Software Development For TEXAC System*

Chan Mo Park, Robert Ledley and Menfai Shiu
National Biomedical Research Foundation
Georgetown University Medical Center
Washington, D.C. 20007 U.S.A.

***This work was supported by NIH Grants GM 10797 and RR 05681.**

Abstract

Automatic pattern recognition using computers has many potential applications. In many cases successful automatic pattern recognition procedures require an analysis of texture of a picture by a conventional digital computer; but this involves a type af computing that is at the present time extremely time-consuming and thus very expensive. In response to this situation, TEXAC was designed and built which is capable of most whole picture operations at a rapid television rate of 1/30th of a second. The results of operations are displayed continuously on color and black-and-white television monitors. This paper presents the architecture of the TEXAC, the picture processing language developed and some of the results of TEXAC whole picture operations.

1. Introduction

Automatic Pattern recognition has brought much attention in recent years. Many potential applications of computer pattern recognition, however, have not been successfully carried out on a feasible basis, because of time-consuming operations required in the analysis of the texture of a picture by a digital computer. This problem occurs because texture analysis involves the comparison of each picture point with every other point of the picture. For example, suppose the picture to be analyzed for texture has 500 lines and 500 picture points per line and the reasonable neighborhood size is 20 points X 20 points. If 10 assembly-language instructions are required per neighborhood operation, then $(500\times500)\times(20\times20)\times10=10^9$ instructions must be executed for each complete picture-texture operation. Assuming the computer executed 10^6 instructions per second, each texture-type operation would take 10^3 seconds or 16.7 minutes. The cost of executing a reasonably useful algorithm would thus be extremely high.

A computer capable of rapid picture analysis would have potential applications in such areas as the analysis of chest X-rays, Papanicolaou smears, differential white blood counting, and so forth. We therefore designed and developed the TEXAC (Texture Analysis Computer), an auxiliary computer capable of performing most picture operations quickly and economically (see Figure 1).

The principal advantage of the TEXAC is its ability to operate on the picture as a whole at TV rates, and display the results continuously on the TV monitors. In other words, all the picture points are considered as one "picture-word" in the TEXAC and once instructed by the host computer to perform an operation, TEXAC operates on the whole picture at a very high speed, taking 1/30th of a second (standard TV rate).

In order to instruct the TEXAC whole picture operations easily we have developed a picture processing language. Computer programs implementing various image processing algorithms are also developed by using this picture processing language.

2. Hardware System

Basically, the TEXAC employs three picture memories, controls for the memories, a high-speed, special purpose processor which interprets the computer command and channels the flow of the steps involved in the command, a unit that routes the data from and to the memory units, input-output devices and the main computer, and various registers and buffers (see Figure 2).

2.1 High Speed Refresh Memory Units:

There are three such units for three "picture words" A,B and C. The TEXAC raster size is 512 horizontal lines with 768 pixels or points per line. Each pixel has 8 bits (256 gray levels). The picture memory is built up from the new self-refreshing CCD units (charge-coupled devices).

The slave timing unit of each memory keeps circulating these points in each plane synchronously at TV rates with appropriate intervals for horizontal and vertical retrace, just as in a TV camera or monitor, except that the information is in digital form. The information to and from the memory is in serial-parallel form. The 8 bits of pixel are handled in parallel and the pixels are handled serially. Operations on the memory are repetitive until the entire (or pre-programmed segment of the) picture is processed in 1/30th of a second.

Basic Memory Specifications:

Visible elements per scan	768
Visible lines per scan	512
Fields per frame	2
Video channels	3
Element rate/channel	16.1×10^6/sec.
Element dwell time	62.1ns
Bits per element per channel	8

Note that the time per element, 62.1 nanoseconds, is nearly impossible to achieve even with today's high-speed memory devices used in the conventionl random access mode. The TEXAC memory design and its application in the special purpose auxiliary computer is superior to the conventional memories in speed, performance and price.

