiated, it starts off the digitizing program and the whole operation continues as described in chapter 5. The menu structure discussed in that chapter then becomes operational. However, at this level, as indeed at all the menu levels in Figure 6-2, provision is made for the user to quit to the main menu to commence a new project or initiate other actions or to terminate the entire process.

Although, this section has discussed the menu sequence followed when using the IMT SYSTEM, the pattern for the other systems is the same.

Chapter 7

Conclusions and Recommendations

This study has presented the analytical photogrammetric system as a workstation consisting of tools and procedures for transforming photographic measurements into corresponding object space information and vice versa, sometimes in real time. Apart from the analytical plotter hardware and the host computer, most of the tools were recognized as software-based subsystems which perform tasks such as measurement of image space (or model space) coordinates, triangulation, stereomodel definition, stereomodel restoration and digitizing. The mathematical formulations which underlie the development and functioning of these components were presented entirely in vector symbology. Using the **R**-space and the **C**-space concepts, a new methodology called the ARDOVS relations of vector spaces was developed. This enabled the derivation of simplified vector-oriented algorithms for implementing the software tools for the workstation. It was noted that while the basic concepts of such functional models were not new, the vector-based representation adopted in this work stimulated a new understanding of the basic technologies and facilitated the implementation of efficient programs for their applications; and since only little vector arithmetic was required in the process (and may be routinely applied), substantial savings in the thought processes and in the efforts involved in developing the tools for a wide variety of applications were achievable with the vector approach.

Furthermore, the various options available for the basic tasks were explored. Different operational configurations evolved through an interplay of the available methodologies for accomplishing the basic tasks; and these provided the software design alternatives (system architectures) for the workstation. Consequently, eight software designs were implemented on the DSR-11 through the integration of the basic processes into different operational systems. The research then conducted extensive and thorough practical investigations and testing of all the systems developed, using both real and simulated data. The tests were designed to collect data concerning the operational characteristics of each system architecture for use in a mathematical analysis. Automation content, speed of operation, production cost, and achievable accuracy of the derived spatial information were the criteria used as constraints in a linear optimization strategy in an attempt to select the optimum system design. For this purpose, an empirical cost function and a set of constraint functions required for the optimization scheme were developed. This also involved system costing (hardware, software and human labour). A rigorous analysis led to the selection of the most efficient software design for the workstation.

Having established the optimum software design that satisfied the stated requirements and implemented it on the DSR-11, we arrived at an efficient and software-intensive mapping workstation capable of providing accurate 3-D spatial information for use in diverse applications. However, a second major objective of the study was to achieve the integration of the photogrammetric workstation into the CARIS GIS for real time operational communication. As noted in the thesis, the need for such integration is growing rapidly. The necessity to minimize data acquisition cost by eliminating expensive intermediate processes of data conversion, the need to ensure data integrity through instant drawing of the captured data on a graphics terminal for comparison with the measured object, sustained improvement in digital technology leading to an ever decreasing time lapse between data acquisition and data utilization in some new automated real time processes etc. suggest a growing popularity of direct integration of the data source to the GIS environment. In this work, the integration was achieved through the CARIS server interface and the multitasking facility available in the Vax VMS operating system environment. The developmental processes and operational mechanisms of the driver module developed, and the methodology of partitioning it into parent and child driver modules were presented. Moreover, it was explained how, using the Mailbox technology, a real time communication channel was established between the parent and the child modules in synchronous operation. Finally, it was noted that the user interacts with a number of program units in the utilization of a complex system, and that well-structured and user friendly interfaces conceal the complexity, and provide the convenience and flexibility which improve the usefulness of the system. Hierarchical decompositions of functions and tree-structured menu arrangement were utilized in the system developed.

Based on the results of this study and the analyses performed, the following conclusions are made:

1. There are differences in the performance of the software design possibilities for a photogrammetric workstation depending on the requirements for project time (speed of operation), operator involvement, and accuracy of the derived spatial information.

2. The optimum software architecture for an analytical photogrammetric mapping workstation, in terms of criteria of minimum project time, minimum operator participation, and high accuracy of the spatial information produced, all simultaneously applied, is a comparator-based design which includes a bundle triangulation, supplying the object space control points (including minor control points) to a stereomodel definition unit which employs the automated relative and absolute orientation computations to recover the model restoration data. Using the notations employed in this work, this conclusion suggests that significant savings in both system time and manual labour involvement plus high spatial fidelity of the final product can be realized by using the BT_570 architecture for the photogrammetric workstation.

