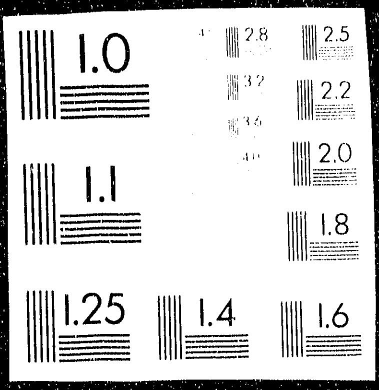
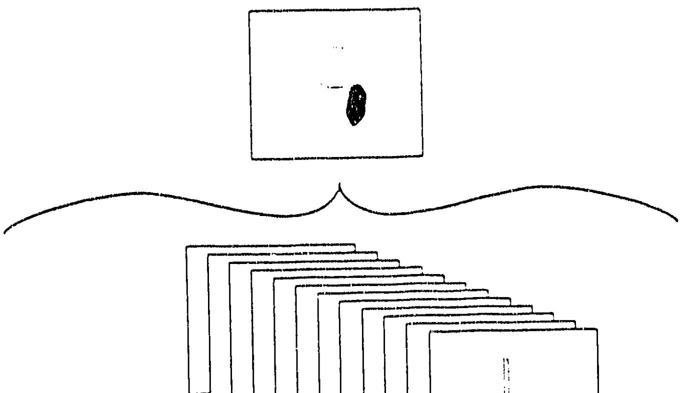
1 OF 1





Composite Feature Image

Feature Images

R91 0684

Idaho
National
Engineering
Laboratory

Managed
by the U.S.
Department
of Energy

EGG-PHY-10134

October 1991

Imaging System (INEL) FY-91 Summary Report

J. K. Partin
G. D. Lassahn
J. R. Davidson

EG&G Idaho

Work performed under
DOE Contract
No. DE-AC07-76ID01570

EGG-PHY--10134

DE92 012366

Imaging System (INEL) FY-91 Summary Report

J.K. Partin
G.D. Lassahn
J.R. Davidson

HQ PROJECT MANAGER - Steve Herrick
PROJECT NUMBER - ST474E

October 1991

Idaho National Engineering Laboratory
EG&G Idaho, Inc.
Idaho Falls, Idaho 83415

Prepared for the
U. S. Department of Energy
Office of Arms Control
Under DOE Field Office, Idaho
Contract DE-AC07-76ID01570

ABSTRACT

This progress report gives a brief description of the general automatic target recognition system algorithms developed for this project, and also summarizes the progress in fiscal 1991. An appendix discusses the proposed computer hardware for this system.

CONTENTS

FIGURE

TABLES

Imaging System (INEL)
FY-91 Progress Report

INTRODUCTION

This Imaging System project is a cooperative effort among several laboratories to develop an airborne system comprising several types of imaging sensors, to detect the presence of certain not-yet-specified objects (which we will call targets) on the ground. Desirable features of this system include (1) minimal requirements for human intervention, and (2) ability to distinguish among similar but different targets; thus, this is a multisensor Automatic Target Recognition system.

The several cooperating laboratories are addressing different aspects of producing this system. The Idaho National Engineering Laboratory (INEL) is working on designing computer hardware and software that can input images from the several sensors and output a list of what targets are present, the target locations, and the uncertainty or confidence in the identification of each target. (The INEL group has extensive experience in developing specialized sensor/measurement packages, including optical sensors and systems, analog signal processing, and digital data acquisition hardware. Specific examples include a system developed to map the Three Mile Island reactor core and a terrain perspective viewing system developed for the US Air Force to use in various types of imagery and sensor data for mission briefing.) Two special problems in the software development are fusing the data from the several sensors and estimating target identification uncertainties.

ACCOMPLISHMENTS

Specific accomplishments to date include:

1. Review literature
2. Attend short courses
3. Select a general approach to data analysis
4. Select first phase hardware
5. Partial completion of first phase software

These items are discussed in the following sections of this report.