2.2 Inputs:

Input to the TEXAC can be from a number of different types of scanners, such as a television raster camera. The picture to be scanned can be from an X-ray plate, a microscope slide or a photograph. As designed in our model, TEXAC uses a standard television camera as the input device. The camera is directly driven by the TEXAC master timing control, which generates the sync and blanking signals for the camera video signal as per EIA standards.

Along with the digitized signals from the television camera, input to the TEXAC also comes from the host computer via the line buffer. The A/D converter and the line buffer are described below.

2.3 High-Speed Video Processor Unit:

This unit consists of an 8-bit Test ALU (Arithmetic Logic Unit) and an 8-bit Operation ALU. The inputs to the ALU's are the three memories, the line buffer and the digitized video, and two of these inputs are selected as operands of the ALU's. The operation to be performed is set up depending on the results of the Test ALU. The 8 bits of the results can be shifted or rotated under program control and sent to one or more of the outputs, the outputs being the three memories, the three memories, the displays and the line buffer.

2.4 Intensity Transformation Units (ITU'S):

These are 256×8 bit RAM (Random Access Memory) units. Data from a memory may be channelled through an ITU where separate intensity values can be assigned to each element of the picture memory itself. The transformed levels can be used for visual picture enhancement or for computations. With two additional ITU'S, the C memory can be used for the RBG (Red, Blue, Green) color display.

2.5 Outputs:

With 3 high-resolution standard TV monitors and a color monitor, the contents of each memory can be seen as a video picture that is channelled through the corresponding digital to analog converter.

A second type output is to the host computer via the line buffer. Any given picture-memory output can also be routed back as an input to itself or the other memories.

2.6 Digital Video Crossbar Switch:

This unit operates as a selector or router of the signals from either the video camera interface or the picture memories to the high-speed video processor or the line buffer. The crossbar switch selects input data paths as well as output data paths.

2.7 A to D Converter:

For successful evaluation of pictorial texture, at least a 32 gray-scale discrimination is desirable. In TEXAC, the video signal from the TV camera is digitized with 6 bits per pixel, that is 64 gray levels. The analog to digital converter (video camera interface in the block diagram) is a custom-designed unit involving 64 reference levels and comparators operating in parallel, so that conversion of pixel analog video information into 6 bits is completed before the next pixel is sampled at the desired video rate. In general four picture frames are read in from the TV camera to create a 256 gray-level picture.

2.8 Line Buffer:

The line buffer is used for temporary storage of 1 line of data, which can then be one of the inputs to the TEXAC or the output to the host computer from the TEXAC. The data from the slower host computer can thus be a part of the much faster TEXAC data flow.

2.9 Host Computer PDP 11/34 and Unibus Interface:

TEXAC takes advantage of the PDP Unibus concept which allows control over many peripherals on the bus. Thus processing and storage of nonpictorial data such as identification, history, etc., can be done external to TEXAC. The general purpose computer can perform further manipulations on TEXAC results, program the TEXAC and carry out further pattern recognition functions in a unified system. The interface between the PDP-11 and the TEXAC is standard interface circuitry.

The TEXAC and the Minicomputer communicate with each other through a number of special-purpose registers in the TEXAC. For example, the crossbar switch consists of Input Dath Path and Output Data Path Registers. Data transfer takes place via a Data Register. Then there are registers for the operation codes, the picture displacement controls, operation mode control, etc.

As an example of the function of the control/data registers, We can examine what happens with the TEXAC macro instruction MOV A,B. With MOV A,B the Input Data Path and Output Data Path Registers are loaded with the bit patterns that select Memory A as the input or source memory and Memory B as the output or destination memory. The bit pattern in the Operation Code Register becomes such that output from the ALU is the input. The Control/Status Register tests the ready bit and sets up the go bit which makes the operation to start.

3. Software System

TEXAC on its primitive level is programmed by loading the control/data registers with the proper contents by employing the "load" instructions of the host minicomputer, in this case a DEC PDP-11/34 computer. In order to expedite the programming procedure, however, TEXAC macro instructions have been developed.