3. When the accuracy requirement may be somewhat relaxed (not a whole lot), then the software design (BT_000) which employs bundle triangulation supplying only the exterior orientation parameters of the camera (image space) to a model definition

unit which recovers the model restoration data from these parameters will be most economical.

These conclusions are valid for the stated criteria and within the limitations of the experiment conducted here. Naturally, changes in user requirements which are translated to changes in the constraints on the problem may lead to different conclusions.

4. Since analytical photogrammetry deals with positioning in a coordinate system, vector-oriented algorithms make the concepts clearer than the conventional single variable approach. This also facilitates the development of standard and maintainable software for a wide variety of applications. The use of the **R**-space and the **C**-space formulation provides a better understanding of the relationships between the various vector spaces used in photogrammetry.

5. Direct integration of the photogrammetric workstation to a GIS eliminates the need for intermediate data transfer which often requires data conversion from one format to another, a common feature of most of the CAD/DBMS systems currently in use. However, problems of data format disparity, solitary operation of digitizing software and limitation of the computing device stand in the way of achieving direct integration. These problems were circumvented in this study using the CARIS SERVER interface and the multitasking provision in the Vax VMS environment which allows many programs to operate in parallel and to share data synchronously.

Recommendations

1. The system developed in this study was specifically designed to handle the conventional analogue photograph. In view of the growing popularity of digital image systems, it is suggested that the system be improved and adapted to operate with digital images preferably in a PC-based system. Although the currently achievable accuracy of the digital system is low, it is proving to be useful for a wide variety of applications such as small scale map updating from satellite imageries or scanned photographs, preparation of thematic maps, generation of digital orthophotos, and also can facilitate research into the means to automate the photogrammetric mapping process.

2. It is suggested that the vector approach be adopted as the standard technique for instruction in photogrammetry. Not only will this reduce the learning time and efforts, but also stimulate an appreciation for the applicability of the technology to such interesting areas as computer vision, robotics, industrial process control, measurements of dynamic processes, etc. While it is easy for a beginner in the Surveying Profession to intuitively relate for example, a measured distance between two points to the coordinates of such points, it often requires a greater imaginative skill to relate measurements on a photograph to the object space coordinates of such points with the long-hand formulations currently in use. However, the vector representation makes the task many times easier.

References

Adby, P.R. and M.A.H. Dempster (1974). Introduction to Optimization Methods, Chapman and Hall Ltd., 11 New Fetter Lane, London, EC4P 4EE.

Aguilar, R.J. (1973) Systems Analysis and Design in Engineering, Architecture, Construction, and Planning, Prentice-Hall, Inc., Toronto.

Allam, M.M. (1982) "Limitations Of Current Automated Photogrammetric Instrumentation and the Potential of Future Fully Automated Systems", International Archives of Photogrammetry, Vol. 24-11, pp. 131-147.

Armstrong, J.M. and P.C. Ryner (1981) Ocean Management: A New Perspective. Ann Arbor Science Publishers, Inc., Ann Arbor, Michigan 48106.

Bailey, R.W. (1982) Human Performance Engineering. Prentice-Hall, Inc., Toronto.

Bevington, P.R. (1969). Data Reduction and Error Analysis for the Physical Sciences, McGraw-Hill Book Company, Inc., New York.

Blachut, T.J. (1963) "Opening Remarks. Second International Photogrammetric Conference on the Analytical Plotter", The Canadian Surveyor, Vol. 17, No. 2, pp. 130.

Blachut, T.J. (1979) "Basic Features, Design Considerations and Performance of the ANAPLOT", The Canadian Surveyor,Vol. 33, No. 2, pp. 107-124.

Blanchard, B.S. and W. J. Fabrycky (1981) Systems Engineering and Analysis. Prentice-Hall, Inc., Englewood Cliffs, N.J. 07632.

Bonjour, J.D., Eitel, G. and U. Kolb (1988). "Experiences with the Wild System9-AP and its Stereo Image Injection System", International Archives of Photogrammetry and Remote Sensing, Vol. 27, part B9, pp. IV183-IV195.

