AN INTERACTIVE EXPERIMENTATION SYSTEM FOR THE ANALYSIS OF TIME RELATED DIGITAL IMAGES

S.L. Becker and R. Bajcsy

Computer and Information Sciences Department
University of Pennsylvania

ABSTRACT

One of the newest and most exciting areas of research in the area of pattern recognition and computer vision is that of dynamic scene analysis. Only since 1974 has significant investigation taken place in this area. It has become obvious that the number of features which can be used in the analysis of dynamic scenes is quite large and diverse and as such, no one method can be applied to a majority of image situations.

The diversity and specificity of the available features and methods have led us to the development of an interactive experimentation system for analyzing time related images. The system allows the user to test the various feature detectors and segmentation algorithms on his specific input conditions and his desired results. This provides a quick and easy method of determining the type or types of algorithms which will be effective for the given problem.

The system provides the user with the ability to interactively do the following with the input frames: display, save, add, subtract, average, threshold, filter, histogram, encode binary regions, and measure optical flow. The system is modular to provide easy implementation of additonal features or algorithms. Examples of all of the above system features are shown on images either created experimentally or gathered with a CCD camera.

RÉSUMÉ

Un des secteurs de recherche les plus nouveaux et les plus intéressants dans le domaine de la reconnaissance de diagramme et de la vision d'ordinateur est celui de l'analyse de scène dynamique. Ce n'est que depuis 1974 qu'une étude importante a été faite dans ce domaine. Il est devenu évident que le nombre de caractéristiques qui peuvent être utilisées dans l'analyse de scènes dynamiques est relativement élevé et diversifié et, comme tel, aucune méthode précise ne peut être appliquée à une majorité de situations d'images.

La diversité et la spécificité des caractéristiques et des méthodes accessibles ont entraîné le développement d'un système d'expérimentation interactif pour analyser les images qui varient dans le temps. Le système permet à l'utilisateur de vérifier les divers détecteurs et les algorithmes de segmentation selon les conditions d'entrée spécifiques et les résultats désirés. Cela constitue une méthode facile et rapide de déterminer le ou les types d'algorithme qui conviendront à un problème donné.

Le système permet à l'usager de réaliser interactivement les opérations suivantes avec les séquences d'entrée: affichage, conservation, addition, soustraction, établissement de moyenne, établissement de seuil, filtrage, histogramme, encodage des régions binaires et mesure du flux optique. Le système est modulaire et met facilement en application les caractéristiques ou algorithmes additionnels. Des exemples de toutes les caractéristiques de système mentionnées ci-dessus apparaissent sur les images créées expérimentalement ou regroupées avec une caméra CCD.

I. Introduction

The basic motivation for developing a system for analyzing time related digital images comes from a study of scene analysis techniquies which use motion as a cue for segmentation or use motion information as input to knowledge about a scene. There are many problems associated with the broad area of dynamic scene analysis, or even just motion detection. These problems can be arranged hierarchically to reflect increasing specificity and definition of the problem. At the highest level is the question of whether the motion or change in the scene simply need be detected or whether this change needs to be quantitatively described. We must also consider the characteristics of the image space. The problem is quite different if the camera is moving than if it is static. Consideration must also be made for the type of motion objects will be allowed to undergo and the number of moving objects in the scene. Different techniques must be used to differentiate between translation, rotation, scale changes and occlusion.

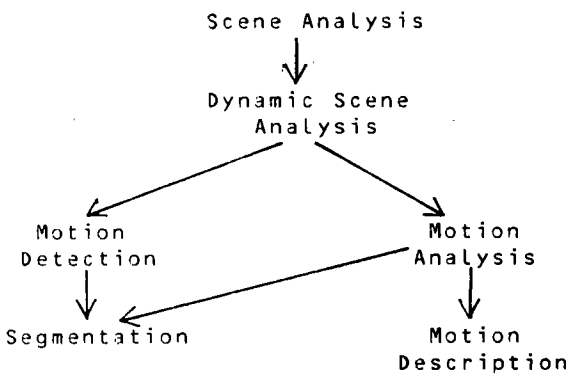

figure 1.

The problem of describing the motion within a scene was examined by Badler in 1974, but his work was only on simulated images. Significant research has been done analyzing the motion of artificial 2-D polygonal regions and the motion of clouds [Martin and Aggarwal 1978]. Nagel and his associates [Nagel 1977; Jain, Militzer, and Nagel 1977; Jain and Nagel 1979] have done a substantial amount of work with real video images detecting motion with a stationary sensor. A totally different approach has been taken by researchers who have attempted to measure the velocities of brightness points within the image as a means of understanding the motion of the underlying objects [Fennema and Thompson 1979; Horn and Schunch, 1980]. Horn refers to this measurement as Optical Flow. Horn, however, presents results only from simulated images.