3.1 TEXAC Macro Instructions:

The macro intructions are designed to facilitate implementation of the picture processing algorithms. The basic instruction format it a three letter operation code followed by, in general, one to five operands. In most cases three addresses are used as operands, two for the source inputs (such as image refresh memories and/or constant registers) and the other the other the output destination, which can be one or more memories. The operands are image refresh memories, integer constants, the line buffer, and other proper control/data registers.

The library of TEXAC macro instructions includes transfer, arithmetic, logical, and comparison instructions. In addition, there are special instructions such as spectrum count, spatial shifts, and intensity transformation instructions. For example, the transfer instruction MOV A,B moves the entire contents of Memory A to Memory B, pixel by pixel in 1/30th of a second. The arithmetic instruction ADD A,B,C causes the entire contents of Memory A to be added to those of Memory B, and the total contents to be transferred to Memory C, again pixel by pixel, in 1/30th of a second. SUB B,B,B clears Memory B in 1/30th of a second. If an instruction CGT A,K50, B,K100, B follows the instruction SUB B,B,B then we obtain the binary-valued picture (0 and 100) of the image of Memory A in Memory B. In other words any pixel in Memory A whose gray value is greater than 50 will create a pixel with the gray value of 100 in Memory B at the corresponding position while those pixels whose gray values are not greater than 50 will leave the corresponding pixels in Memory B as they were (i.e.,0). A list of selected TEXAC macro instructions is included in Table 1. Each operand represents one or more memories or an integer constant.

3.2 Interface with General Purpose Language:

For user convenience the TEXAC can be operated not only from the console using the macro instructions, but also from a familiar general purpose computer language such as FORTRAN. TEXAC macro instructions are included in the system library of the host computer, so that they cane be called by a user program in the form of subroutines In such a situation, the operation code becomes the subroutine name, and the operands are specified as the arguments of the subroutine called.

One sample FORTRAN program which utilizes the TEXAC macro instructions is given in Table 2. This program does the 10-step expansion of an artist's sketch of a dog (see Figure 3). The main part of the algorithm is implemented by the statements in the lines 18 through 28 which includes moving the binary image of the picture in Memory C to Memories A and B, shifting the image in Memory A by 10 pixels (10 steps) to the left, right, up or down, comparing the images in Memories in A and then creating a new layer in Memory C. Total time required including the picture reading time from the disk is less than 25 seconds.

3.3 Algorithm Development:

Many picture processing algorithms, including such whole picture manipulations as enhancement of the edges of objects, discrimination of objects, texture synthesis, elimination of noise, determination of picture properties, and enlargement and contraction of objects, have been developed for the TEXAC. These algorithms generate whole-picture results in a fraction of a second, whereas it would take an enormous amount of time to perform similar operations using a conventional computer.

Some of the results obtained by applying the TEXAC algorithms to pictures of an artist's sketch of Snoopy and muscle fibers are shown in Figures 4 and 5.

Figure 4a is an artist's sketch of Snoopy as seen on a TV monitor. The algorithm developed for the 10-step dog expansion has been modified to expand the object boundary inward by 1-step and applied to the picture of Figure 4a to obtain the result shown in Figure 4b which illustrates the skeleton of the original Snoopy picture. Such picture manipulation may ae useful in data reduction for storing images.

The set of photographs, Figures 5a to 5j, prtray TEXAC whole picture operations on a muscle fiber picture. Figure 5a is an original muscle fiber picture as read in from a microscope. In Figure 5b the original muscle fiber image has been smoothed by 16×16 neighboring pixels. The next picture, Figure 5c, shows the difference between the original and the smoothed images; here the boundaries of the muscle fibers have been enhanced by considering neighboring pixel points.[1] In Figure 5d, the broken boundaries are connected by applying the expansion-shrinking process.[2] Figure 5e is a histogram of the frequency distribution of the gray levels of the whole muscle fiber picture, and the cummulative frequency distribution is shown in the curve of Figure 5f. In the next histogram, Figure 5g, the background and all unimportant points have been eliminated in order to obtain a new frequency distribution chart. Figure 5h shows one cell of interest, specified by a cursor and shaded; its area is computed and then printed out on the console. In Figure 5i, the same cell is isolated and enlarged by a desired factor (in this case a factor of 3), while the texture of the cellis enhanced in Figure 5j.