Brown, R.G. (1983). Introduction to Random Signal Analysis and Kalman Filtering, John Wiley & Sons, Inc., New York.

Budlong, J.P. (1978) Sky and Sextant: Practical Celestial Navigation. 2nd Edition. Van Nostrand Reinhold Company, New York.

Burnside, C.D. (1979) Mapping From Aerial Photographs. Granada Publishing Ltd., New York.

Campbell, C.S. and S.L. Egbert (1990) "Animated Cartography/Thirty Years of Scratching the Surface". Cartographica, Vol. 27, No. 2, pp. 24-46.

CanLab INSPIRE Documentation (1987).Digital Mapping Laboratory, Department of Surveying Engineering, U.N.B. (Unpublished).

CARIS Documentation (1990)."Introduction to CARED: User's Guide", Universal Systems Ltd, Document No. CAREDINT-JAN90-VMS (Unpublished).

Case, J.B. (1981) "Automation in Photogrammetry". Photogrammetric Engineering and Remote Sensing, Vol. 47, No. 3, pp. 335-341.

Chapuis, A., Berg J. and V. Den (1988) "The New Kern DSR Series of First Order Analytical Plotters". International Archives, of Photogrammetry and Remote Sensing, Vol. 27, part B8, sectn . II, pp. 38-45.

Chestnut, H. (1965) Systems Engineering Tolls. John Wiley & Sons, Inc., New York.

Chrisman, N. (1988) "The Risks of Software Innovation: A Case Study of the Harvard Lab.". The American Cartographer, Vol. 15, No. 3, pp. 291-300.

Clapp, J.A. (1976). "A Review of Software Cost Estimation Methods", Document Prepared for Deputy for Command and Management Systems, Electronic Systems Division, Air Force Systems Command, United States Air Force, Hanscom Air Base, Bedford, Massachusetts, Document No. ESD-TR-76-271.

Cogan, L. and A. Polasek (1988) "Kern Integrated Systems Hardware and Software Architecture". International Archives of Photogrammetry and Remote Sensing, Vol. 27, part B2., pp. 84-90

Collatz, L. (1966). Functional Analysis and Numerical Mathematics, Academic Press, Inc., 111 Fifth Avenue, New York, New York 10003.

Cooper, M.A.R. (1987). Control Surveys in Civil Engineering, Nichols Publishing Company, New York.

Cooper, M.A.R. and S. Robson (1990) "Methods for high precision Photogrammetric Monitoring of the Deformation of a Steel Bridge". Photogrammetric Record, Vol. XIII, No. 76

Corcodel, G. (1988). "On-Line Technology in Analytical Aerial Triangulation", International Archives of Photogrammetry and Remote Sensing, Vol. 27, part B3., pp. 122-127.

Coulson, A.E. (1970). An Introduction to Vectors, SI edition, Longman Group Ltd., London.

Coutinho, J.S. (1977) Advanced Systems Development Management. John Wiley & Sons, Inc., New York.

Dale, P.F. and J.D. McLaughlin (1989) Land Information Management. 2nd Edition, Oxford University Press, Walton St., Oxford OX2 6DP

Dequal, S. (1984). "RELAX: A Program for the Simultaneous Relative and Absolute Orientation of Models in Analytical Plotters", International Archives of Photogrammetry and Remote Sensing, Vol. XXV, part A8a., pp. 71-82.

Detwiler, R.W. (1980) "AUTOPLOT-The Analytical Stereoplotter from System House". Proceedings, Analytical Plotter Symposium and Workshop, Reston, Va., pp. 137-144.

Deutsch, R. (1965) Estimation Theory. Prentice-Hall, Inc., Englewood Cliffs, N.J.

Dinham, S.M. (1976) Exploring Statistics: An Introduction for Psychology and Education. Brooks/Cole Publishing Company, Monterey, California.

Dorrer, E. (1976) "Software Aspects in Desk-top Computer-assisted Stereoplotting". Photogrammetria, Vol. 33, pp. 1-18.

Dorrer, E. (1981). "Real Time Orientation as Integral Part of On Line Analytical Aerial Triangulation", Photogrammetria, Vol. 36, pp. 111-118.

Dorrer, E. (1984). "Operational Aspects of On-Line Phototriangulation", International Archives of Photogrammetry and Remote Sensing, Vol. XXV, part A8a., pp. 197-237.

