J. Chem. Inf. Comput. Sci. 1987, 27, 86–95

SIMIPS: Secondary Ion Mass Image Processing System

YONG-CHIEN LING, MARK T. BERNIUS, and GEORGE H. MORRISON*

Baker Laboratory of Chemistry, Cornell University, Ithaca, New York 14853

Received September 15, 1986

A secondary ion mass image processing system (SIMIPS) is presented as a quantitative image analysis tool, with emphasis on an efficient man–machine interface. The combined applications of digital image processing and pattern recognition ensure an intelligent problem-resolving scheme and optimal extraction of information. The system performance is evaluated, and typical applications are presented to illustrate the versatility and usefulness of SIMIPS in analyzing digital images.

INTRODUCTION

Secondary ion mass spectrometry (SIMS), with its high elemental sensitivity and high spatial resolution imaging capability, is a unique and powerful analytical technique for the microcharacterization of solid-state devices, geological samples, and biological tissues.[1] The secondary ions are registered in an image detector, maintaining a one-to-one spatial relationship with the sample surface to form a SIMS micrograph. Each pixel (picture element) in a given image holds the ion intensity measured at a given mass-to-charge ratio (chemical identity) and corresponds to a particular location in an object field (spatial information). This is collected with a unique setting of energy filtration (chemical energy information) at a certain time (depth information). Ideally, two-dimensional compositional and morphological information should be obtainable from a single image. In addition, a three-dimensional reconstruction of the sample's mass distribution, which is obtainable from a series of micrographs collected at various depths of the sample, is also possible.[2–4]

However, the practice of this image information extraction scheme is tedious and difficult (if not impossible) at the current stage of development. This is due not only to the vast amount of multidimensional information contained in the images but also to the complexity caused by the various degrees of image degradation at the many transducing stages in the image-generating process. It is obvious that only with the help of computers and digital image processing (DIP) techniques can the information created by this multidimensional technique be fully realized. The increasing importance of DIP as an invaluable tool for the analysis of SIMS micrographs is evidenced by recent publications that cover a wide range of applications.[5–7]

This study is part of a continuing effort in this laboratory to explore the extraction of quantitative geometric and densitometric information from stigmatic SIMS images by DIP. In this paper, SIMIPS (secondary ion mass image processing system), an integrated, modular-designed software package with a convenient and efficient man–machine interface to facilitate the implementation of human intelligence into the overall information distillation process, is described. Features such as user-friendliness, reliability, flexibility, power, and speed are incorporated.

HARDWARE CONFIGURATION

The block diagram of the hardware system is presented in Figure 1. All images were collected from a CAMECA IMS-3f stigmatic ion microscope, which is described in detail elsewhere.[8] The microscope is interfaced to a Hewlett-Packard 9845B microcomputer for instrumental control and data acquisition. Use is made of the second generation MIDAS (microscopic image digital acquisition system),[9] which is built around a PDP-11/34A (Digital Equipment Corp.) 16-bit minicomputer (with 256 kbyte of core memory) and two removable RL02 hard disks (DEC, each with 10 Mbyte of storage capacity) for primary mass storage. Each is capable of storing up to 83 images of dimension of 256 × 240 pixels, 16 bits per pixel. Usually one disk is used as the system disk to control program execution while the other is used as the data disk to store the images. In this way, the backup of image data can be hastened by copying the contents of the entire disk (instead of copying by the files) on to a 1600-bpi (bits per inch) magnetic tape (Kennedy Model 9000 digital type drive). From our experience, it is important to have enough mass storage memory and an efficient backup scheme due to the large amount of data involved. Furthermore, these hard disks are indispensable for SIMIPS because it is a disk-resident program. The RL02 disks were selected because of their reliability, fast I/O speed, and inherent temporary backup capability (simply swapping in a new disk).

In the direct on-line digital image acquisition process, there are three transducing stages that have an immense impact on the final quantification results. In order of implementation they are (1) the microchannel plate of the microscope, (2) the intensified silicon intensified target (ISIT) video camera, and (3) the video digitizer. In order to adapt to a wide range of incoming signals to increase the dynamic range, there is a gain control at each stage. In addition, there is also an offset control at the video digitizer. Each of these instrumental parameters has an effect on the accuracy of the calibration curve. They are read into the PDP-11/34A through an LPS-11 interface.

From the DIP point of view, the heart of the SIMIPS is the TRAPIX image processor (Recognition Concepts Inc. Model 55/32), which is capable of grabbing, storing, and processing images at video-frame rate (33 ms per frame, so-called real time). The TRAPIX is interfaced to the PDP-11/34A at the bus level to facilitate traffic between them. Four control registers of the TRAPIX are mapped as part of the PDP's memory and are directly accessible to the PDP. The TRAPIX is capable of storing up to four 512 × 512 pixel images, 8 bits per pixel, and one 80 × 25 alphanumeric overlay. Two of those 8 bits per pixel images can be concatenated to a 16 bits per pixel image under software control. This is a very useful feature when the images are acquired integratively to improve the signal-to-noise ratio (S/N). With the 8-bit video digitizer, a total of 256 frames can be summed together into this 16-bit image. In practice, the maximum number of frames to be summed without saturation is determined by the pixel intensity in the acquired image.

Once the image is in the TRAPIX it can be processed in a variety of ways, all in real time and controlled by the pipeline image processor (PIP) which is the heart of the TRAPIX, as Figure 2 indicates. Inputs to the PIP are selected from two of the following: (1) the video digitizer, (2) a constant, and

0095-2338/87/1627-0086$01.50/0

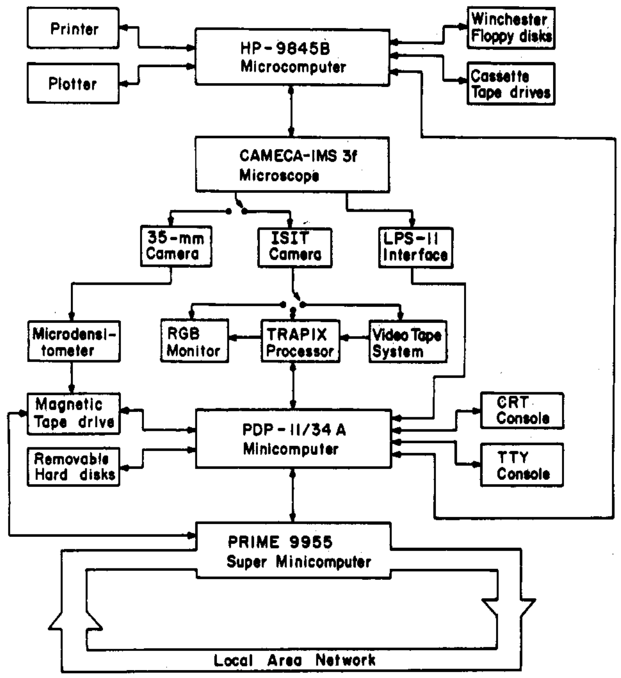


Figure 1. Hardware system configuration of SIMIPS.