4. Potential Applications

The TEXAC system has many potential applications in the fields such as biomedical pattern recognition, computer-assisted design,

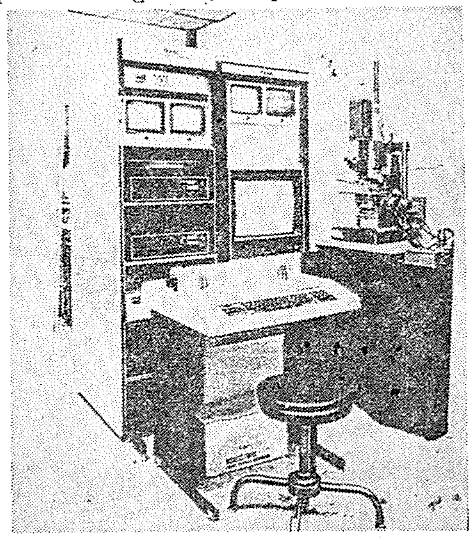
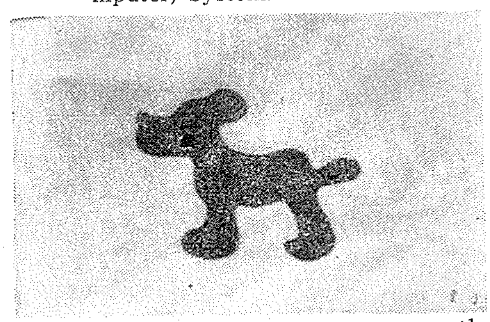
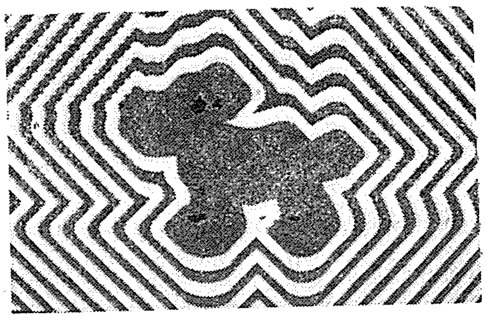


Figure 1. The TEXAC (Texture Analysis Computer) System.



3a. The Original Drawing as seen on the TEXAC TV Monitor.



3b. The Boundary Expanded 10 Steps at a Time.