Dowman, I.J. (1977) "Developments in On-line Techniques for Photogrammetry and Digital Mapping". Photogrammetric Record, Vol. IX, No. 49., pp. 41-54

Dowman, I.J. (1990) "Digital Plotters: Current Status and Future Developments". Photogrammetric Record, Vol. 13, No. 75, pp. 331-345.

Eliezer, C.J. (1963). Concise Vector Analysis, Pergamon Press Ltd., New York.

Faig, W. (1984). "Aerial Triangulation And Digital Mapping", Monograph 10, School of Surveying, University of New South Wales, Kensington, Australia, 107p.

Fritz, L.W. (1984). "Trends in Analytical and Hybrid Instrument Developments: Report of Working Group II/1", International Archives of Photogrammetry and Remote Sensing, Vol. XXV, part A8a., pp. 85-100.

Fuller, W.A. (1987). Measurement Error Models, John Wiley & Sons, Inc., New York.

Funk, I.K. (1963) New Standard Dictionary of the English Language. Funk & Wagnalss, New York.

Gasson, P.C. (1983) Geometry of Spatial Forms. Ellis Horwood Publishers, Ltd., Ontario.

Gelb, A. (1986). Applied Optimal Estimation, ninth printing, The Analytical Sciences Corporation, M.I.T. Press, Cambridge, Massachusetts.

Ghosh, S.K. (1988) Analytical Photogrammetry. 2nd Edition. Pergamon Press Canada, Toronto, Ontario.

Giles, J. R. (1987). Introduction to the Analysis of Metric Spaces (Australian Mathemetical Society Lecture series), Cambridge University Press, Cambridge.

Goodchild, M.F. (1988) "Stepping Over the Line: Technological Constraints and the New Cartography". The American Cartographer, Vol. 15, No. 3, pp. 311-319.

Gottfried, B.S. and J. Weisman (1973). Introduction to Optimization Theory. Prentice-Hall, Inc., Englewood Cliffs, New Jersey.

Green, R. (1990) "Geographic Information Systems in Europe". The Cartographic Journal, Vol. 27, No. 1, pp. 40-42.

Greve, W.C. (1980) "The Analytical Photogrammetric Processing System-IV". Proceedings, Analytical Plotter Symposium and Workshop, Reston, Va., pp. 79-85.

Gruen, A.W. (1982). "Phototriangulation with Analytical Plotters Using Natural Tie points", Proceedings of the Symposium on Mathematical Models, Accuracy Aspects and Quality Control, ISPRS Comm. III, Helsinki, Finland, International Archives of Photogrammetry, Vol. 24-III, pp. 202-211.

Gruen, A.W. (1984). "Algorithmic Aspects in On-Line Triangulation", International Archives of Photogrammetry and Remote Sensing, Vol. XXV, part A3a, pp. 342-362.

Guralnik, D.B. (1979) Coles Concise English Dictionary. Coles Publishing Co., Ltd., Toronto.

Halmos, P.R. (1958). Finite-Dimensional Vector Spaces, 2nd Edition, D. Van Nostrand Company, Ltd., 25 Hollinger Road, Toronto 16, Canada.

Heikkila, J. and E. Kilpela (1984). "Aerial Triangulation by Analytical Plotter". International Archives for Photogrammetry and Remote Sensing, Vol. 25, part A3a, pp. 393-399.

Helava, U.V. (1982) "The Analytical Plotter in a Photogrammetric Production Line". Technical Papers of the ACSM-ASP Convention, Denver, Colorado, pp. 251-256.

Helava, U.V. (1988) "On System Concepts for Digital Automation". Photogrammetria, Vol. 43, pp. 57-71.

Hobbie, D. and H. Rudenauer (1984) "The Zeiss Planicomp Family: A user-oriented solution to practical requirements". International Archives of Photogrammetry and Remote Sensing, Vol. XXV, part A2, pp. 232-244.

Hodgson, M.E., Barrett, A.L., and R.W. Plews (1989). "Cartographic Data Capture Using CAD", Auto-Carto 9 Proceedings, Ninth International Symposium on Computer-Assisted Cartography, Baltimore, Maryland, pp. 357-366.

Hoffman, K. (1975). Analysis in Euclidean Space, Prentice-Hall, Inc., Englewood Cliffs, New Jersey.

