

XIV Congress of the International Society for Photogrammetry
Hamburg 1980

COMMISSION II
WORKING GROUP II-3

PRESENTED PAPER

Rune Larsson

Department of Photogrammetry
Royal Institute of Technology
Stockholm, Sweden

THE CONSTRUCTION OF A COMPUTER-ASSISTED PHOTOGRAMMETRIC SYSTEM

ABSTRACT

When photogrammetric organizations are interested in modernization of their equipment, they are often faced with the fact that they are forced to use their existing photogrammetric instruments in connection with new computer equipment. Here some of the experiences gained from building such a system are discussed in general. Different aspects of the problem are: Central or distributed computer intelligence, interactivity, feedback of information to operator, system sensitivity to computer breakdown and the impact of microcomputers. Conclusions are drawn from two different points of view: The educational institution and the photogrammetric organization.

0 SUMMARY

During the years 1975 to 1979 the author was working with the task to computerize the photogrammetric instruments of the Department of Photogrammetry, Royal Institute of Technology, Stockholm, Sweden. The system which was built is taken as an example demonstrating different ways of designing computerized photogrammetric systems. The goal has been to achieve a system, capable of producing photogrammetric work economically and efficiently and also to serve as an instrument for education, research and development in the the department.

1 PHOTOGRAMMETRIC EQUIPMENT

The organisation in which this work was done is an educational institution where the instruments are used for research, development, education and commissions. The instruments are: Wild A7 stereoplotter from 1953, Wild A8 stereoplotter (1958) and Wild StK-1 stereocomparator (1963). Although they are much used and comparatively old, they are still very accurate and are well suited for their task. Instrument development has mechanically indeed not advanced significantly in the 27 years since our A7 was bought, with a few exceptions. Moreover, our instruments represent a great value for us, and it would be very expensive for us to replace them with new equipment. We are thus interested in how to modernize them in other ways.

2 COMPUTERS

I will not go into detail in the development of the multitude of computers we are familiar with today, from the large main-frame to the one-chip-microcomputer. In photogrammetry we have followed the development through its phases, and we now use all different sorts of computers. We need number-chrunchers for the adjustment of blocks, calculations of DTM's, digital image processing and so on. We use in-house minicomputers for smaller tasks like calibrations, orientations, stripformations, resections and sometimes also for bigger tasks e.g. block-adjustments. We use them also for administrative data treatment. The microcomputers, if I may collect them to one group, are finally used in terminals, tabletop computers, digitalization equipment, plotters, personal computers and what you have.

3 HARDWARE SYSTEM

Presently our system consists of the following hardware:

- Central Processor (CPU) General Automation 16/440
- Core Memory 64 k 16-bit words
- Disk Memory 5 plus 10 Megabytes on four platters
- Three terminals, VDU's, one of which is graphical, 2 micros
- Drum Plotter Calcomp 1037, 80 cm wide, 1 micro
- Digitizer for 4 axes, used with Wild StK-1 comparator, 1 micro
- Digitizer for 3 axes, used with Wild A8 stereoplotter, 1 micro
- Line printer serial, matrix, 160 ch/s, bidirectional, 1 micro
- Two teletypewriters, 10 and 30 ch/s
- Paper tape input and output.

This system was gradually designed from 1975 and on. As you can see from the list, there are at least six microcomputers contained in the peripheral equipment. They are dedicated microcomputers, designed to remove work from the central CPU. This is an important way to avoid overhead work for the CPU. The principles are that time-consuming, simple tasks should be left to cheaper dedicated computers whenever there are a possibility to do so.

As an example I choose the method of digitizing when you want a sample, say every 10th meter in the terrain. When you proceed in the model, the computer must keep a track on where you are. To do so, co-ordinate data must be streamed into the computer and scaled at a sufficiently high rate to allow the CPU to choose the sample which is closest to 10 meter from the previous sample. This task is time-consuming and simple. It should be solved by a dedicated microcomputer in the digitizing equipment, which send the scaled co-ordinates to the central CPU at the proper intervals. The central CPU has more important tasks to do during the time. This is of course only valid when you have a multi-tasking operating system, that is when you want to serve more than one instrument with one computer system.

4 OPERATING SYSTEMS (OS)

There are different sorts of operating systems for computers. The single-user or single-tasking operating system serves to support one user with his work in an optimal way. A considerably more complicated system is the real-time OS, where multiple tasks are running separately, scheduled according to rules, defined by the programmer prior to the start of the execution. The tasks can execute in a series, on a certain date and time or at the completion of an input or output operation. One application could be that one separate task is allotted to each stereoplotter and are activated for each new input from the plotter. When the task is finished, it will delay itself until the next input arrives. The work to be done could be to transform the co-ordinates to a reference system and store the result on disk memory or magtape.

This type of configuration can be used by a photogrammetric organization. There are several advantages: because each task is programmed and scheduled to one or a few operations, the efficiency of the system will be high. No CPU work is done when it is not needed, and the overall performance of the system should be high. There are also drawbacks. To be able to serve the operator of the plotter, each task must be programmed for several different types of work, e.g. relative orientation, absolute orientation, digitizing. They have to be predefined and programmed before they can be executed and tested under the real-time system. Every operation the operator needs to do, must be thought of and programmed accordingly. This means that the development of these routines are quite laborious. Real-time programs are therefore optimal when there are not too many different operations to perform, when you do not expect to modify these operations too often and when you want to serve as many instruments as possible with one central computer.