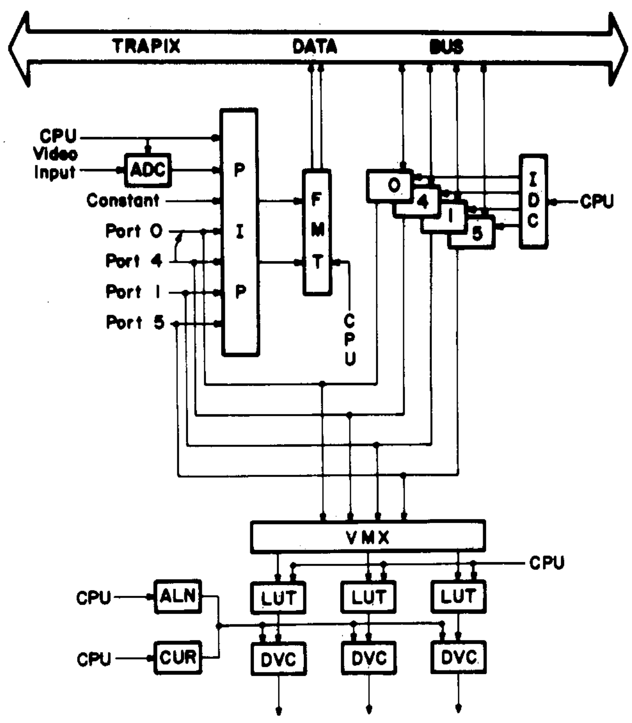


Figure 2. Block diagram of the TRAPIX 55/32 system. Blocks connected to the CPU are under software control by the PDP-11/34A.

(3) the 8-bit image data ports. The PIP, which is controlled by the PDP-11/34A, carries out the arithmetic and logical operations upon the incoming images to facilitate a plethora of real-time image-processing jobs. The resulting output image is then fed into the formatter (FMT) to allow three different samplings to reduce its size to 256 × 256, 128 × 128, or 64 × 64 pixels. Thus the fixed image memory (512 × 512 pixels) can display a large series of images (4, 16, or 64). The images can be selected, scrolled, roamed, and zoomed under the command of the image display controller (IDC). These massaged images are then sent to a video mixer (VMX), mapped through the look-up tables (LUTs), and fed directly into the digital-to-video converter (DVC) to be displayed on a high-resolution RGB color monitor capable of displaying 512 × 480 pixels. The displayed images can be superimposed with alphanumerics to append explanatory information or superimposed with cursors with an accuracy of one pixel and manually controlled by a joystick to allow interactive interrogation of pixel position and intensity. With these capabilities, a subregion of the stored image can be magnified in real time to fill the screen, yielding detailed spatial information. The LUTs selectively redefine each pixel value of the input image to a corresponding output value at video-frame rate. Thus when the content of 256 bytes of the three LUTs (compared to 512 × 480 pixels) is modified, the visual perception of the entire image can be revised in real time. To anticipate future computational needs, a 9600 baud rate RS-232 digital link is built to connect the PDP-11/34A to the department 9955 PRIME computer which is equipped with 14 Mbyte of core memory and a MAP-6420 array processor (CSPI Inc.). With this communication link, the PDP-11/34A becomes a member of the local area network and has access to an enormous amount of computational resources.

SOFTWARE OVERVIEW

1. Design Considerations. SIMIPS was developed on the basis of experience and knowledge acquired through the practice of ion microscopic DIP. The following properties were incorporated into the final product and were used as guidelines in designing the program:

a. Interactive Processing and Flexibility of Options. Each implemented command is identified in advance and is restricted in size. The entire system is configured modularly, and more elaborate schemes can be built from these elementary operations by linking them together according to the operations required. By this approach, flexibility, reliability, and power are guaranteed.

b. Batch Processing. The established optimal scheme, which is an assembly of individual commands, can be stored permanently as a procedural command file and can be shared easily with other users to process larger data sets. Intrinsic programming facilities similar to those of high-level languages, such as parameter setting, arithmetic expression evaluation, and conditional branching, are provided for the flexible assembly of commands for more complex jobs. Combining these built-in programming facilities and the large collection of supported functions, SIMIPS functions as a powerful high-level DIP language.

c. Ease of Operation. A novice is rapidly acquainted with the most commonly used features of SIMIPS by invoking the demonstration command. The user enters either a three-letter mnemonic command or a numeric one to initiate the specific operation with the option to request the details about the invoked command or to undo it before the execution. A procedural command file may then be called in to perform complicated DIP works or to manipulate a large number of data sets. The tutorial menu can be activated for command syntax or to request on-line help for details about the command action. All the working parameters are entered diagnostically, in numeric form, and in a single line. Additional lines, in alphanumeric form, are required if disk *I*/*O* operations are involved. The most common default values are assigned to these working parameters beforehand. The user has the option to modify the parameters or not, thus achieving efficient and convenient man–machine interface. Any input error causes an immediate error message, allowing the user to resume after correcting the mistake(s). Direct interrogation of each pixel is possible through the joystick or via the terminal. A transparent switching mechanism between the RT-11 operating system and SIMIPS is provided without aborting current DIP task. Within this framework, SIMIPS appears more like a powerful DIP operating system.

d. Rapid Response. One key prerequisite for interactive processing is indeed fast computation. This is accomplished partially with a dedicated computer (such as the PDP-11/34A) and efficient programming (see System Performance Evaluation section). Even so, the processing of a 512 × 480 pixel image is still a slow process. This overload problem is alleviated effectively by employing the PIP for operations such as vector drawing, template filling, and simple arithmetic and logical operations. Additional speed is achieved by modifying the content of LUTs for image enhancement. In summary, the most efficient way to relieve the computational burden of the PDP-11/34A is to allocate its computational tasks, as much as possible, to the TRAPIX and execute them in real time.

e. Integrated Design. SIMIPS is capable of manipulating image, line, and scalar types of data. All of these data are treated as operands and can be saved permanently as disk files. In the course of processing, they are placed in the memories of PDP-11/34A and TRAPIX to lessen disk *I/O* overhead. The output of one command may become the input to the next command if desired. This is the so-called piping technique because the user may think of the data as traveling down a pipeline from the first to the second command. With this instrumentality, the flow of data between commands is smooth and flexible. Thus the user can concentrate more on data analysis instead of dealing with data compatibility and communication. The controlling kernels of SIMIPS are two register arrays of 32 elements each, one of real-number form (32 bits) and the other of integer-number form (16 bits). The former is employed for storing intermediate scalar results to ease switching between application sessions, while the latter is used to reveal the current operating environment for system control. All *I/O* and display-related parameters are initialized in the main program and passed through common areas to speed communication between commands and ease future program maintenance work.

f. Program Development and Maintenance. Step-by-step instructions are provided to facilitate the appending of new commands to the current system and the related re-forming jobs. New commands can be developed easily by adding features into a protocol subroutine with standard statements for input, output, and error handling to assure integrity and efficiency. The practice of detailed documentation is enforced by building the new command from the well-documented protocol subroutine. System command files are built to perform jobs such as compiling and linking of programs. With the above implementations, program development is standarized and maintenance is automated. This is vital to ensure the long life cycle of SIMIPS, especially for a research-oriented laboratory where high turnover of research personnel is inevitable.

