Window Based Interactive Image Processing

Ericsson, Toni; Nielsen, Lars

1989

Document Version:
Publisher's PDF, also known as Version of record

CODEN: LUTFD2/(TFRT-7440)/1–5/(1989)

Window Based Interactive Image Processing

Toni Ericsson
Lars Nielsen

Department of Automatic Control
Lund Institute of Technology
December 1989

Department of Automatic Control
Lund Institute of Technology
P.O. Box 118
S-221 00 Lund Sweden

Document name
Report

Date of issue
December 1989

Document Number
CODEN: LUTFD2/(TFRT-7440)/1-5/(1989)

Author(s)
Toni Ericsson and Lars Nielsen

Supervisor

Sponsoring organisation
The Swedish Board for Technological Development

Title and subtitle
Window based interactive image processing

Abstract

Paper at the Swedish IAPR (SSAB) Conference, Göteborg, March 15–16, 1989.

Key words

Classification system and/or index terms (if any)

Supplementary bibliographical information

ISSN and key title

ISBN

Language
English

Number of pages
5

Recipient's notes

Security classification

Window based interactive image processing

Toni Ericsson, Lars Nielsen
Department of Automatic Control
Lund Institute of Technology
Box 118, S-221 00 Lund, Sweden

1. Introduction

A current trend in image processing for personal computers is to provide plug-in boards for image handling together with image software libraries for image processing. Another line of development is window based interactive software systems (Newman and Sproull, 1978). Such software can be used to create very user friendly interfaces e.g. to image processing libraries, and hence to image processing hardware.

Such a system has been implemented by combining "off the shelf" commercial products. More specifically, Microsoft Windows has been used with a Matrox imaging card and an IBM PC/AT. The work has emphasized on general solutions in order to provide maximum flexibility and hardware independence. Considerations on what an image processing environment should contain have also been addressed.

2. Design

User interface

From the users point of view, the operations that one can invoke should be easily accessible and self explanatory. Similarly, the output from an operation should be presented in a way that makes it easy to extract the relevant information. A graphical window interface satisfies these requirements. Separate windows can be used to group operations that are similar.

The use of menus, e.g. pulldown or popup types, facilitates for the user to get an overview of the features available. Buttons are beneficial when it comes to setting or displaying the state of the hardware.

Most of the operations are invoked through the pointing device. However, the user should also be able to use the keyboard when this is more appropriate. Since there can be several windows visible that can accept keyboard input, the window which currently receives the input should be highlighted in some way.

Internal structure

When designing software for existing hardware a major design consideration is modularity. It should be possible to port the program to other hardware without having to rewrite major parts of the code. Different function groups

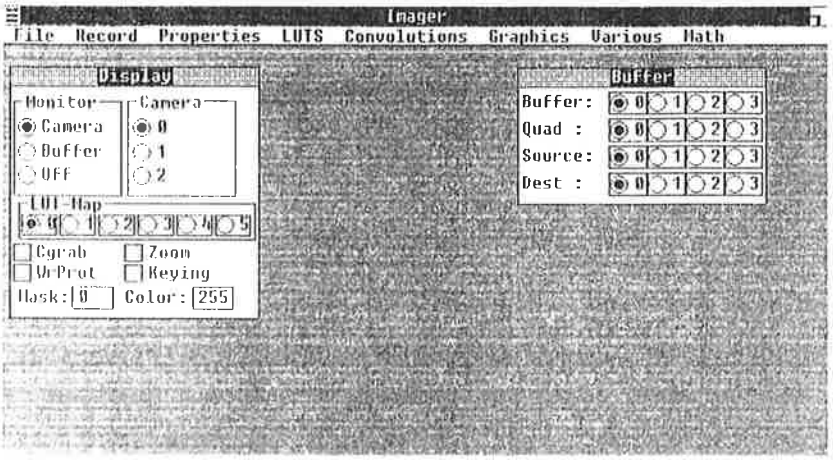

Figure 1. The opening display.

should be independent of each other and only e.g. interface to the menu handling functions that invoke them and hardware specific functions.

It should also be possible to easily insert new functions and display windows, or to remove such. The new functions inserted should also be able to communicate with other parts of the system without having to modify these parts.

Implementation

A program which based on the guidelines given above has been fully implemented. It contains function groups which are relevant to image processing such as storing and retrieving images from magnetic media, image acquisition, statistical analysis, edge detection etc. An abstract model of the hardware has been made so that the user easily can manipulate images in a structured way. Separate windows lets the user set the state of the hardware as well as graphically view measured data. Windows can overlap on the screen, and the user can remove them.

Each window is handled by a separate function which makes it easy to modify its appearance. Buttons, x-y diagrams etc, have been implemented as objects. Communication between different objects are handled via messages.