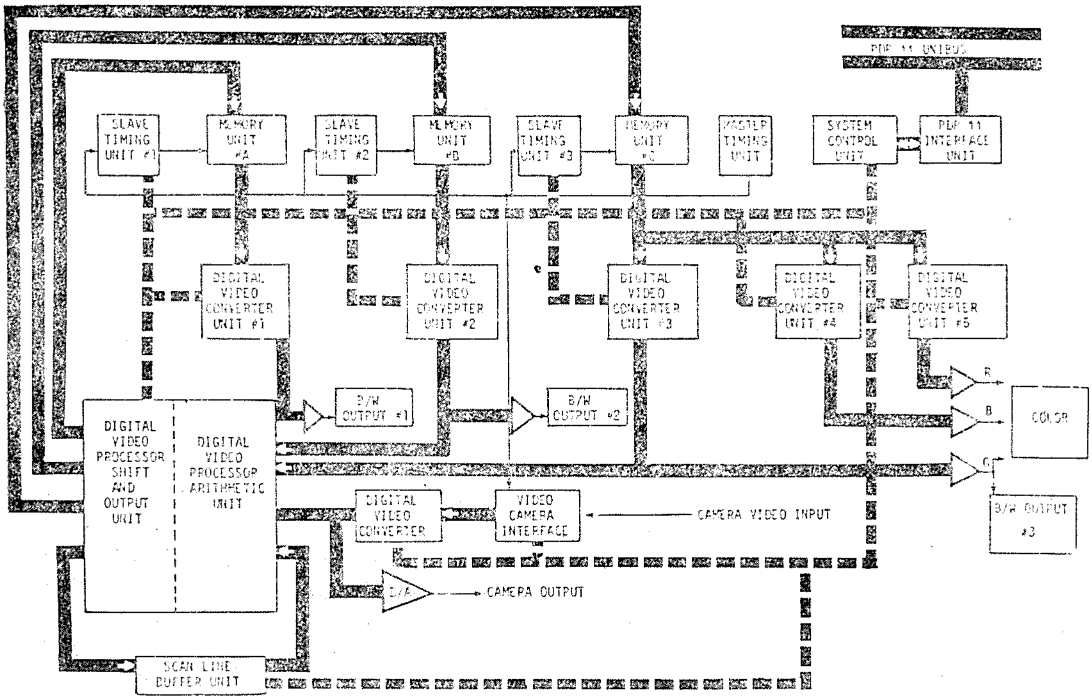
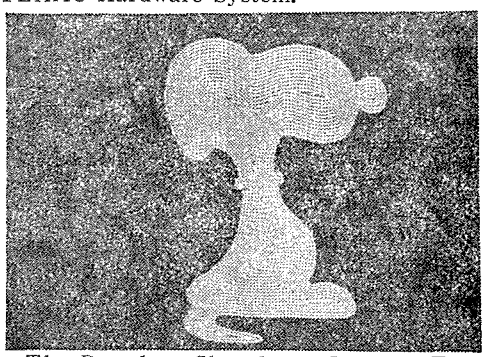
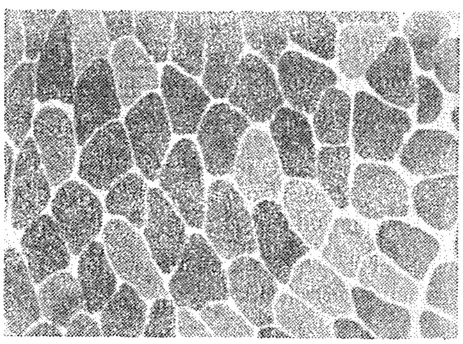


Figure 2. A Block Diagram of the TEXAC Hardware System.

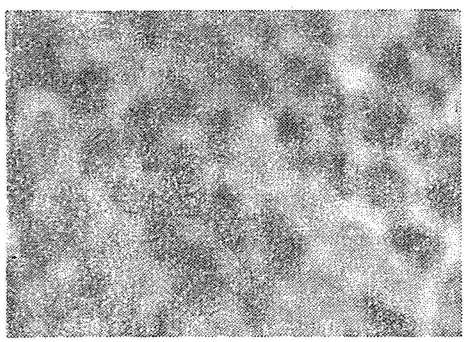
4a. The Original Drawing as seen on the TEXAC TV Monitor.



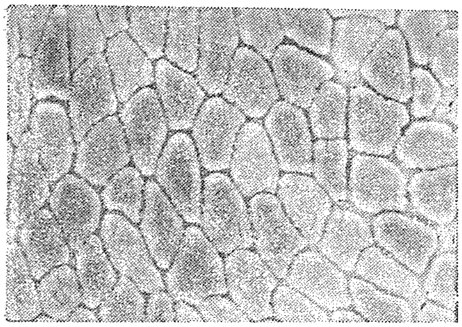
4b. The Boundary Shrunken 1 Step at a Time to give Skeleton.