2. System Descriptions. SIMIPS is coded in FORTRAN IV (except the TRAPIX-dependent *I/O* subroutines that are supplied by the vender and coded in MACRO-II to gain speed) and runs under the RT-11 operating system. It is a collection of 120 subroutines with a 1-Mbyte executing program. An overlay program structure, organized into four program levels, is designed to fit into the 64 Kbyte of programmable core memory. The highest level consists of a supervising program which translates the input command into internal code and then dispatches it to the appropriate executive program in the next level. The specified processing module (in the third level) is then loaded into the core memory and executed. Additional library subroutines (in the fourth level), such as TRAPIX *I/O* and mathematical operations, can be loaded into the memory if required. Upon completion of the current task, the control is returned to the next higher level calling program to resume execution. With this control structure, the execution speed

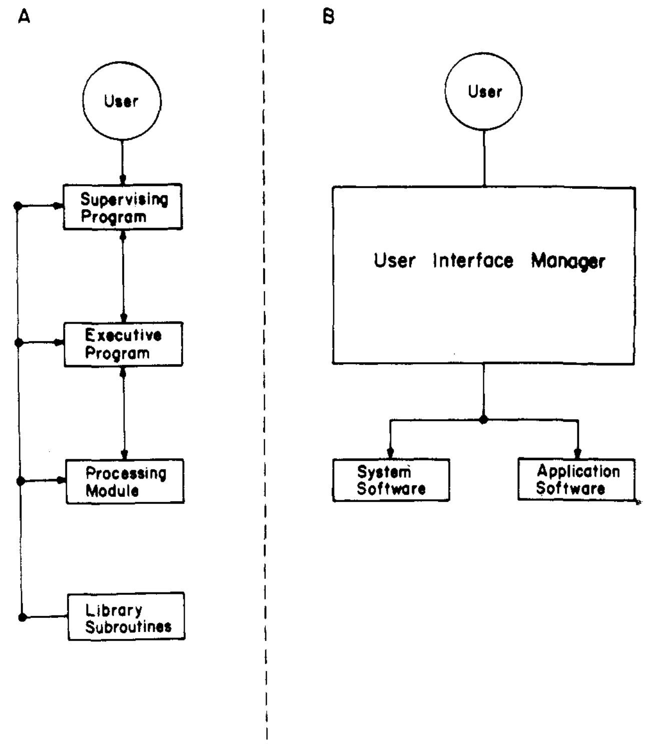


Figure 3. Software architecture of SIMIPS: (A) four-level program structure; (B) user's perception as a bilevel tree structure. User interface manager consists of a supervising program and an executive program. System and application software are invoked from the processing modules.

is optimized by eliminating program returns to the RT-11 operating system, and the core memory requirement is also minimized by reading in the processing module from the disk as it is needed. To the user this four-level program structure is transparent and is perceived as a bilevel tree structure, as Figure 3 illustrates. The user communicates with SIMIPS by entering a command, through the user interface manager, which in turn deciphers the command and transmits the control to the related processing module. The module is described by two broad categories: system software and application software.

SYSTEM SOFTWARE

The purpose of the system software is to assure efficiency by supporting a variety of repetitive services, in addition to command and control purposes, required by the application software. SIMIPS system software conducts three major functions, described as follows.

1. System Control. This is the core of SIMIPS and is responsible for the following:

a. Changing the Default System Configuration. The entire 512 × 480 display memory can be partitioned into 4, 16, or 64 subimages for simultaneous viewing and rapid processing. The tutorial menu can be turned on for command explanation or to select the desired command type (mnemonic or numeric).

b. Changing Communication *I/O* between Man and Machine. Besides supporting the conventional "interactive" and "batch" processing, this also allows a procedural command file to be created with or without execution. The outputs can be displayed in the CRT-terminal or stored into a disk file or both.

c. Scheduling the Various Processing Tasks. This is done by tracking the system time and controlling the scheduling functions.

d. Viewing, Retrieving, and Altering the Contents of the System Registers. With this service, the user can fully exploit the built-in programming facilities.

e. Supporting On-Line Help and Easing Program Maintenance and Development.

2. Data Management. This provides the mechanisms for cataloging, retrieval, and storage of image files. Additional file-service utilities are used to search, delete, rename, and copy files. Image-acquisition-related information is retained automatically in the file header section, and an additional note section is allocated for saving individual-related information provided by the user (if desired).

3. RT-11 Operating System. While the operating environment is under the control of RT-11, a vast amount of work can be completed with previously established programs such as RAP and TURBO-RAP for image spatial resolution studies[10,11] and a videotape system for continuous video data acquisition.[12] The system also has the ability to create and update the procedural command files, to emulate the terminal, and to transfer and receive files between the PDP-11/34A and a central computing facility for heavy number-crunching tasks.

APPLICATION SOFTWARE

This software system provides the means to acquire, amend, and interpret digital images. They are, according to their functions, classified into 10 categories.

1. Image Acquisition. Direct digital image acquisition from the IMS-3f ion microscope or from a negative print of an ion micrograph is supported. An image can be integrated with a certain number of frames, to a specific period of time, or until saturation is detected. Up to 256 single-frame images can be grabbed at video-frame rate with some loss of spatial resolution (each image is 64 × 64 pixels, 8 bits per pixel). In the context of studying short-life images, this continuous real-time digital image grabbing capability is an advantage in that it serves as a preliminary tool and sidesteps the necessity of relying on the videotape system first. Acquisition-related information is stored temporarily on system registers and saved into the header portion of a disk file to facilitate subsequent correction operations. It is worth mentioning that one additional advantage of on-line digital image acquisition is that the computer, besides grabbing images, can also assist in the tuning of the instrument to acquire images with minimal distortion. This is because the computer is capable of analyzing and expressing the incoming images in various formats such as a single-line profile (and its first derivative), to which human perception is more sensitive, or as a digitally magnified image to enlarge small features.

2. Densitometric Correction. In order to reveal quantitative compositional information about the analyzed sample, densitometric correction is crucial. It includes rectification to compensate for the detector nonuniformity response,[13] differential sputtering,[14] and topographic artifacts.[15] In addition, the correction involves calibration to convert the pixel values back into numbers with physical meaning. A calibration curve is utilized, taking into consideration the various gain and offset settings of the A/D digitizers. Finally, normalization is required to transform the pixel size into physical length under the specific instrumental condition. A uniformity index is provided to determine the need for correction. Standard procedures are also provided to facilitate the generation of a calibration curve.

3. Geometric Correction. This correction is vital for proper image registration and image correlation, which is used to identify features on different micrographs (such as SIMS, light microscope, and scanning electron microscope) for the purpose of verifying that these features contain related information. The SIMS micrographs could even be collected under different magnifications (zoom correction is necessary), from different regions on the specimen (transformation correction is necessary), or in dark-field mode (feature shift correction is necessary).[15] Images collected from different instruments and by using different detection systems usually require all of the above corrections. A registration index can be calculated to objectively monitor the accuracy of geometric correction.