Literature Review

There is a vast quantity of literature relevant to automatic target recognition data analysis. We have compiled a bibliography of 179 books and journal articles that we have read or expect to be important, but this is only a small fraction of the available literature. The literature indicates that there are several general techniques used for various aspects of image analysis and automatic target recognition, and most of the relevant journal articles describe minor variations or improvements in detail of these methods. To review all the literature would be very time-consuming and not very productive. The reasonable approach is to be aware of the general methods; select one general method of automatic target recognition; and, review in detail only that part of the literature that can help with our particular method. Taking this approach, we are not continuing with an exhaustive literature survey, but we will pursue only those literature items that we feel will directly benefit our development efforts.

Short Courses

One of our team has attended two short courses sponsored by SPIE: Digital Image Processing Fundamentals presented by Mohan M. Trivedi of the University of Tennessee, and Advanced Concepts in Automatic Target Recognition presented by Hatem N. Nasr of Honeywell Systems and Research Center. These courses were of a broad overview nature, describing general methods without taking time to delve into details of implementation. They were very valuable in that they ensured that we are aware of the current state of the art and are not overlooking any potentially valuable techniques for automatic target recognition.

General Approach Development

Obviously, the choice of the general framework of the automatic target recognition data analysis is of fundamental importance, since this choice determines the ultimate capabilities and limitations of the system. We have chosen a general approach that is simple enough to be implemented with a minimum (but still significant) effort and the limited resources available in this project, but can also be expanded to a much more sophisticated and powerful system as additional resources are available. The basic system requires minimal image registration; it requires no segmentation or clustering calculations; it does not require detailed knowledge about the targets; it requires minimal (but still significant) computer power for execution in the field; and, it automatically accomplishes image fusion and uncertainty estimation. On the negative side, this basic system does require a training process, which would benefit from having a large number of images like those that will be obtained in field applications; and, this system is not expected to be very powerful in its ability to discriminate between targets and other objects that look a lot like targets. We will refer to this system as the Phase 1 system, on the assumption that it may someday be expanded to a more powerful system with better target discrimination capabilities.

This Phase 1 system is a statistical target recognition system. From a given set of scene images, the system extracts local features (lines, edges, speckle intensity, etc.), combines the several local features in an optimal manner that is determined by a prior training process (thus accomplishing image fusion), and decides on the basis of statistics from past training information whether this combination of features indicates the presence of a target in any relatively small, fuzzy region of the scene. The estimated uncertainty of the result was already calculated in the training process. This system is described in more detail in a later section of this report.

Hardware Selection

The best choice of computer hardware for the final automatic target recognition system will depend partly on which data analysis methods are used. Our feeling is that it would be better to delay this final hardware selection until the software development is much more complete. For the Phase 1 development effort, we have chosen to use a set of 17 T800 transputers (parallel processors) that can reside on either an IBM PC or a Sun workstation as the host machine. There are several reasons for this choice:

1. The transputers and host computers are immediately available at no cost to this project.
2. These transputers are general purpose devices, not having the restrictions that might be expected in specialized image analysis hardware systems. This allows greater flexibility in experimentation and selection of algorithms.
3. The transputer system is quite fast. It may not be as fast as specialized image processing hardware systems, but it is much faster than a single processor system and is certainly adequate for the development phase of this project.
4. The software developed for the transputer system should be easily convertible to other hardware systems, since it is written in standard FORTRAN except for some added subroutines that allow communication between transputers.

Currently, we are using an IBM PC as the host computer, because it is more convenient for the programmer and for access to currently available image data. With this host, the major limitation of the transputer system is the time required to move the large amount of image data between the host disk and the transputer memory. We expect this problem to be less significant with the Sun used as the host computer.

Software Development

We have developed two major programs, called A and B, and some minor incidental software for the phase 1 automatic target recognition system. Program A extracts local features (such as edges, lines, speckle intensity) from images. This program consists of a basic framework to which can be added modules to select whichever local features are desired. This program A is complete and has been exercised in preliminary development applications. It requires about 2.4 seconds to apply a 5x5 pixel convolution operator to a 512x480 pixel image. Its slowest part is the input and output of images to disk files, requiring about 18 seconds per image. This input/output process can hopefully be speeded up by using a Sun workstation instead of an IBM PC as the host machine.