Holm, K.R. (1988). "Further Use of Givens Transformations in On-Line Phototriangulation", International Archives of Photogrammetry and Remote Sensing, Vol. 27, part B3, pp. 332-341.

Hoppe, A. (1991). “Concurrent and Distributed Programming in UNIX Environment”, International Journal of Mini and Microcomputers, Vol. 13, No. 3, pp. 109-116.

Hunter, D.J. and J. Smart (1988) "Image Orientation On the Kern DSR". International Archives of Photogrammetry and Remote Sensing, Vol. 27, part B2,pp.

International Systemap Corporation (1991). Photogrammetric Engineering and Remote Sensing, Vol. LVII, No. 2, pp. 134.

Jacobson, K. (1984). "Data Collection for Bundle Block Adjustment on Analytical Plotters", International Archives of Photogrammetry and Remote Sensing, Vol. XXV, part A3a, pp. 435-439.

Jaksic, Z. (1976) "The Significance of Analytical Instruments for the Development of Methods and Techniques in Photogrammetric Data Processing ". International Archives of Photogrammetry, Vol. XXI, part 3, secti on 2 04 194, pp. 1-23

Jaksic, Z. (1980). "Evaluation Of Functional And Structural Characteristics Of Analytical Plotters". Invited Paper, International Archives Of Photogrammetry, Vol. XXIII, part B2, pp. 106-116.

Jaksic, Z. (1983) "Man-Machine Photogrammetric Systems and Systems Components". The Canadian Surveyor, Vol. 27, No. 4, pp. 308-319.

Jaksic, Z. (1984a) "Photogrammetric Processes and Systems", International Archives of Photogrammetry, Vol. XXV, part A2, pp. 271-281.

Jaksic, Z. (1984b) "Instrumention for Data Reduction and Analysis: Report on the Activities of ISPRS Commission II, 1980-1984", International Archives of Photogrammetry, Vol. XXV, part A2, pp. 262-270.

Jensen, C.R. and K.E. Kleinn (1980) "Analytical Stereoplotter Model Setup ". Proceedings, Analytical Plotter Symposium and Workshop, Reston, Va., pp. 159-166.

Kahmen, H. and W. Faig (1988) Surveying. Walter de Gruyter & Co., New York.

Kavouras, M. (1982). On the Detection of Outliers and the Determination of Reliability in Geodetic Networks, Technical Report No. 87, Dept. of Surveying Engineering, U.N.B.

Kent, A. (1971) Information Analysis and Retrieval. 3rd Edition. Becker and Hayes, Inc., New York.

Kern DSR-11 Documentation (unpublished). Kern Company Ltd., 5000 Aarau, Switzerland. (1987).

Killen, J. (1983). Mathematical Programming Methods for Geographers and Planners, Croom Helm Ltd., Provident House, Burrel Row, Beckenham, Kent, BR3 1AT.

Klaver, J. (1984). "The Kern Computer Aided Mapping System: New Developments", International Archives of Photogrammetry and Remote Sensing, Vol. XXV, part A8, pp. 60-62.

Kligman, E.J. (1973). "How much does it cost ? Automated Traffic Control Software", Traffic Engineering, Vol. 44, No. 1, pp. 49-53.

Konecny, G. and G. Lehmann (1984). Photogrammetrie, Walter de Gruyter, New York.

Kratky, V. (1982). "Recent Progress in the NRC On-Line Photogrammetric Triangulation", Proceedings of the Symposium on Mathematical Models, Accuracy Aspects and Quality Control, ISPRS Comm. III, Helsinki, Finland, International Archives of Photogrammetry, Vol. 24-III, pp. 284-295.

Kratky, V. (1988). "On-Line Non-Topographic Photogrammetry". Non-Topographic Photogrammetry, 2nd Edition, ASPRS, 210 Little Falls Street, Virginia, pp. 107-127.

Kreiling Walter (1982) "The Software of the Analytical Plotting System AVIOLYT". Technical Papers of the ACSM-ASP Convention, Denver, Colorado, pp 228-237.

Lambourne, J.J. and F.R. Sutherland (1990) "A Vector and Raster Graphic Mapping Tool on Micro-computers". The Cartographic Journal, Vol. 27, No. 1, pp. 20-23.