4. Image Enhancement.[16,17] This process is very useful is converting the images to a form (without regard to the reproducibility of their fidelity) that is better suited to human or machine analysis by improving their visual appearance. Enhancement of an image can be performed in a variety of ways. For example, the contrast of an image with a low densitometric range can be improved by rescaling the amplitude of each pixel to exploit the full densitometric range. Also a function that describes the frequency of occurrence of each gray level in the image, i.e., a histogram, can be modified to a desired form to ensure proper information distribution. An indispensable tool to diminish the spurious effects in digital images with low S/N is "noise cleaning". The noise-suppressed image is created by replacing each pixel with a new value which is the result of a convolution of window pixels with a kernel. Optional window size is adaptable to the various experimental requirements. The image may also be enhanced by a technique known as "edge crispening" which is used to detect the presence of a local edge, a small area in the image with rapidly changing gray levels. These edges outline the boundaries of features, and the clear perception of them is an a priori necessity for successful ongoing information extraction. The edge-crispened image is formed, with mathematical operations similar to those of the noise-cleaning processes, by convoluting the original image with an edge operator, defined by a mask of fixed window size. Finally, the potential use of "pseudo-color enhancement" cannot be overlooked. A color may be assigned to each pixel on the basis of its intensity, exploiting the improved sensitivity of the huamn visual system to color differences as opposed to dim gray level differences. Due to the various kinds of images involved and different types of information desired, the above five techniques are diversely implemented. In addition to global enhancement (upon the entire image), local enhancement (upon the region of interest) is also supported. Figure 4 exemplifies the enhancement effects upon a $^{40}Ca^+$ image of a plastic-embedded biological section of goblet cells in human colon, which is used as a subject for DIP examples throughout this paper.[18]

5. Mathematical/Logical Operations. One key element for efficient interactive DIP is that the user has the capability to view several images distinctively and to compare them simultaneously. This is done by reconfiguring the unit image size into smaller ones of 256 × 240, 128 × 120, or 64 × 60 pixels, thus permitting accommodation of 4, 16, or 64 images simultaneously on a monitor of 512 × 480 displaying elements. Utility commands to cut, move, and copy these subimages are provided in addition to the mathematical operations, such as add, subtract, multiply, and divide, and logical operations. With these basic commands and the capability of linking them together, the user can devise a number of problem-tackling schemes for quick overview.

6. Frequency Domain Processing. The digital image (DI) results from the summation of the convoluted image, the convolution of the original image (OI) with the instrument transfer function (H), and the system noise (N). In the spatial domain, both information and noise are superimposed pixel by pixel, as eq 1 indicates, where the star is used to indicate

$$DI = H \star OI + N \qquad (1)$$

the convolution of two functions. The extraction of the information of interest, OI, can be facilitated either by suppressing the noise contribution through noise cleaning or by enhancing the original image through edge crispening based on an ad hoc processing in the spatial domain. Another ap-

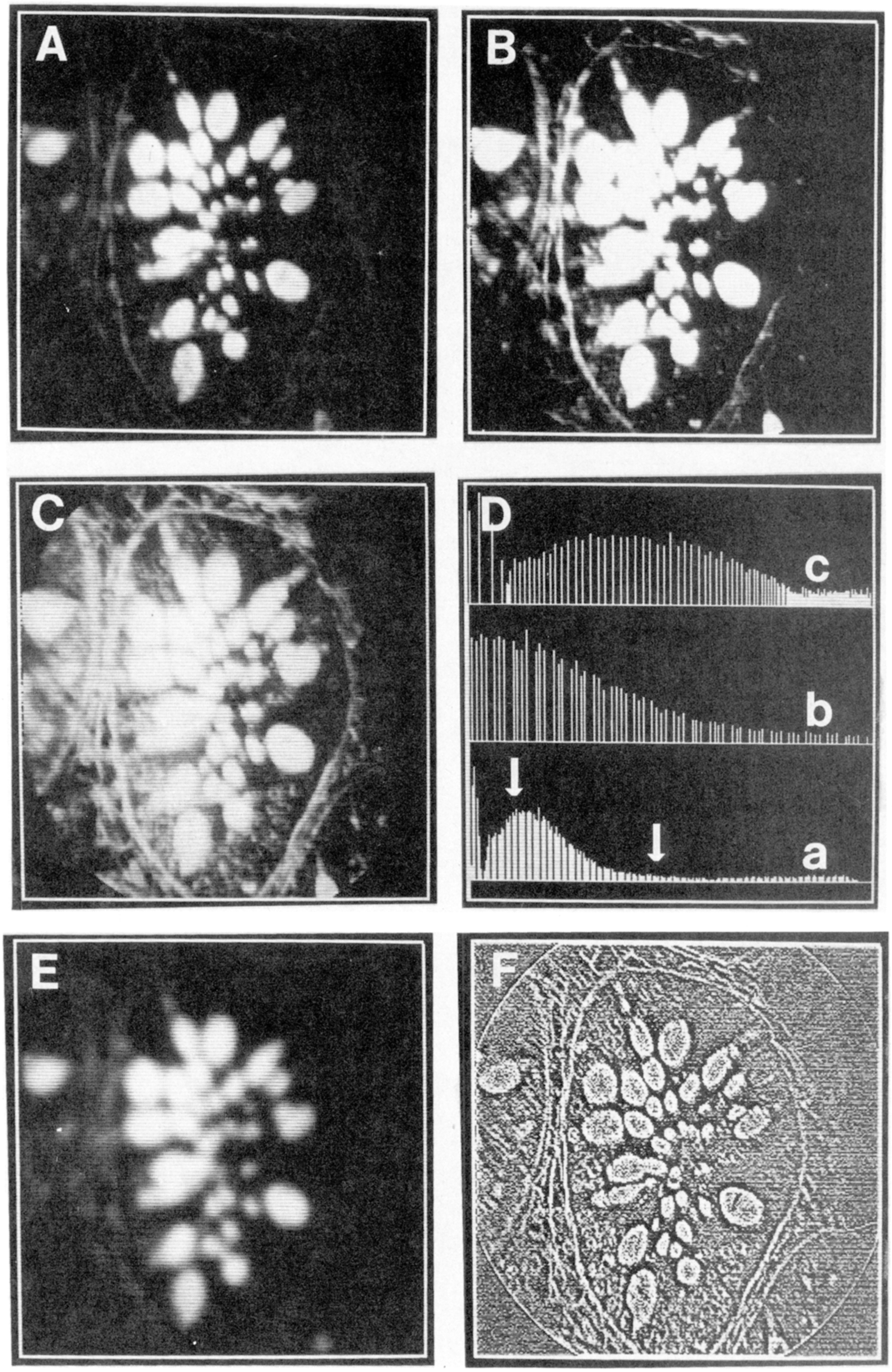


Figure 4. Examples of image enhancement. (A) Original $^{40}Ca^{+}$ image of the goblet cells in human colon. The image is 256 × 240 pixels. The pixel intensity is expressed as, and proportional to, the brightness. (B) Contrast-enhanced image obtained by clipping the original image between gray levels 30 and 120 and linearly mapping to the entire gray-level range from 0 to 255. Notice the superior structural details of the noncellular components compared to those of the cells. (C) Histogram-equalized image. Notice the same tradeoff between the structural details of the cells and the noncellular components similar to those of the contrast-enhanced image. (D) Histograms of corresponding images A, B, and C. The arrows in A cover the clipped gray-level range from 30 to 120. (E) Noise-cleaned image with a box filter of window size 9 × 9. Notice the noise-cleaning effects (as improved uniform brightness inside the cells) and the feature-blurring effects (as increasing unresolved cells). (F) Edge-crispened image with a Laplacian filter. Notice the improved cell distinction and the superior feature demarcation.