Program B will calculate the optimum values of the coefficients used to combine feature images to give the composite feature image that indicates the presence or absence of targets. This program B is partly finished; it is capable of giving the first estimate of the optimized coefficient values, but it cannot yet do the fine adjustment of these values. The program B version currently under development is limited to one single training scene (there may be multiple sensor images of this one scene); it will be expanded later to allow for multiple training scenes.

Preliminary tests of this software indicate that this general approach to automatic target recognition does have some real potential. The method can easily distinguish between different types of image structure, such as the detailed engraving and the simple block letters printed on a dollar bill. As expected, the method does require a good initial selection of a set of local image features, which may not always be trivial.

PHASE 1 DATA ANALYSIS SYSTEM DETAILS

Functional Description of Phase 1 Data Analysis

There will be several different sensors, each of which will record an image of a scene which may include certain target objects that have previously been specified as being interesting. The sensors might be, for example, a synthetic aperture radar system, a visible light camera, and an infrared camera.

The images from the several sensors will differ in two important respects: (1) a single object will look different to different sensors, and (2) the images will (usually) represent different perspectives, because the sensors will view the scene from different points.

The problem of different perspectives in the several scene images is common. "Image registration" is a term used to describe the process of using geometrical transformations to distort the several images so that they all represent the same perspective. An approximate registration can be done using information about the properties of the sensors and their locations and orientations when the scene images were acquired. Such an approximate registration will probably be adequate for the first (Phase 1) automatic target recognition system being developed in this project. Improvements in the system will require more careful image registration, which will use information from the images themselves to aid in the registration process.

Even if the scene images are in perfect registration, a given object may appear very different in the several scene images. An infrared sensor, for example, may see a warm truck engine as a very bright spot, but may be completely insensitive to a tree that would be obvious in a visible light image. In a radar range image, a house may have a shape very different from that in a visible light image. These differences represent a very important advantage in image analysis, in that they allow us to better distinguish between very different kinds of objects that may appear very similar to one single sensor. This is, of course, the reason for using several sensors. The system developer must be aware of these differences and allow for them in designing the automatic target recognition system. For the particular type of system being developed here, the developer must know which local image features are important in which types of image. For example, regions of uniform intensity may be important in infrared images but not in radar range images. Sharp edges may be a very useful feature in high resolution visible light images, but may never occur in an image from a low resolution sensor.

After the several scene images are registered, we do a feature extraction process. The relevant features of a scene image may be such things as local average intensity, straight line segments, sharp steps in intensity (edges), or speckle amplitude. These features are all local, in that they are defined by a region no larger than the targets being searched for. Different scene images from different sensor types will probably have different sets of important local features. For each feature from each scene image, we construct a "feature image", which is simply a map indicating how much of the feature occurs at which locations in the scene image. We then form a weighted sum of all the feature images to produce one "composite feature image". Ideally, this final image will have bright regions where there are targets and dark regions elsewhere. This final image will give essentially no information about the details of shape or orientation of the targets; it will only indicate the presence or absence of targets.

Note that after the feature images are formed, they are all treated on an equal basis, regardless of which scene images they came from. That is, all of the useful information (for this particular target recognition process) has been extracted from the scene images and put into one pool where it is treated equally. Thus, image data fusion has been accomplished.

This software system must be taken through a training process before it can be used to search for targets in field images. In the training process, the system is presented with numerous (perhaps a hundred) images like those expected in the field, including some target and some background. The operator divides each training image into three regions, representing target, background, and unspecified. The training part of the software then selects weighting coefficient values that minimize the target identification error rate. These coefficients are used to form the composite feature image during the analysis of field data. The training procedure also provides an estimate of the target recognition error rate.

Although this Phase 1 system has limited capabilities in terms of detailed target discrimination, we expect that it will be a useful system. We feel that development of a better system may not be possible with the current limitations of time, budget, and information about specific application of the system. However, if these limitations should change, this Phase 1 system would be ideal as the first step of a more sophisticated target recognition system. It should be quite easy to add other modules to this software to make a target recognition system that would be more specific and accurate.