Leatherdale, J.D. and K.M. Keir (1979) "Digital Methods of Map Production". Photogrammetric Record, Vol. 9, No. 54, pp. 757-778.

Lee, Y.C. (1988). "Information About Earth Integrated at Laboratory", Sightings, Progress In Research at the University of New Brunswick, Vol. 2, No. 4, P. 2.

Luenberger, D.G. (1969). Optimization by Vector Space Methods, John Wiley & Sons, Inc., New York.

Maddox, I.J. (1988). Elements of Functional Analysis, Second Edition, Cambridge University Press, Cambridge.

Makarovic, B. (1976) "Trends in the Development of Computer Supported Analogue Instruments". International Archives of Photogrammetry, Vol. XXI, part 3, section 2 07 194, pp. 1-30

Makarovic, B. (1982) "Integral Dynamic Performance Tests for Analytical Plotters". International Archives of Photogrammetry, Volume 24-11, pp. 30-41.

Makarovic, B. (1988) "Integrated Photogrammetric Systems: Working Group II-6 Report". International Archives of Photogrammetry and Remote Sensing, Vol. 27, part B8, pp 30-37

Maling, D.H. (1989) Measurements from Maps: Principles and Methods of Cartography. Pergamon Press Canada Ltd., Toronto.

Marble, D.F.; Calkins, H.W. and D.J. Peuquet (1984) Basic Readings in Geographic Information Systems. SPAD Systems Ltd., Williamsville, NY 14221.

Marton, G. (1984). "On-Line Analytical Aerial Triangulation Using POLIFIM System", International Archives of Photogrammetry and Remote Sensing, Vol. XXV, part A3b, pp. 678-680.

Masry, S.E. (1972) Real-Time Digitizing and Editing From Photogrammetric Instruments. Technical Report No. 12, Department of Surveying Engineering, University of New Brunswick.

McEntyre, G. (1979) Land Survey Systems. John Wiley & Sons, Inc., Toronto

Methley, B.D.F. (1986). Computational Models in Surveying and Photogrammetry, Blackie & Son Limited, Bishopbriggs, Glasgow, G64 2NZ.

Meyer, S.L. (1975). Data Analysis for Scientists and Engineers, John Wiley & Sons, Inc., Sydney.

Mikhail, E.M. (1976). Observations and Least Squares, Harper & Row Publishers, New York.

Moffitt, F.H. and E.M. Mikhail (1980). Photogrammetry, 3rd Edition, Harper & Row, Publishers, New York.

Moniwa, H. (1977). "Analytical Photogrammetric System with Self-Calibration and its Applications", Ph.D. Dissertation, Department of Surveying Engineering, U.N.B.

Monmonier, M.S. (1982) Computer-Assisted Cartography, Prentice-Hall, Inc., Englewood Cliffs, N.J. 07632.

Monmonier, M.S. (1985) Technological Transition in Cartography. The University of Wisconsin Press, 114 North Muray st., Madison, Wisconsin 53715.

Niedzwiadek, A.H. (1980) "Aspects of Data Storage and Retrieval Pertinent to the Use of Analytical Plotters". Proceedings, Analytical Plotter Symposium and Workshop, Reston, Va., pp. 193-200.

Olaleye, J. and W. Faig (1992). "Reducing Interior Orientation Time for photographs with crossing but non-intersecting Fiducial marks", Photogrammetric Engineering and Remote Sensing, Vol. 58, No. 6, pp. 857-862.

Papalambros, P.Y. and D.J. Wilde (1988). Principles of Optimal Design: Modeling and Computation, Cambridge University Press, New York.

Parker, D. (1990). Land Information Databases, Engineering Surveying Technology, Edited by TJM Kennie and G. Petrie, John Wiley & Sons, Inc., New York.

Petrie, G. (1981) "Hardware Aspects of Digital Mapping". Photogrammetric Engineering and Remote Sensing, Vol. 47, No. 3, pp. 307-320.

Petrie, G. (1990a) "Developments in Analytical Instrumentation". ISPRS Journal of Photogrammetry and Remote Sensing, Vol. 45, pp. 61-89.

Petrie, G. (1990b). Digital Mapping Technology: Procedures and Applications, Engineering Surveying Technology, Edited by TJM Kennie and G. Petrie, John Wiley & Sons, Inc., New York.