The next type of operating system I will discuss is the time-sharing system. They are today more used than real-time systems, due to their greater flexibility. We can explain their principle of work by saying that each user is running a real-time task of special type. That task is programmed to respond to commands like the single-user OS responds to. It can thus load and execute programs, assign peripheral units, perform program development i.e. interpret, compile and link programs. Each user is served promptly because for each second they are all allotted one fraction of a second to execute, if they have anything to execute. If a task is waiting for input from the terminal or from the plotter, its time-slice is given to another task. Time-sharing systems are possible to implement because CPU activity in single-user OS are very low, only a few percent of the time.

Now, what are the advantages with a time-sharing system? First of all, it is more flexible. You are not restricted to the execution of one program which must do everything for you. Instead you can finish one program, exit from it, and then you return to a monitor, which allows you to choose among several programs which you can load and execute. If your real-time program comes to a halt by a mistake or other misfortune, the operator will often be left sitting at a dead terminal, not able to do anything until the systems programmer comes and makes a restart. Often data are lost and measurements must be repeated. Also restart often forces the operators to finish their work. If your program stops in a time-sharing OS, the operator can always interrupt the system, return to monitor level, and restart the program without disturbing the work at the other instruments. Time-sharing OS are naturally better if you have a lot of program development to do.

Now, these advantages must be paid, and you pay it by losing some of the efficiency of the real-time OS. You will have a greater CPU overhead at any given configuration. This means, to serve a number of work-stations, you must buy more computer power. This becomes less and less important now when the disk memories are getting faster and bigger. However, there will always be a difference because of timing and swapping problems.

5 INTERPRETING TIME-SHARING SYSTEMS

The interpreting system works in a different way than ordinary compile-link-go systems do. In an interpreter, the source program is the prime source of information, also for the computer. The interpreter scans the program, and for each new line of text, it produces the machine-code that performs the operations stated in the line. The machine-code are not saved. Interpreting systems are well suited for program development, because of the quick way you can modify and run the program, without the need to compile and link the program for each new modification. The source text in an interpreter are often stored in a very space-saving way, which means that you need less core for each given operation. This also simplifies the construction of time-sharing systems, because often all users can share the core memory without the need to swap their programs to and from the disk.

The drawback of interpreting systems are mainly due to the interpreting work, which must be made for each new execution of a certain source statement. The speed of execution is therefore lower by a factor 10-20 of the speed of a compiled machine-code. If the computer has a lot to do for each sample the execution time may become too long for productive work. If you can define what tasks you will perform with your computer and what demands on response-time you have for each task, you will be in a better position to evaluate different systems in this respect. A good solution to this problem is that some computers offer both an interpreting and a compiling system for the same source language. This means that you will be able to make program development using the advantages of the interpreting system, and when the programs are tested, you can execute them as compiled and thus avoid all the overhead of the interpreting process.

6 THE PHOTOGRAMMETRIC SOFTWARE

With the hardware and system software discussed above, you can design and program your own application programs. There are, however, some problems which has to be considered first. If there are several instruments to be served by a central computer we must ask ourselves what will happen if and when the computer stops abnormally. Which data are destroyed? How long is the time for restart? If you have trouble with the central computer, how will that affect the work or is the production stopped until the computer works again? The solution to these problems are different depending on the type of organization involved.

For the educational institution the loss of computer power are less serious, because there are seldom time-critical commissions to deliver at a certain time. In our department the computers are mainly used for program development, testing, diploma work by students and so on. If the computer halts for one or two days, people can often work with coding, system work, writing reports or other occupations until the computer is put into order again. This is not the case for mapping organizations, where the prime problem is to get the products delivered on time. This of course affects the design of the system.

What can you do to avoid computer halts? One way is to duplicate the critical parts of the central computer, so that one unit can take over work if the other is under repair. This will raise the availability of the computer system, but in a rather expensive way.

7 THE DISTRIBUTED COMPUTER SYSTEM

The solution to the problem of computer stops which will halt the work for several operators, is to make it possible for them to continue work without the support of the central computer. If each instrument has a sufficiently intelligent digitizing unit, or it is connected to a local general microcomputer system with local data-storage possibilities, the work will be able to continue, and data will be saved locally until the central computer is able to receive it again. At our department, we are designing a system for large scale semi-automatic mapping. In

the present design, the map is generated instantly on a graphical screen beside the instrument. The interactive work is performed by the central computer, and the map data are saved on a map data base on disk.

This system works well as long as the computer is running. In the final design, the interactive work should be programmed on a microcomputer and run individually at each instrument. Only when the map data blocks are dumped into the central map data base the central computer has to work. This does not only reduce the work for the central computer, it also makes a much safer system. If one microcomputer halts, only one instrument will be unproductive. If the central computer halts, the satellites will be able to store the map data locally. Finally, if you have some satellite computer in store, you might replace a faulty unit and avoid halts.

To build a distributed map data processing system is now economically feasible thanks to the low cost of personal computers and potent table-top computers. The local data storage system should preferably be on flexible disks, which are cheap and fast and can store considerable amounts of data.

Now, why do we need the central computer? This must be a question to be solved individually. It is useful to have a larger computer for heavy tasks, like administration of the data base, adjustment of blocks, calculations of DTM's and the like. When it comes to the potentiality of the satellite computers, it depends mainly on how complicated the dialogue is and what the demands are for reply-times.

FINAL REMARKS

As in all system work, the individual solution is dependent on several presumptions and restrictions pertinent to the different installations. This discussion is a result of a successive development which has led to a certain final suggestion for the system. As in all development, the final system is never reached, it is always changed. This paper is a way of presenting my experiences in this field. A final advice is that you should let the professional programmers design the system software and let photogrammetrists program the photogrammetric software.