In summary, this Phase 1 automatic target recognition system has the following advantages and disadvantages:

+ image data fusion occurs automatically
+ field data analysis is fast and simple
+ uncertainty estimates are provided
+ the system can be extended to improve accuracy
- training is required
- target recognition is general, not specific

Image Registration

The images from the several different sensors may not include exactly the same scene and may not be taken from exactly the same angles. It will usually be necessary to transform the images so that they do represent the same scene, on the same scale, viewed from the same point in space. This process of making the several images conform to the same geometry is called image registration.

An approximate registration can be done using information about the known camera locations and orientations when the images were recorded. More precise registration requires using the content of the images, and making corresponding features line up in the several images. The details of this process depend strongly on which types of images are being registered, because a given physical feature may produce very different image features for different kinds of sensors. Development of precise image registration algorithms may be a very difficult task.

The basic automatic target recognition system being developed here does not require precise registration. However, if the system is developed further to enhance its capabilities, precise registration will probably be required. For the moment, we will only worry about the coarse registration that can be done with information about camera locations and orientations.

Selection of a Set of Features

This basic automatic target recognition system depends on the system developer selecting a set of image features that is adequate to distinguish between targets and background. Although there may be some basic features that almost everyone would include, the process of selecting an adequate set of features is something of an art and requires insight and understanding of the sensors used to acquire the scene images. We have started this process, first considering features that can be extracted using correlation or convolution calculations in local regions of the images. The mathematics and understanding of this type of feature extraction process is greatly enhanced by using orthonormal operations, and we have developed orthonormal operators that extract features that represent intuitively simple structures such as edges and lines. These same orthonormal operations can also extract features that may have less obvious intuitive meaning but are just as important as the more intuitively desirable features. In fact, one of the advantages of this type of operator is that one can always construct a set that is complete, so that any possible operator of this type can be represented by the chosen set.

In addition to the just-mentioned convolution operators that extract intuitively meaningful features such as edges and lines, we have worked with simpler convolution operators based on orthogonal polynomials. Although these polynomial operators do not have the obvious intuitive appeal of the other operators, they are surprisingly effective at separating different types of image structure. These will be evaluated again when we have real data with images of real targets in real backgrounds.

The general data analysis approach is in a sense linear, in that the composite feature image is a linear combination of the individual features. This may seem to be a serious limitation, since some target/background distinctions may be easy with non-linear feature combinations but impossible with linear feature combinations. This apparent limitation can be circumvented by defining additional features that are non-linear combinations of the basic features, such as simple products of basic features.

Of course, there are other types of operations, other than the local correlation operations, that may be very useful in extracting important features for target recognition. We are using our own creative resources and also delving into the literature to find potentially useful local feature extraction operations.

Training Software Development

We have developed the algorithm for training the system; these procedures have yet to be programmed. The algorithm is quite straightforward conceptually, but the large amount of data involved may make the practical implementation of the algorithm a little awkward. The basis of the algorithm is the minimization of the rate of errors in target recognition. This error minimization criterion determines the weighting of each local feature in deciding whether a target is present.

Ideally, the training process would use a large number (perhaps hundreds) of images that include both targets and background, with conditions spanning the full range of conditions to be encountered in field deployment of the system. In practice, it may be possible to train the system using only a few images or using an operator's guesses about what the images would look like if they were available, but one would not expect the results from such limited-data training to be as good as the results from training based on a large data set.

Field System Development

The program to be used in the field after the training is done will be much smaller and will be mostly included in the program used for training the system. Thus, development of the field-deployable part of the system should be quite easy, requiring little more than the development of good man-machine interface software.

CONCLUSIONS

We have made significant progress on the development of data analysis software and hardware for the automatic target recognition system. The Phase 1 system is simple enough to fit into the current limitations of development resources, but has the potential for expansion later. Preliminary results from the completed parts of this system give some confidence in the ultimate capabilities of this general approach.