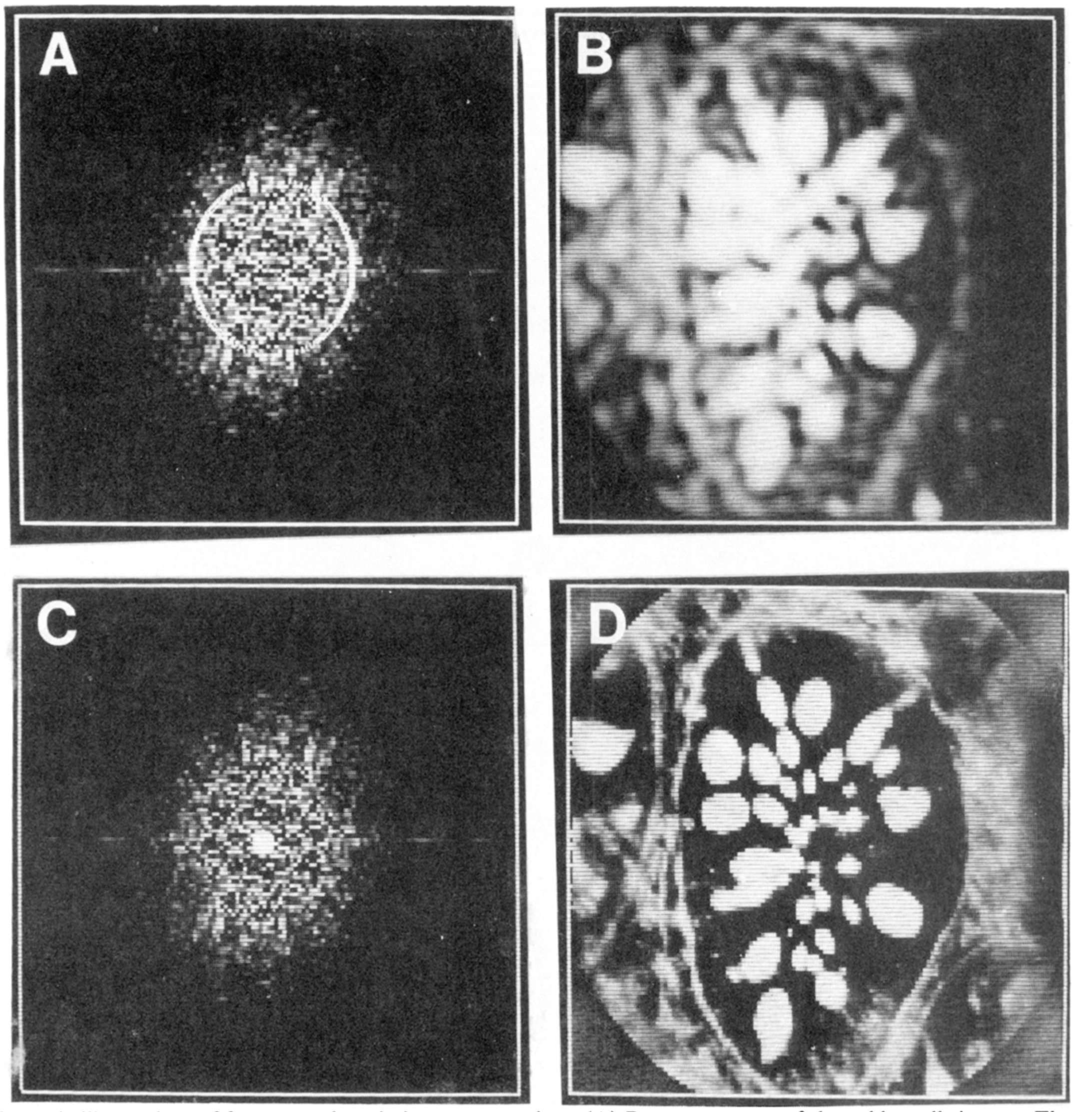
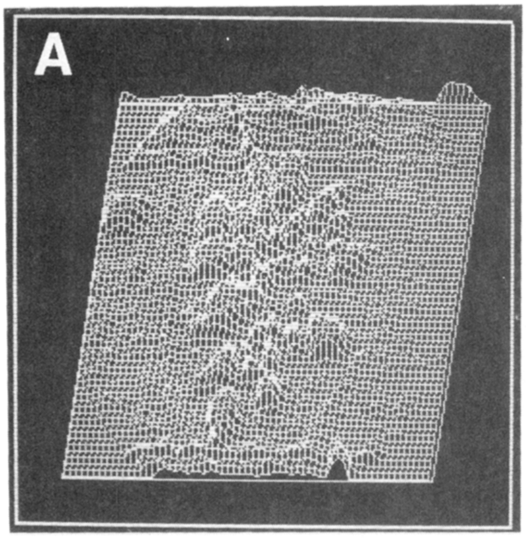


Figure 5. Schematic illustrations of frequency domain image processing. (A) Power spectrum of the goblet cells image. The low-pass filter with a radius of 21 pixels contains 85% of the image power. The retained power is mainly of low-frequency components enclosed by the circle. The origin (the dc level or the lowest frequency component) is located at the center of the 128 × 120 pixel spectrum. (B) Resulting low-pass-filtered image. Notice the same noise-cleaning and feature-blurring effects similar to those of the box-filtered image of Figure 4E. (C) Power spectrum of the goblet cells image. The high-pass filter with a radius of 3 pixels excludes 10% of the image power. This unused power is mainly of low-frequency components covered by the solid circle. (D) Resulting high-pass-filtered image. Notice the same improved cell distinction and better feature demarcation effects similar to those of the Laplacian-filtered image of Figure 4F. Both filtered images are histogram-equalized for better contrast.

proach is to take advantage of the fact that the spatial frequency components of the OI, which are usually low, are easily distinguished from those of the N (usually high). Hence by Fourier transform of the DI, the separation of OI from N becomes a straightforward operation. One simply deletes the high-frequency-noise components with an appropriate low-pass filter. The effects of this low-pass filtering are equivalent to the spatial domain noise-cleaning process. Edge-crispening processes can also be simulated in the frequency domain with a high-pass filter. One additional benefit of frequency domain processing is the deconvolution technique, which removes the degrading effects of the instrument-transfer function, for so-called "image restoration".[19] Figure 5 demonstrates various frequency domain processing techniques.