There are many possible applications, including our own, of a motion detection system which cannot fit the constraints of any one of the above algorithms or methods. It is therefore necessary to be able to specify a new set of constraints and algorithms wich will work for the given application. It is with this in mind that we have designed this interactive system. In order to save large amounts of development time, it is necessary that the user have available to him a wide range of operations and feature detectors with which to test his ideas on his specific data.

II. Overview

The system consists of 3 basic units: the command processor, the image database, and the operations modules. The command processor accepts commands which direct various combinations of the operations modules to operate on the image database. The system has been designed with the inexperienced user and with flexibility of modification

in mind.

The first and foremost motivation for the system is to allow a user with a particular set of data to implement proven algorithms or to design his own algorithms using the system as an evaluation tool. To this end the system is purely an interactive system. All commands to the system must be entered as they are needed and the results are immediately and conveniently displayed. A menu of commands is provided with a help function to explain the use of each system command. Various display formats are available to allow the user to view his data in the most convenient manner.

The second requirement of the interactive system is that it be flexible enough that a user with specific needs can modify or add to the system without changing the entire system. The modular package of operations can be added to or modified without alteration to any of the other modules or to the database. Once a new operation is defined the programmer needs only to add the apropriate prompt line and help function description of that module to the command processor. This allows quick and easy addition to the system without the need for modification or re-compilation of large amounts of code.

Figure 2 illustrates the modular design of the system. Each of the units can be modified in most cases without any changes to the remaining functional units. Figure 3 illustrates the basic concept of the interactive system. By eliminating the link between the user and the system operations and forcing all transactions to pass through the interactive interface, the programmer is forced to remain within the design constraints of the system. Operations interfaces and input protocols are thus standardized.

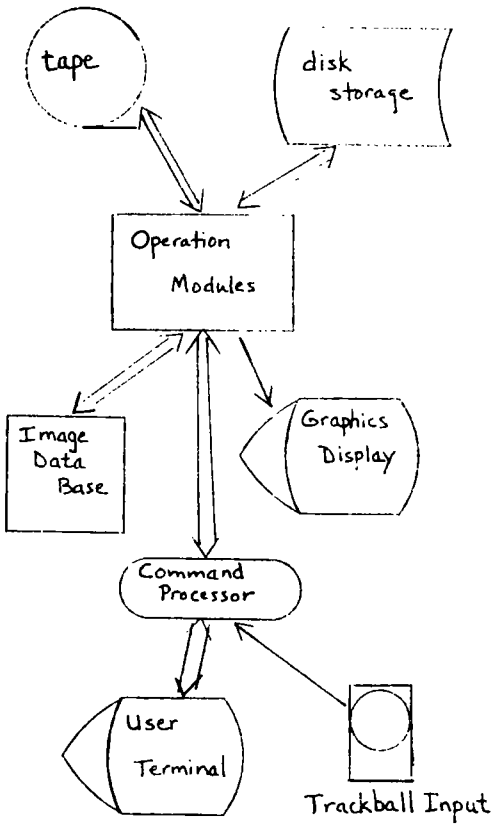

figure 2. System Functional Blocks

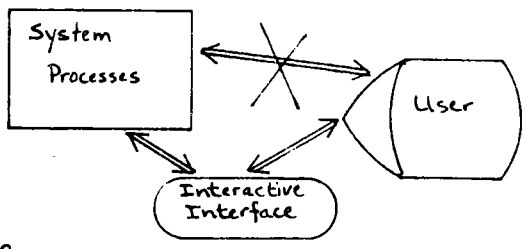

figure 3. Design of an Interactive System

III. Image Database

The image database used in the system is a buffer of 10 frames of image data active at one time with an information header for each frame. The current implementation of the system uses a frame size of 100 by 100 brightness cells. The actual size of the images used is only limited by available memory and graphics display resolution. All operations modules are designed to operate on images of externally specified dimensions.

Associated with each frame of data is a header which contains the current screen location (if any), statistics about the image, and a 20 character title or label for that frame. The statistics computed are minimum, maximum, mean, and standard deviation of the image brightness. The frame headers are displayed on the terminal when the system is waiting for a command or when the user requests that they be displayed.

IV. Operations Modules

The following is a short description of each of the operations modules currently in the system. A more in depth look at the implementation of some well known segmentation algorithms for time related images using the system is presented later.