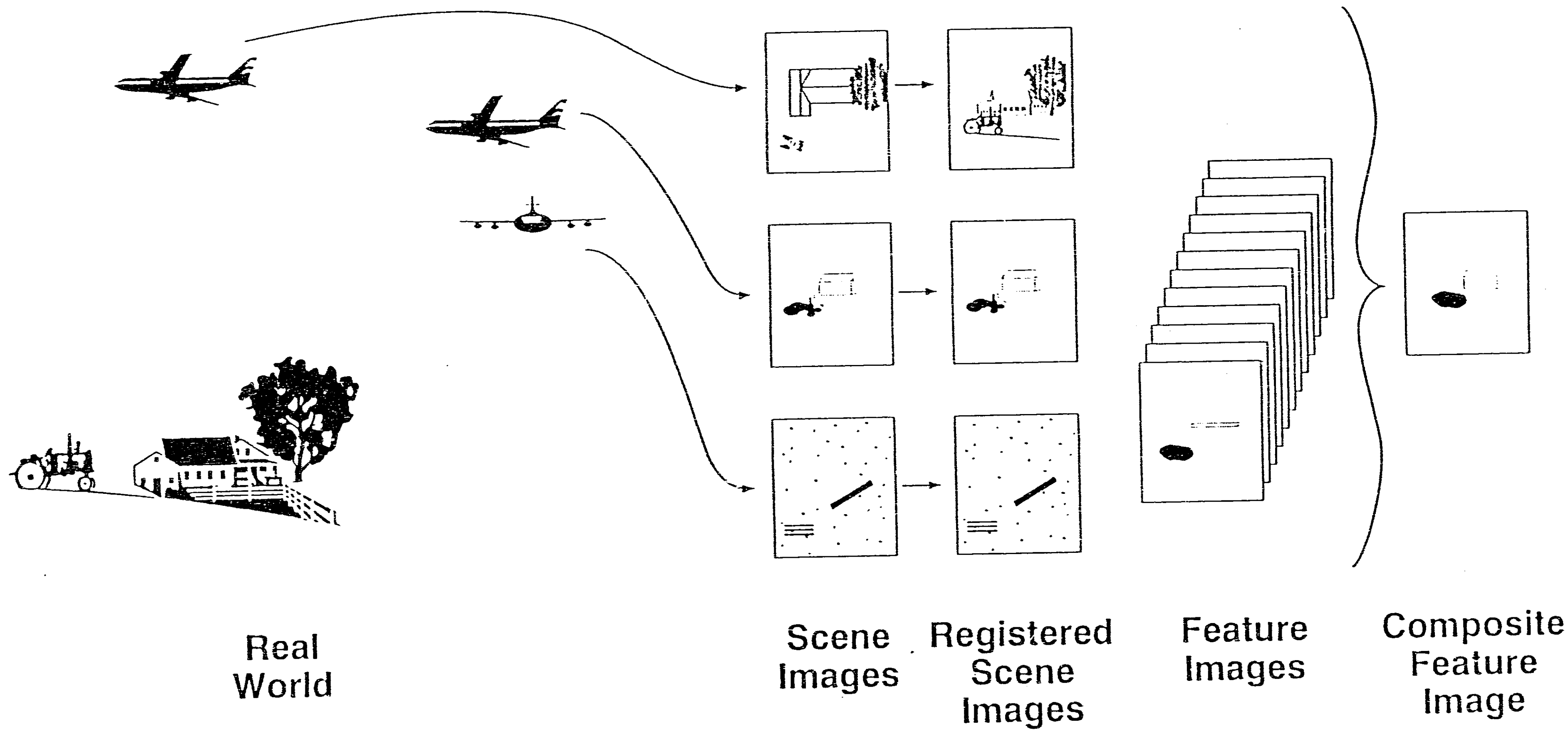


Figure 1. Imaging System Schematic

R91 0684

Appendix A

Proposed Hardware

Appendix A

Proposed Hardware

The Idaho National Engineering Laboratory part of this Department of Energy Imaging System project is to develop a computer software and hardware system to analyze raw image data and report the presence of interesting objects in the imaged scenes. This appendix contains recommendations for a computer hardware system for the image analysis.

The primary characteristic of image analysis is that it involves manipulating large amounts of data. This immediately implies the need for large mass storage devices and perhaps a large amount of fast memory. The data processing is for the most part quite simple, usually not involving complicated calculations. However, because of the large amount of data, even simple calculations require a significant amount of computer processor time. Therefore, after large data storage capabilities, fast processing is an important consideration.