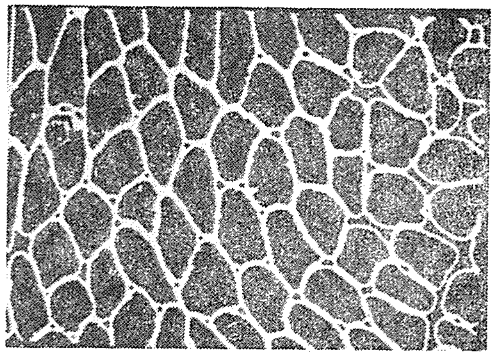
5a. The Original Picture as read in from a Microscope.



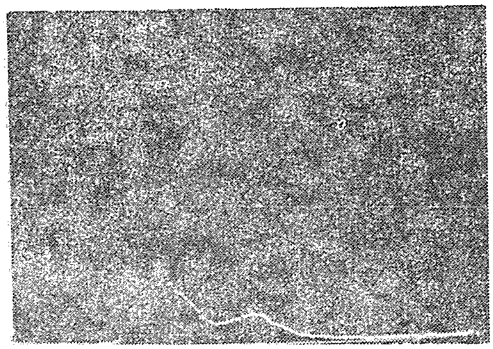
5b. The Original Picture smoothed by 16×16 Neighboring Pixels.



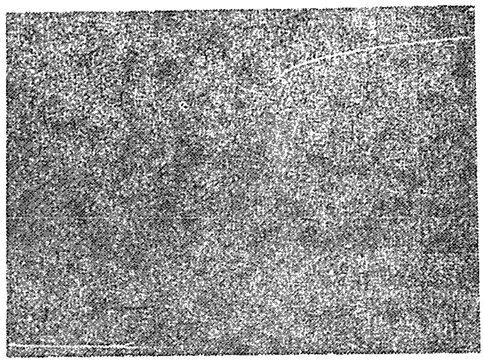
5c. The Muscle Fiber Boundaries Enhanced.



5d. The Broken Boundaries Connected.

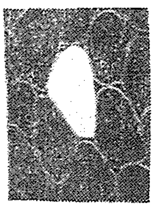


5e. The Frequency Distribution of the Gray-Levels of the Whole Muscle Fiber Picture of Figure 5a.

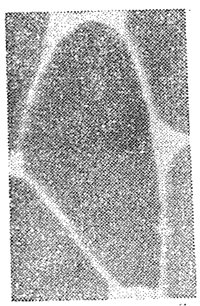


5f. The Cummulative Frequency Distribution of the Gray-Levels of the Whole Muscle Fiber Picture.

5g. The Frequency Distribution of the Gray-Levls of the Muscle Fiber Picture Which fall in the Range of the Mean±one standard deviation.



5h. A Cell of Interest Shaded.



5i. The Previous Shaded Cell is isolated and enlarged by a Factor of 3.

5j. The Texture o the Previous Isolated, Enlarged Cell Enhanced.

land data analysis and land planning, military image analysis and processing.

One illustration of the potential use of the TEXAC system is with chromosome spreads.[3] Here the chromosome spred could be digitized with our computer system and through the TEXAC programs the chromosome arms as well as the boundaries of the individual chromosomes could be more clearly distinguished (see Figure 6). This could result in a more efficient analysis of chromosome spreads.

There are certainly further possibilities for the TEXAC in the field of computerized biomedical and other pictorial pattern recognition, such as analyzing and processing histological tissue sections for medical pathology and other investigations of aggregates of cells. TEXAC will undoubtedly make an important contribution to this field with a system that is innovative, powerful, useful, fast, versatile, and economical.

TABLE LEGENDS

Table 1. Typical TEXAC Macro Instructions.