Petrie, G. (1990c) Analogue, analytical and digital photogrammetric systems applied to aerial mapping,.Surveying Technology, Edited by TJM Kennie and G. Petrie, John Wiley & Sons, Inc., New York.

Pierre, D.A. (1969). Optimization Theory with Applications, John Wiley & Sons, Inc., New York.

Pillmore, C.L., Denholm, K.S., Jepsen, H.S. and C.H. Schuch (1981) "Computer-Assisted Photogrammetric Mapping Systems for Geological Studies - A Progress Report". Photogrammetria, Vol. 36, pp. 159-171.

Raisz, E. (1962) Principles of Cartography. McGraw-Hill, Inc., Toronto.

Ramirez, J.R. (1989). "Understanding Universal Exchange Formats", Technical Papers, ASPRS/ACSM Annual Convention, Baltimore, Vol. 2, pp. 108-116.

Reece, C.J. and K.E. Kleinn (1980). "Data Required For Analytical Stereo Plotter Model Setup". Proceedings of The ASP Analytical Plotter Symposium and Workshop. Reston, Va., pp. 159-166.

Reeler, T. (1990). "Caris Server Interface", Universal Systems Ltd., Document No. SERVER-JAN90-VMS, 14p.

Richardus, P. (1984) Project Surveying. 2nd Edition, A.A. Balkema Publishers, P.O. Box 230, Accord, MA 02018.

Riley, H.D. (1980) "US-2 Analytical Stereoplotter". Proceedings, Analytical Plotter Symposium and Workshop, Reston, Va., pp. 87-99.

Roberts, T. (1989). "Recent Advances in Kern Photogrammetric Software", Technical Papers, ASPRS/ACSM Annual Convention, Vol. 1, pp. 243.

Rosove, P.E. (1967) Developing Computer-based Information Systems. John Wiley & Sons, Inc., New York.

Ross, D.A. (1978) Opportunities and Uses of the Oceans. Springer-Verlag Inc., New York.

Sabins, F.F. (1987) Remote Sensing: Principles and Interpretation. Second Edition, W.H. Freeman and Company, New York.

Schut, G.H. (1964). "The N.R.C. Program for Analytical Aerial Triangulation on the IBM 1620 and the IBM 650", Division of Applied Physics, NRC, Canada, NRC-8310, 34p.

Schut, G.H. (1966). "An Introduction to Analytical Strip Triangulation, with a Fortran Program", Division of Physics, NRC, Canada, NRC-9396, 86p.

Schut, G.H. (1973). "An Introduction to Analytical Strip Triangulation, with a Fortran Program", Revised Edition, Division of Physics, NRC, Canada, NRC-13148.

Shneiderman, B. (1987). Designing the User Interface: Strategies for Effective Human-computer Interaction. Addison-Wesley Publishing Company, Inc., California.

Slama, C.C.(1982). "Generalized Software For Analytical Plotters". Proceedings, Commission II Symposium On Advances In Instrumentation For Processing And Analysis Of Photogrammetric And Remotely Sensed Data, Ottawa, pp. 102-109.

Sonnenberg, G.J. (1988) Radar and Electronic Navigation, 6th Edition. Butterworths & Co., Ltd., Toronto.

Sprinsky, W.H. (1987) "Transformation of Positional Geographic Data from Paper-Based Map Products". The American Cartographer, Vol. 14, No. 4, pp. 359-366.

Stirling, D.M. (1990). Close-range photogrammetry, Engineering Surveying Technology, Edited by TJM Kennie and G. Petrie, John Wiley & Sons, Inc., New York.

Swain, P.H. and S.M. Davis (1978) Remote Sensing: The Quantitative Approach. McGraw-Hill Inc., New York.

Szekielda, K. (1988) Satellite Monitoring of the Earth. John Wiley & Sons, Toronto.

Tetley, L. and D. Calcutt (1986) Electronic Aids to Navigation. Edward Arnold Publishers,Ltd., 3 East Read St., Baltimore, Maryland 21202.

Thomson, D.B.; Wells, D.E and W.H. Falkenberg (1979) An Introduction to Hydrographic Surveying, Lecture Note No. 53, Department of Surveying Engineering, U.N.B.

Tomlinson, R.F. (1988) "The Impact of the Transition from Analogue to Digital Cartographic Representation". The American Cartographer, Vol. 15, No.3, pp. 249-261.