7. Interactive Interrogation. For SIMIPS commands to perform the desired operations, some specific operational parameters are necessary. They are entered from the keyboard by the user because of the convenience and precision of the keyboard input, i.e., for discrete numerics and alphanumerics. For pixel-related arguments, cursors (maneuvered manually via a joystick) are used to direct the communication between man and machine. Specific locations of pixels could be selected with which various commands may be performed, such as reading (or writing) or generating standard X–Y plots of these identified pixels. With these implementations (better called the "image editor"), the user can quickly input and modify pixels, delineating features of interest and selecting an operational region of interest. Futhermore, when the desired histogram is drawn, a gray level transforming function can be derived and loaded into the LUTs. Nonlinear image enhancement is thus possible.

8. Display. The most appropriate way to deliver the large amount of image-data information to a human is by displaying it. For a single image, the pixel location is coded as the row and column offset relative to the origin, while the intensity may be expressed either in color or as deflection above the X–Y plane. The former approach yields a conventional two-dimensional color image familiar to human vision perception, while the latter approach generates a three-dimensional-perspective image of the sample surface as Figure 6 shows. Explanatory alphanumerics can be overlaid on these displayed images to further clarify their information content.



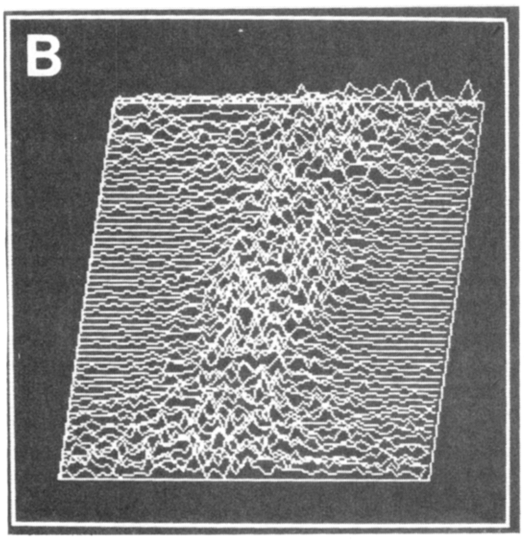


Figure 6. Three-dimensional-perspective images (of 64 × 60 pixels). (A) Goblet cells of Figure 4A. Notice only the cells stand out as bumps. (B) Power spectrum of Figure 5A. Notice most of the power is contained in the vicinity of the origin and along the vertical direction.

9. Segmentation. The objective of segmentation is to extract and classify the image into recognizable features of interest. The technique used is based on the gray-level differences between features, which are of different physical origin. It allows the user to manually specify the thresholds, or, in a more robust way, if the number of kinds of features are known a priori, the optimal thresholds can be obtained automatically by decomposing the histogram into components.[20] These thresholds are then used to reduce the gray-level values into several distinct states, each representing a particular class of the feature. Figure 7 shows the optimally segmented image of goblet cells.

10. Information Extraction. All the previous operations have one goal in common, namely, to extract the most reliable information in the most condensed and comprehensive form. From a single image, SIMIPS is capable of measuring the feature-specific parameters, such as shape, size, number, and statistics (so-called "feature descriptions"). Moreover, "similarity descriptions" are also obtainable by assembling the feature descriptions from individual images through pattern-recognition techniques.[21] With established calibration schemes, all these parameters can be simply converted into numbers of physical meaning to express the compositional morphological characteristics of the analyzed samples. Figure 8 outlines this paradigm of image analysis. It is worth mentioning that the addition of pattern-recognition techniques to DIP broadens the information content as well as ensures better reliability of the information. Consequently, better selection and parameterization of DIP algorithms are obtained.

Table I. Disk *I/O* Speed (s)

image size, pixels	type	512[a]	1024[a]	2048[a]	4096[a]	8192[a]
128 × 120	read	2.10	1.18	0.97	0.83	0.77
256 × 240	read	6.73	3.72	2.18	1.63	1.32
512 × 480	read	25.47	13.52	7.35	5.12	3.92
128 × 120	write	3.02	2.35	1.97	1.82	1.68
256 × 240	write	7.63	4.92	3.45	2.70	2.33
512 × 480	write	26.17	12.72	9.35	6.35	4.83

[a] Bytes.

Table II. TRAPIX *I/O* Speed (s)

image size, pixels	type	1[a]	2[a]	4[a]	8[a]	16[a]
128 × 120	read	0.17	0.08	0.03	0.02	0.02
256 × 240	read	0.60	0.30	0.15	0.08	0.03
512 × 480	read	2.27	1.13	0.57	0.28	0.15
128 × 120	write	0.15	0.08	0.03	0.02	0.02
256 × 240	write	0.50	0.25	0.13	0.07	0.03
512 × 480	write	1.90	0.93	0.47	0.25	0.12

[a] Lines.

SYSTEM PERFORMANCE EVALUATION

A practical interactive DIP system must be capable of accepting and executing commands and displaying results in a period of time that is bearable to the user. This high-speed requirement can be achieved by employing a dedicated computer and special-purpose hardware (see Hardware Configuration section) or, for more flexibility, by implementing efficient algorithms. In general the computational speed is measured by, and is inversely proportional to, the elapsed time (which is the time required to finish the processing task after the commands are issued by the user). The elapsed time can be partitioned into *I/O* (input/output) time, which is required to transfer data and commands between the central processing unit (CPU) and peripheral devices (such as the disk and the TRAPIX), and CPU time, the processing time required by the CPU.

1. Disk *I/O*. Since the operation of SIMIPS is heavily *I/O* based, all programs as well as data need to be loaded into the core memory first, and subsequent processing requires frequent data transfer between the TRAPIX and CPU. A good starting point for improving the speed is to analyze the operational bottlenecks associated with these two kinds of *I/O* operations as a function of parameters that are software controllable. Table I details the elapsed time to transfer images of 128 × 120, 256 × 240, and 512 × 480 pixels, 16 bits per pixel, between the disk and the PDP-11/34A. These images can be transferred one row (i.e., 128, 256, or 512 pixels) or several rows at a time, which is determined by the available size of the *I/O* buffer used. From the table, it is clear that the larger the buffer size, the smaller the elapsed time. However, the relative speed improvement decreases as the buffer size increases. Moreover, availability of core memory limits the implementation of a larger *I/O* buffer size. Therefore, a fixed amount of core memory (8192 byte) is set aside specifically for the *I/O* buffer, to compromise the conflict requirements of speed and memory.