Type	Op-Code	Operands	Function
Transfer	MOV	A,B	B←A
	CAM	A,K	A gets picture image from camera with camera offset of K
	RDK	A	Read in desired picture image from a disk to A
	WDK	A	Write picture image in A to a disk
Arithmetic	ADD	A,B*,C	C←A+B
	SUB	A,B*,C	C←A−B
	AVE	A,B*,C	C←(A+B)/2
	MUL	A,B,K	B←A·K
	DIV	A,B,K	B←A/K
	CNT	A,B*,C*	Count spots in A which are B≦A<C
Logical	AND	A,B*,C	C←A∧B
	LOR	A,B*,C	C←A∨B
	XOR	A,B*,C	C←A EXOR B
	CMP	A,B	B←Ā
Compare	CEQ	A,B*,C,D*,E	If A=B, then C=D, or else C=E
	CGT	A,B*,C,D*,E	If A>B, then C=D, or else C=E
	CLT	A,B*,C,D*,E	If A<B, then C=D, or else C=E
Shift	SHF	A,K1,K2	Shift A horizontally by K1 and vertically by K2
	RES	A	Reset A to original position
Intensity Transformation	SPC		Load desired color spectrum
	ITS	A,K	If K=0, A uses intensity transformation table If K=1, A does not use intensity transformation table

K(K1,K2)=Integer constant
*=Memory or constant

Table 2. A Sample FORTRAN Program for Ten-Step Expansion of Dog.

Table 2. A Sample FORTRAN Program for Ten-Step Expansion of Dog

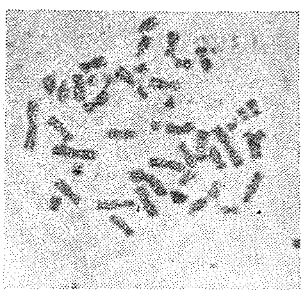
```
      PROGRAM DOGTEN
      INTEGER A,B,C,D,E,F,G,H,K1,K2
      DATA A,B,C,D,E,F,G,H,K1,K2/2,4,8,0,6,10,12,14,18,20/
      INTEGER IX(4),IY(4),NUM(4)
      DATA IX/10,-10,0,0/,IY/0,0,-10,10/,NUM/50,100,150,200/
      CALL EYS(C,0)
      CALL RES(H)
      CALL RDK(A)
      CALL CMP(A,A)
      CALL SUB(B,B,B)
      CALL SETCNT(80,150)
      CALL CST(A,K2,B,K1,B)
      CALL SHF(B,5,0)
      CALL MOV(B,C)
      CALL RES(H)
    5 READ(5,100) ICODE
  100 FORMAT(A1)
      DO 10 J=1,4
      CALL SUB(B,B,B)
      CALL SETCNT(25,0)
      CALL CST(C,B,B,K1,B)
      CALL MOV(B,A)
      DO 10 I=1,4
      CALL SHF(A,IX(I),IY(I))
      CALL SETCNT(NUM(J),0)
      CALL CST(A,B,C,K1,C)
      CALL RES(A)
   10 CONTINUE
      GO TO 5
      END
```


REFERENCES

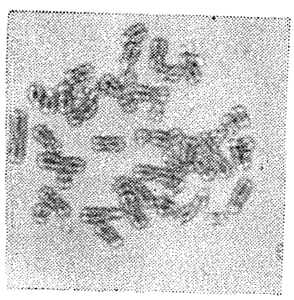
1. Ledley, R.S., "Texture Problems in Biomedical Pattern Recognition," Proc. IEEE Conf. on Decision and Control, 1972.
2. Park, C.M. and Rosenfeld, A., "Noise Cleaning in Digital Pictures," Electronic and Aerospace System Convention Record 1969, pp. 264—273, October, 1969.
3. Ledley, R.S., Lubs, H. A., and Ruddle, F.H., "Introduction to Chromosome Analysis," Computers in Biology and Medicine, 2(2) : 107—128, 1972.

FIGURE LEGENDS

Figure 3. Whole Picture Operations on an Artist's Sketch of Dog.
Figure 4. Whole Picture Operations on an Artist's Sketch of Snoopy.
Figure 5. Whole Picture Operations on a Pictue of Muscle Fibers.
Figure 6. Whole Picture Operations on a Picture of a Chromosome Spread.



6a. The Original Picture as read in from a Microscope.



6b. The Chromosome Arms Determined.

6c. The Chromosome Boundaries Determined.