File Operations:

READ - image frames will normally be stored on disk. This operation will read the next frame record from the current disk file to the specified destination buffer.

WRITE - frames may be sequentially output to the current output disk file from the specified source buffer.

INFILE - opens a new input file for the READ operation.

OUTFILE - opens a new output file for the WRITE operation.

OUTPIC - will output a frame in PIC file format compatible with other processing programs.

Single Frame Operations:

SCALE - maps the existing range of grey values into a new range of values. Histogram expansion and compression can be accomplished with this operation.

THRES - a binary image is created by thresholding the image at a given value.

THIN - once a binary image is created, thresholding based on connectivity can be done with this operation.

SOBEL - the Sobel gradient operator is applied to the image point by point.

GRAD - a gradient operation slightly different than the Sobel operator is applied to the image. Given that the point being operated on is X and the neighbors are defined as:

```
.   A   .
B   X   C
.   D   .
```

The operator is defined as

$$g = (|B-X| - |X-C|)/4 + (|A-X| - |X-B|)/4$$

ZOOM - a small portion of the image will be expanded to the size of a full frame.

BLUR - the image is blurred by replacing each pixel value with the

average value of its 8 nearest neighbors.

Multi-frame Operations:

ADD - two images are added point by point.

SUB - normal subtraction of two images is done point by point.

DIFF - absolute subtraction is performed on the two input images.

MASK - one image is used to mask the other frame. Masked points are set to zero and all others remain unchanged.

AVE - any number of frames are averaged on a point by point basis.

TGRAD - an estimate of the gradient in time as defined by Horn [1980].

$$E_t = \{E(i,j,k+1) - E(i,j,k) + E(i+1,j,k+1) - E(i+1,j,k) + E(i+1,j+1,k+1) - E(i+1,j+1,k)\}$$

Display and Output Operations:

SHOW - display the frame on the RAMTEK screen at the specified screen location.

LIST - create a line printer output of the values of the specified frame.

HIST - compute and display a histogram for the image. Any range of values and number of buckets may be histogrammed.

LABEL - place the text description label from the frame header on the graphics screen.

Special Operations:

The following operations are more complex and specific. They are examples of the type of operations which can be added by the user.

FLOW - The Optical Flow, as defined by Horn is computed. The output of this operation is a special display. See Horn and Schunch, 1980 , for a description of this operation.

CREATE - simulated images can be created with this operation. As defined now it will fill a frame with a sinusoidal brightness pattern. Successive calls to this operation will create frames in which this pattern moves in a specified direction and magnitude.

GROW - a region growing algorithm is applied to a binary image. Each region is chain encoded and then labelled for output. The user is prompted for the filename to output the chain encoding of the regions if wanted.

V. Results

Problems associated with photographic reproduction have prohibited us from including in this paper any visual examples of the system in use. These will be presented, however. We have been using the system to test several segmentation algorithms using data gathered with a 100 x 100 pixel CCD camera. Real scene data using objects in the laboratory moving about the room have been input to the system. The interactive system has proved invaluable in the efficient evaluation of algorithms as they have either been designed or brought to our attention. Many of these methods have been ruled out quickly by doing "quick looks" on the system.

VII. References

Fennema, C.I. and Thompson, W.B., Velocity Determination in Scenes Containing Several Moving Objects, CGIP 9, April 1979, pp. 301-315.

Horn, B.K.P. and Schunch, B.G.: Determining Optical Flow, A.I. Memo No. 572, MIT, April, 1980.

Jain, R. Militzer, D. and Nagel, H.H.: Separating Non-stationary for Stationary Components in a Sequence of Real World T.V. Images, Proceedings of the Fifth IJCAI, 22-25 August 1977, pp. 612-618.

Jain, R. and Nagel, H.H.: On a Motion Analysis Process for Image Sequences and Real World Scenes, IEEE Transactions on Pattern Analysis and Machine Intelligence 2, April 1979, pp. 206-214.

Jain, R.: Peripheral Processes in Machine Perception of Motion, Proceedings of the IPR Conference, Miami, 1980, pp.1243-1245.

Martin, W.N. and Aggarwal, J.K.: Survey - Dynamic Scene Analysis, CGIP 7, 1970, pp. 356-376.

Nagel, H.H.: Analyzing Sequences of T.V. Frames, Proceedings of the Fifth IJCAI, 22-25 August, 1977, p.626.

VI. Acknowledgement

This work has been partially supported by

RCA Corporation, ATL,
Image Processing Division in
Camden, NJ.