2. TRAPIX *I/O*. The same arguments about the conflicting requirements of speed and memory also hold true for the *I/O* operation between the TRAPIX and the PDP-11/34A. However, the time saved (see Table II) is not as much as that saved by the disk. For example, for an image of 256 × 240 pixels, the read operation is varied from 0.60 to 0.03 s for the TRAPIX and from 6.73 to 1.32 s for the disk while changing the buffer from 512 to 8192 byte. In addition, the software

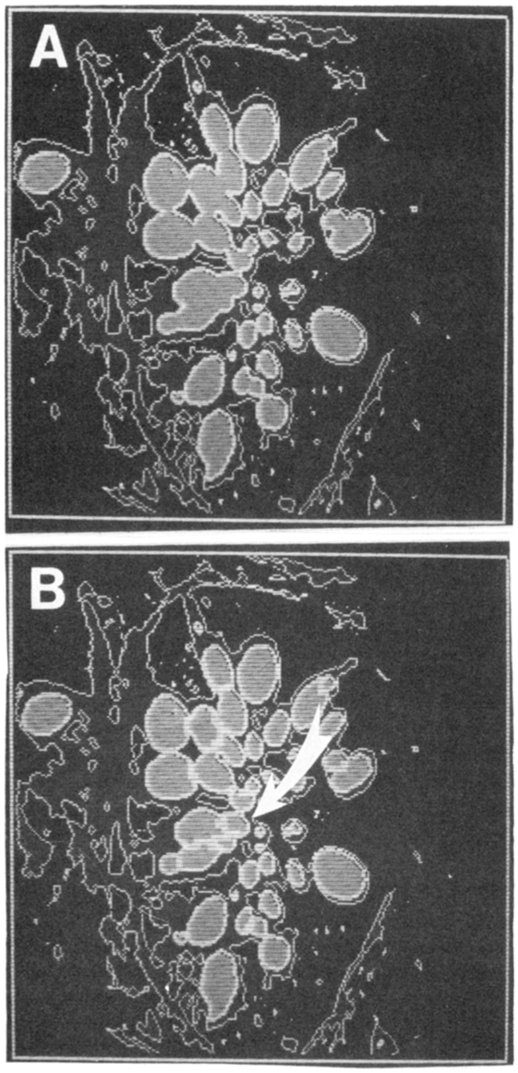


Figure 7. (A) Segmented image of goblet cells with superimposed boundaries. The segmented image is obtained by thresholding the original image (with threshold values of 56 and 141 gray levels) into three feature classes (as different brightness). (B) Revised version (manually specified with the image editor) of the above segmented image further delineates the unresolved cells. Compared to A, notice the superior cell demarcation shown by the arrow.

overhead required to decipher the individual row from a block of pixels is expensive. Thus the image data from the TRAPIX is read in one row at a time.

3. CPU Time. Convolution-based techniques, such as noise cleaning and edge enhancement, are very powerful general purpose image-processing tools. They can be described mathematically by

$$g(x,y) = \sum_{i=-m}^{m} \sum_{j=-m}^{m} h(i,j)\, f(x+i,y+j) \qquad (2)$$

where the new pixel, $g(x,y)$, at location (x,y) is generated by summing the products of the old pixel $f(x+i,y+j)$ with the convolution kernel $h(i,j)$ over the entire window of size $(2m + 1) \times (2m + 1)$ pixels. A direct approach is to read in all $(2m + 1)$ rows each time the new-role pixels are calculated. However, in that technique, each row is read redundantly ($2m$ times). This inefficiency can be remedied by employing a row pointer to update the relative row orders and only reading in the new row. The same philosophy of exploring existing information, such as image rows already in the computer core memory, can be further extended.[22] For example, referring to Figure 9, a simple 3 by 3 box filter operation for pixel $f(2,2)$ can be calculated as

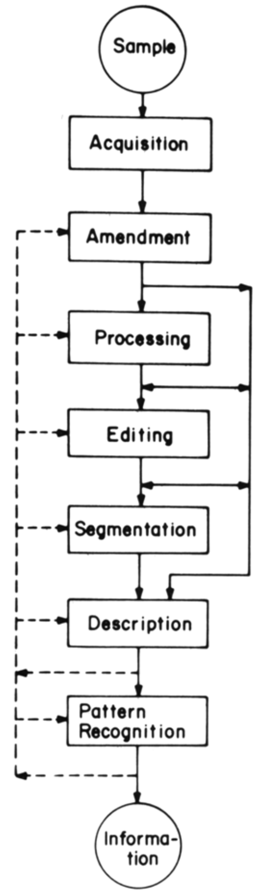


Figure 8. Paradigm of image analysis. Shown is the block diagram of the information extraction process. The processing and editing stages are optional. Feature descriptions are generated from the combined inputs of the segmented image and the amended (processed or edited) image. The feedback (shown as dashed lines) of feature descriptions and similarity descriptions (obtained from pattern recognition) may assist the user in adjusting the individual processing scheme (shown as blocks). This cyclic process of DIP–pattern recognition is repeated until the desirable information is extracted.

$$g(x,y) = (1/9) \sum_{i=-1}^{1} \sum_{j=-1}^{1} f(x+i,y+j) \qquad x = 2,\ y = 2 \qquad (3)$$

or as (after expanding the terms)

$$\text{buffer} = f(1,1) + f(1,2) + f(1,3) + f(2,1) + f(2,2) + f(2,3) + f(3,1) + f(3,2) + f(3,3) \qquad (4a)$$

$$g(2,2) = \text{buffer}/9 \qquad (4b)$$

This operation requires nine additions and one division. The new pixel, $g(3,2)$, can be calculated with fewer mathematical operations, such as

$$\text{buffer} = \text{buffer} - f(1,1) - f(1,2) - f(1,3) + f(4,1) + f(4,2) + f(4,3) \qquad (5a)$$

$$g(3,2) = \text{buffer}/9 \qquad (5b)$$

As the result, only six additions are required instead of nine. A $1 - (2m + 1) \times [(2/(2m + 1))/(2m + 1)]$ percentage of time is saved. This advantage in time saving is more evident as the window size increases. For example, from a window size of 3×3 to 15×15, the saving is increased from 33% to 87%. Figure 10 compares the elapsed time of this box-filtering technique, with and without this buffer-updating implementation, as a function of the window size. It is clear that the additional effort on efficient algorithm design pays generously with the reduction of elapsed time. This buffer-updating

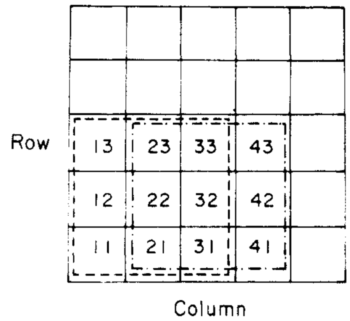


Figure 9. Example of convolution-based image enhancement technique using a simplified 5 × 5 pixel image with the location information shown. The 3 × 3 window (shown as dashed lines) encompasses the pixels for calculating the enhanced pixel.

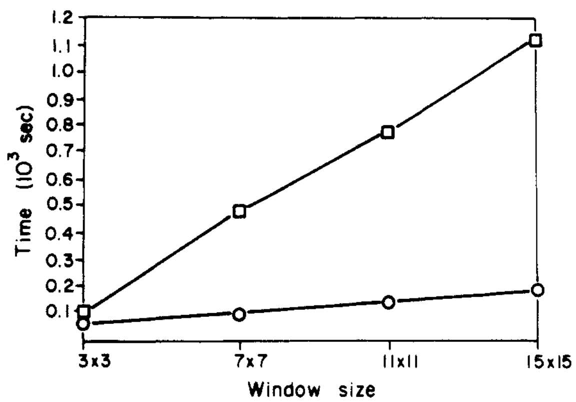


Figure 10. Comparison of elapsed time as a function of window size for an image of 256 × 240 pixels: (O) with buffer-updating technique; (□) without buffer-undating technique.

algorithm, in both column and row directions, thus has been implemented on all convolution-based operations in SIMIPS.