The preliminary development work for this Automatic Target Recognition (ATR) system was done with an IBM AT Personal Computer serving as a host for a set of 17 T800 Transputers used as a parallel processing system. (We refer to this system as an ATR despite the fact that we are not seeking targets for a weapons system, because all the requirements for the image analysis and object recognition system in this project are precisely the same as the requirements for ATRs for which there is a large body of literature and established terminology.) We have been working with images comprising 512 x 480 pixels, stored on hard disk using 1 byte per pixel. Although this system is not state of the art, and the image sizes used here may not be the same as those to be used later, we can still get a reasonable feeling for speed and memory requirements by looking at results from this preliminary development system.

The ATR system as it now exists can be considered as a sequence of several major operations:

1. Read 3 raw images from hard disk.
2. Calculate approximately 20 feature images.
3. Calculate a result image.
4. Write the result image to hard disk.

This is the "production mode" or field operation mode. This must be preceded by a training process which does not need to be done often but which requires many more raw images and more processing per image. The major operations for the training process are:

1. Read tens or perhaps hundreds of raw images from hard disk, 3 raw images and 1 mask image per training scene.
2. Calculate hundreds of feature images, approximately 20 per scene.
3. Write all feature images to hard disk.
4. Read all feature images from hard disk.
5. Calculate coefficients to be used in production mode.

The training process also requires an operator to mark "target" and "background" regions to create a mask image of each scene used during training. This of course may require a large amount of time, but in practice the speed of this operation is not limited by the computer system so we do not include it here.

SPEED REQUIREMENTS

Table A-1 gives estimated times for typical production mode operation. The entries in this table are explained in the following text.

About 18 seconds is required to read a single image from the PC hard disk into the first transputer, with the analysis software in its current configuration. It is possible to optimize the analysis software to reduce this image read time to less than 10 seconds. It is apparently possible to read an image from hard disk in less than 2 seconds, if we use different host interface software to interface between the host PC and the transputers; we do not have such software, and it may require a substantial effort to develop.

About 0.57 second is required to distribute a single input image among the several transputers. This number is practically independent of the number of transputers used. Because of the parallel nature of the transputer processing, most of this distribution time can overlap the time to input the image from the disk. Thus, with optimized analysis software, we will require about 10 seconds to input a single image and distribute it among the transputers. If we really use 3 images per scene (from 3 different sensors), we can expect to take about 30 seconds to read the raw data for one scene from hard disk. If the number of images per scene increases, the raw data read time will increase proportionally.

There are hard disk systems (and perhaps other mass storage devices) that are much faster than the PC hard disk used in the current system. Use of one of these faster hard disks, with the appropriate bus and software, could make the disk read time less than the time required to distribute the image among the several T800 transputers, so that the effective total image input time would be essentially the time required for transputer intercommunication, about 0.57 seconds per image. One possibility is to access a fast disk drive via one of the transputer intercommunication links instead of via the usual system bus. A gigabyte disk and a transputer interface could cost as little as $3000. This approach has the disadvantage that the fast disk would not be directly accessible by the host computer, so that the initial input of raw images might be a little awkward and time-consuming. This option is described as "external disk" in Table A-1. A second possibility is to use a host computer that is faster (has a faster bus) than the IBM PC and has a fast disk drive. This system would not need a separate interface between the transputers and the disk drive, but the host computer would cost substantially more than the IBM PC so that this option would be somewhat more expensive. An advantage of the alternate host computer approach is that this faster host computer would be much more efficient than the IBM PC at doing related tasks, such as marking target and background regions for the training process or simply allowing human inspection of images.

Measurements on the current system indicate that the time to write an image to the hard disk is slightly longer than the image read time. For the estimates in Table A-1, we assume that the image write time is the same as the image read time for all disk and software combinations.

Calculating a single feature image takes roughly 9 seconds with the current software, more or less depending on which feature we want to extract. If we change from the current developmental software with its very general capabilities, to software optimized to extract specific preselected features efficiently, we can probably reduce the time to roughly 3 seconds per feature. The currently used set of 17 T800 transputers can be expected to take about 60 seconds to calculate a typical set of 20 feature images per scene. The processing time in approximately inversely proportional to the number of transputers; doubling the number of transputers should cut the calculation time in half.