At the bottom layer there are driver functions which directly access the hardware. Since input is handled by the window interface only these functions have to be altered in order to port the program to other processing hardware.

3. System description

The main part of the system is the IBM PC/AT with an EGA graphics display and equipped with a mouse as pointing device. For image acquisition the Matrox PIP 1024B image processing board was chosen, and the software operating environment is the Microsoft Windows windowing system.

The IBM PC/AT

The host computer, the IBM PC/AT, is essentially a 16 bit, single user system. It's CPU is the Intel 80286 and virtually all hardware resides on a single board. It provides hardware interface Through several expansion slots connected to the main bus. Our system was equipped with 512 KB of RAM, a 20 MB hard

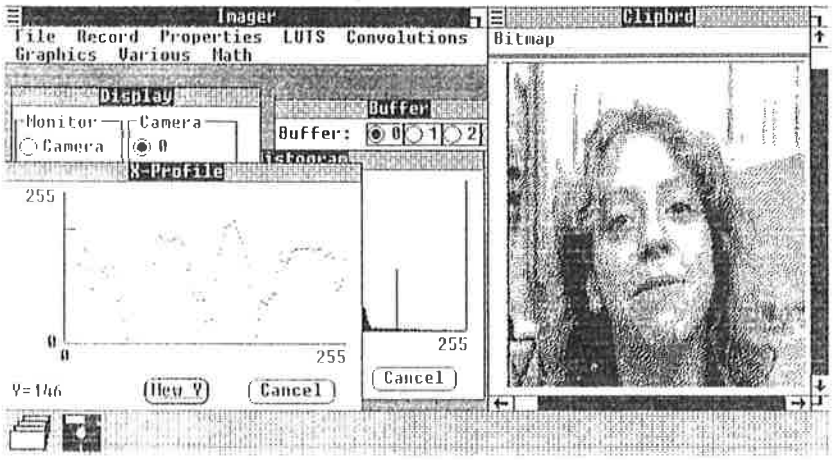

Figure 2. Working with several windows.

disk and IBM Graphics dot matrix printer. The system ideas can be viewed as an extension of earlier systems in the same flavor (Nielsen, 1985, 1988).

The Matrox PIP 1024B

The PIP is a plug-in video frame grabber board, available for different types of computers. It digitizes a video signal coming from e.g. a video camera or video recorder in real time at a resolution of 256 by 256 or 512 by 512 pixels, and with 8 bits per pixel. The board itself can be viewed as consisting of 3 sections, namely the input section, the grabbing section and the output section.

In principle the input section digitizes the incoming video signal, which is stored by the grabbing section and displayed on a separate monitor by the output section.

The raster memory is due to limitations imposed by the host computer not memory mapped, instead it is accessible Through a set of I/O ports. These ports can be moved in the I/O address space to prevent collisions with other hardware.

The Microsoft Windows operating environment

Microsoft Windows, or just Windows, resides on top of MS-DOS, the operating system for the IBM PC/AT, and provides the user with three main capabilities. They are a graphics oriented user interface, a multitasking capability, and hardware independence. When running Windows the user has a uniform environment for all applications he or she is running since the programs all have a very similar layout, practically regardless of what type of program it is. This environment is very similar to the one that became so popular with the introduction of the Apple MacIntosh.

4. Experiences

Developing an interactive image processing system on a personal computer has its advantages and disadvantages. Due to memory limitations it is often necessary to compromise. The result might be lower processing speed or limited functionality. As personal computers get bigger and faster, this problem is likely to disappear soon.

The main advantage is that it is possible to obtain an advanced system at extremely low cost compared to other systems of similar performance. Through the window based user interface it has been possible for users with little prior knowledge of image processing to utilize the hardware and algorithms quickly.

Presently a two monitor solution has proven to be the most flexible. As the hardware gets more sophisticated it will be possible to handle everything on one monitor. This will not change the approach taken in this implementation.

5. Conclusions

The combination of window based software with image plug-in board makes it possible to design a user friendly environment for image processing. Commands can be grouped graphically for the interaction in a natural way. There are several flexible ways to modify or extend the system, if the internal layout is properly structured.

A successful implementation has been made of an image processing system based on commercial hardware. The amount of work has been reasonable, and the basis laid can be extended as new hardware emerges. The result has been quite satisfactory and has resulted in a commercial product that has already been sold in several copies.

6. References

NIELSEN, L. (1985): "Simplifications in visual servoing," Ph.D. thesis CODEN: LUTFD2/TFRT-1027, Department of Automatic Control, Lund Institute of Technology, Lund, Sweden.

NIELSEN, L. (1988): "Automated Guidance of Vehicles using Vision and Projective Invariant Marking," *Automatica*, **24**, No.2, 135-148.

NEWMAN W. M, SPROULL R. F (1978): *Principles of Interactive Computer Graphics*, McGraw-Hill.