CONCLUSIONS

A quantitative image analysis system, SIMIPS, is described as an effective tool to accelerate the extraction of the greatest possible amount of information from digital images. It is tailored to a research-oriented environment with a diverse function repertoire and convenient man–machine interface. The interactive nature allows the user to view the intermediate DIP results (in near-real time), which in turn can be used a guide to gear the ongoing processing direction. The support of batch operations and intrinsic programming capabilities facilitates the innovative design of elaborate problem-resolving schemes without the loss of accuracy. The applications of image analysis are numerous and diverse, and the specific operations required to describe the digital images are varied from one application to another. With the dual disciplines of DIP (for amendment and enhancement) and pattern recognition (for classification and measurement), the information extraction capabilities of SIMIPS are further realized.

SIMIPS has been implemented and utilized for a variety of research projects. It has been used to develop practical techniques for interpreting the spatial resolution and image quality in order to quantify the inherent chemical information in the stigmatic ion micrographs.[10] The proposed objective criteria proved essential for later improvement of spatial resolution[11] and the eventual development of high-resolution imaging techniques.[23] It has also proved essential for the study of dark-field ion microscopy for structural contrast enhancement.[15] The advantage of combining applications of DIP and pattern-recognition technique is manifested in the complementary study of imaging intracellular free and total calcium in cultured cells.[24] These facts reveal that DIP in ion microscopy is just beginning to approach its full potential.

However, DIP alone is not without drawbacks. The existence of a vast variety of DIP algorithms makes the adaption of a specific algorithm to fit the relevant problem a nontrivial task. The incomplete understanding concerning the image-generating process and the improper practice of DIP generally lead to biased results, which can be unnoticeable to the user. A systematic and objective evaluation of the performance of the most popular DIP algorithms is necessary and is currently under study in this laboratory. Future research topics will concentrate on increasing the similarity description capabilities by combining three-dimensional image analysis and correlational image analysis to fully explore the inherent multidimensional information generating capability of SIMS.

ACKNOWLEDGMENT

We thank Mark Moran for helpful discussions and suggestions and Shan-Mei Hsu for writing part of the programs. This work was supported by the Cornell Materials Science Center, the National Science Foundation, the Office of Naval Research, and the National Institutes of Health.

REFERENCES AND NOTES

(1) Morrison, G. H.; Slodzian, G. "Ion Microscopy". *Anal. Chem.* **1975**, *47*, 932A–943A.
(2) Patkin, J. A.; Morrison, G. H. "Secondary Ion Mass Spectrometric Image Depth Profiling for Three-Dimensional Elemental Analysis". *Anal. Chem.* **1982**, *54*, 2–5.
(3) Steiger, W.; Rüdenauer, F. G.; Ernst, G. "Display of Three-Dimensional Microprobe Data". *Anal. Chem.* **1986**, *58*, 2037–2040.
(4) Bryan, S. R.; Woodwars, W. S.; Linton, R. W.; Griffis, D. P. "Secondary Ion Mass Spectrometry/Digital Imaging for the Three-Dimensional Chemical Characterization of Solid State Devices". *J. Vac. Sci. Technol., A* **1985**, *A3*, 2102–2107.
(5) Ramos, L. S.; Beebe, K. R.; Carey, W. P.; Sanchez, M. E.; Erickson, B. C.; Wilson, D. E.; Wangeu, L. E.; Kowalski, B. R. "Chemometrics". *Anal. Chem.* **1986**, *58*, 294R–315R.
(6) See: "Ion Microscopy and Image Analysis" Section of papers in "Secondary Ion Mass Spectrometry SIMS V"; Benninghoven, A., Colton, R. J., Simons, D. D., Werner, H. W., Eds.; Springer: Berlin, 1986; pp 232–279.
(7) Karagacin, M. E.; Kowalski, B. R. "Ion Intensity and Image Resolution in Secondary Ion Mass Spectrometry". *Anal. Chem.* **1986**, *58*, 2300–2306.
(8) Lepareur. *Rev. Tech. Thomson-CSF* **1980**, *12*, 225–265.
(9) Furman, B. K.; Morrison, G. H. "Direct Digitization System for Quantification in Ion Microscopy". *Anal. Chem.* **1980**, *52*, 2305–2310.
(10) Bernius, M. T.; Ling, Y. C.; Morrison, G. H. "Evaluation of Ion Microscopic Spatial Resolution and Image Quality". *Anal. Chem.* **1986**, *58*, 94–101.
(11) Bernius, M. T.; Ling, Y. C.; Morrison, G. H. "Improved Spatial Resolution of the CAMECA IMS-3f Ion Microscope". In "Secondary Ion Mass Spectrometry SIMS V"; Benninghoven, A., Colton, R. J., Simons, D. S., Werner, H. W., Eds.; Springer: Berlin, 1986; pp 245–248.
(12) Brenna, J. T.; Moran, M. G.; Morrison, G. H. "Versatile Video Tape System for Storage and Selective Retrieval of Ion Images for Digital Acquisition and Processing". *Anal. Chem.* **1986**, *58*, 428–433.
(13) Harris, W. C.; Chandra, S.; Morrison, G. H. "Ion Implantation for Quantitative Ion Microscopy of Biological Soft Tissue". *Anal. Chem.* **1983**, *55*, 1959–1963.
(14) Patkin, A. J.; Chandra, S.; Morrison, G. H. "Differential Sputtering Correction for Ion Microscopy with Image Depth Profiling". *Anal. Chem.* **1982**, *54*, 2507–2510.
(15) Bernius, M. T.; Ling, Y. C.; Morrison, G. H. "Dark-Field Stigmatic Ion Microscopy for Structural Contrast Enhancement". *J. Appl. Phys.* **1986**, *59*, 3332–3338.
(16) Pratt, W. K. "Digital Image Processing"; Wiley: New York, 1978; pp 307–344.
(17) Gonzalez, R. C.; Wintz, P. "Digital Image Processing"; Addison-Wesley: Reading, MA, 1977; pp 169–177.
(18) Stika, K. M. Doctoral Thesis, Cornell University, Ithaca, NY, 1980.
(19) Vandeginste, B. G. M.; Kowalski, B. R. "Spatial Enhancement and Restoration of Chemical Images from Secondary Ion Mass Spectrometry and Ion Scattering Spectrometry". *Anal. Chem.* **1983**, *55*, 557–564.
(20) Velasco, F. R. D. "Thresholding using ISODATA Clustering". *IEEE Trans. Syst., Man, Cybern.* **1980**, *SMC-10*, 771–774.
(21) Fu, K. S.; Rosenfeld, A. "Pattern Recognition and Image Processing". *IEEE Trans. Comput.* **1974**, *C-25*, 1336–1346.
(22) McDonnell, M. J. "Box-Filtering Techniques". *Comput. Graphics Image Process.* **1981**, *7*, 65–70.
(23) Bernius, M. T.; Ling, Y. C.; Morrison, G. H. "High Resolution Imaging with the Stigmatic Ion Microscope". *J. Appl. Phys.* **1986**, *60*, 1904–1912.
(24) Chandra, S.; Gross, D.; Webb, W. W.; Morrison, G. H. "Imaging of Intracellular Free and Total Calcium in Cultured Cells". (Manuscript in preparation.)