A new type of transputer, the 9000 series, is expected to be available in 1992. The exact details of the specifications are a little vague, but it seems that the 9000 series transputer intercommunications should be about 8 times faster than the 800 series, and the calculation speed should be about 6 times faster. These numbers give the estimates in the last half of Table A-1.

Table A-2 gives estimated times <u>per training scene</u> for a typical training procedure. The number of training scenes might be any number greater than 0, possibly hundreds.

The time per scene for the training process is much greater than the time per scene in the production (analysis) process, because the training process requires writing and reading each feature image to disk. (There is no reasonable possibility of keeping all the feature images in memory.) Furthermore, the feature images must be read in a fragmented manner so that the time to read each feature image is the full 18 seconds for the IBM PC and correspondingly longer on other systems. The input/output times in Table A-2 will probably be reduced somewhat by the use of feature images with fewer pixels than the raw images in many, but not all, applications; we expect to make this improvement in the software in the near future. However, the fact remains that the relatively long time per scene in the training process, multiplied by the possibly large number of training scenes, emphasizes the need for speed in both input/output and calculation operations.

MASS STORAGE REQUIREMENTS

The need for a high-capacity mass storage device is obvious when we consider storing feature images for the training process. 1/4 megabyte per image, 20 feature images plus 3 raw images plus 1 mask image per scene, with only 10 training scenes gives a storage requirement of 60 megabytes for a relatively small training process. This indicates the need for mass storage in the gigabyte range.

COSTS

T800 transputers with 4 Mbytes of memory can be obtained for less than $1600 each, so a set of 17 would cost about $27200. There is no reliable information available on what the T9000 transputers will cost.

A minimal host system, such as a personal computer and a gigabyte disk, can be obtained for less than $6000. Thus, the total cost for a minimal system using T800 transputers would be about $33200. This system is represented by the "T800 transputers, external disk..." rows in Tables A-1 and A-2.

For an additional $5000 or more, we could get a alternate host computer that would be more versatile and allow other image handling operations more easily, such as a SUN SPARC computer or a Silicon Graphics IRIS computer. Whether this additional flexibility will ever be needed, and whether it is worth the extra cost, cannot be judged with the information currently available. This alternate host option is represented by the "T800 transputers, SUN host" rows in the tables.

HARDWARE RECOMMENDATIONS

Assuming that this hardware system will be dedicated to this one application (Automatic Target Recognition), and not used for anything else, it seems that the only justifiable choice is the minimal system. This would use a small (personal) computer as a host, a set of transputers, and a hard disk interfaced directly to the transputers (not via the host computer's bus).

Table A-1: Production Mode Timing

SYSTEM	I/O TIME	CALC. TIME	TOTAL TIME
T800 transputers, PC hard disk, currently available host interface software	40	60	100
T800 transputers, PC hard disk, new host interface software	8	60	68
T800 transputers, external disk with SCSI interface	2.5	60	62.5
T800 transputers, SUN host	2.5	60	62.5
T9000 transputers, PC hard disk, current host interface	40	10	50
T9000 transputers, PC hard disk, new host interface software	8	10	18
T9000 transputers, external disk with SCSI interface	1	10	11
T9000 transputers, SUN host	1	10	11
This assumes 3 raw images and 20 feature images per scene, and a set of 17 transputers.			

Table A-2: Times per Training Scene

SYSTEM	I/O TIME	CALC. TIME	TOTAL TIME
T800 transputers, PC hard disk, currently available host interface software	600	60	660
T800 transputers, PC hard disk, new host interface software	408	60	468
T800 transputers, external disk with SCSI interface	28	60	88
T800 transputers, SUN host	28	60	88
T9000 transputers, PC hard disk, current host interface	600	10	610
T9000 transputers, PC hard disk, new host interface software	408	10	418
T9000 transputers, external disk with SCSI interface	15	10	25
T9000 transputers, SUN host	15	10	25
This assumes 3 raw images and 20 feature images per scene, and a set of 17 transputers.			

END

DATE
FILMED
6/12/92