Unwin, D. (1981) Introductory Spatial Analysis. Methuen & Co.,Inc., New York. NY 10017.

Vanicek, P. and E.J. Krakiwsky (1986) Geodesy: The Concepts. 2nd Edition, Elsevier Science Publishing Comp., Inc., New York, NY 10017.

VAX Documentation (1984). "Fortran Programmer's Manual", No. AA-Z212A-TE, Unpublished.

Wall, C.T.C. (1972) A Geometric Introduction to Topology. Addison-Wesley Publishing Company, Ontario.

Wells, D.E. and E.J. Krakiwsky (1971). The Method of Least Squares. Lecture Note No. 18, Department of Surveying Engineering, U.N.B.

Wilson, B. (1984) Systems: Concepts, Methodologies, and Applications. John Wiley & Sons Ltd., New York.

Wolf, P.R. (1983). Elements of Photogrammetry, with Air Photo Interpretation and Remote Sensing, 2nd Edition, McGraw-Hill International Book Company, Inc., London.

Wong, W.K. (1980). "Basic Mathematics of Photogrammetry", Manual of Photogrammetry, Fourth Edition, American Society of Photogrammetry, 105 N. Virginia Ave., Falls Church, Va. 22046.

Woolnough, D.F. (1973). A Fictitious Data Generator for Analytical Aerial Triangulation, Technical Report No. 26, Dept. of Surveying Engineering, U.N.B.

Yzerman, H. (1979) "Data Acquisition by Analytical Photogrammetry". Photogrammetric Record, Vol. 9, No. 54, pp. 779-785.

Zarzycki, J.M. (1978) "An Integrated Digital Mapping System". The Canadian Surveyor, Vol. 32, No. 4, pp. 443-452.

Zutter, A.B. (1984). "MAP 300: A Stand Alone Interactive Graphics Station", International Archives of Photogrammetry and Remote Sensing, Vol. XXV, part A4., pp. 550-557.

VITA

Candidates full name: James Olaleye

Place and date of birth: Ibadan, Oyo State
6th December 1953

Permanent address: S5/160, Odinjo, Ibadan
Oyo State, Nigeria.

Schools attended (with dates):

Baptist Day School, Otun, Saki
1958-1964.

Adekile Goodwill Modern School,
Aperin Ibadan, 1966-1967.

Adekile Goodwill Grammar School,
Aperin Ibadan, 1968-1972.

Federal School of Telecommunication,
Oshodi, Lagos, 1973-1976.

Universities attended (with dates and degrees obtained):

University of Lagos, Akoka-Yaba, Lagos,
Nigeria, 1976-1981;
B.Sc. Surveying Engineering.

University of Lagos, Akoka-Yaba, Lagos,
Nigeria,1982-1984;
M.Sc. Surveying Engineering.

University of New Brunswick,
1987-1992, Ph.D Candidate.

Publications:

1986. "History of Photogrammetry in Nigeria", Proceedings of the Symposium on Photogrammetry and Remote Sensing in Economic Development, Commission VI, International Society for Photogrammetry and Remote Sensing (ISPRS), ASCON, TOPO, Badagry, Lagos State, Nigeria, 22-26 sept., pp. 121-125. (with O. Ayeni and A. Ogbechie).

1988. "Comparator Calibration: A Comparison of polynomial and non-Polynomial calibration Models".ACSM-ASPRS Fall Convention, Virginia Beach, Sept. 11-16, pp. 262-270.

1991. "Choosing the Optimum Software Configuration for the Kern DSR-11 Analytical Plotter in an Integrated Spatial Information Environment", Technical Papers, *GIS/LIS '91, ACSM-ASPRS Fall Convention, Atlanta Georgia*, Oct. 28 - Nov. 1, pp. B86-B95. (with W. Faig)

1992. "Integration of the Kern DSR-11 Analytical Plotter into a GIS", *Intern. Archives of Photogrammetry and Remote Sensing*, Vol. XXIX, Part B2, pp. 590-595.

1992. "Reducing Interior Orientation Time for photographs with crossing but non-intersecting Fiducial marks", *Photogrammetric Engineering and Remote Sensing*, Vol. 58, No. 6, pp. 857-862 (with W. Faig).