Semper 6 *Plus*

TUTOR USER GUIDE

Synoptics

Trademarks:

Semper is a registered trademark of *Synoptics Limited.*
IBM PC, PC/XT, PC/AT are trademarks of International Business Machines Inc
Silicon Graphics Personal IRIS 4D is a trademark of Silicon Graphics Inc
Sun is a registered trademark of Sun Microsystems Inc
VAX is a registered trademark of Digital Equipment Corporation
PostScript is a trademark of *Adobe Systems Incorporated*

Manual number: SEM002
Update number: 1
Printed: June 1990

Table OF CONTENTS

Chapter 1: Overview

Chapter 2: Starting Tutor

Chapter 3: Image Processing with Tutor

Chapter 4: Particle Analysis with Tutor

Chapter 5: Tutor Menus

Chapter 6: The Tutor Learning Tool

Chapter 7: Adapting Tutor

Appendix A: Installing Tutor

Appendix B: Troubleshooting

Chapter 1

OVERVIEW

Introduction

Semper 6 *Plus* is an extension to the Semper high-level image processing language, which includes a number of new commands for user interface creation. Semper 6 *Plus* allows you to define your own user interface from a toolbox of screen icons such as panels, menu, cells and textfields. It enables you to define exactly how you interact with Semper – earlier versions of Semper only provided access through a command line. Figure 1-1 illustrates some of the user interface icons that are available with Semper 6 *Plus*. Note that all the icons are on a panel which is the basic building block of the user interface system.

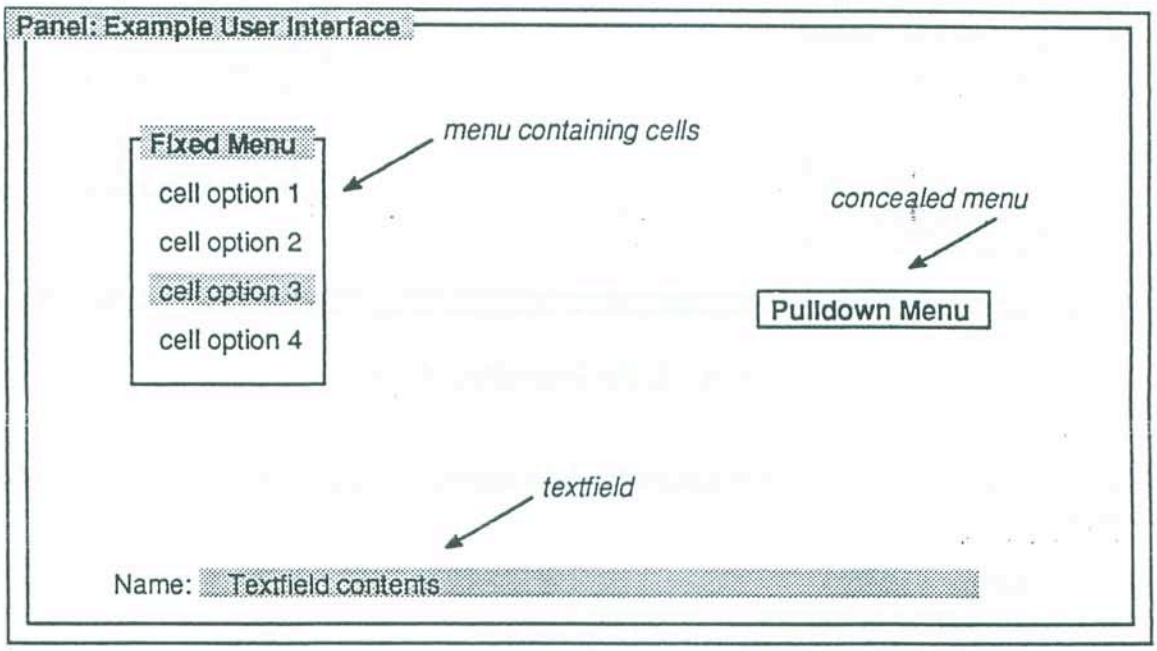


Figure 1-1. An Example User Interface

This manual describes the *Synoptics'* user interface, called *Tutor*, which is supplied with Semper 6 *Plus*. *Tutor* acts both as an example user interface and as a quick and easy way to access Semper if you prefer using a menu driven interface to typing in a series of commands.

Tutor is designed to provide:

- a structured introduction to using image processing software
- a learning tool to help you graduate to the use of the Semper command language
- an uncomplicated and user-friendly access route to Semper
- an example interface that you can modify

Figure 1-2 illustrates the *Tutor* menu interface:

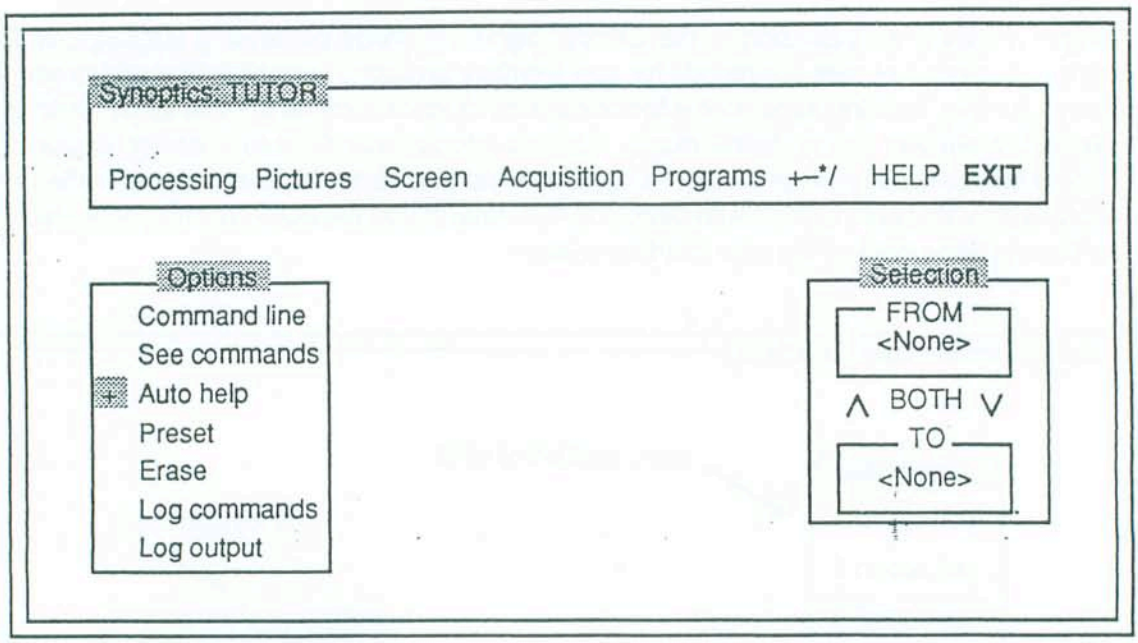


Figure 1-2. The Tutor Menu Interface

Tutor provides menu-based access to many of the commonly used image processing functions such as:

- particle analysis
- geometrical operations
- point by point operations
- spatial filtering
- fourier transforms

Context sensitive help is provided, which can be shown when a particular menu is displayed or when a help *button* is pressed. Access to all Semper's facilities is maintained by providing a command line interface to it. This allows you to try out programs or perform functions which are not currently in *Tutor*'s menus, that is, it lets you experiment with Semper. A logging feature is built into *Tutor*, this lets you record, in separate files, the output from Semper commands and the commands themselves. The command logging provides an easy method of developing your own Semper programs.

How this manual is organised

This manuals contains three tutorials, that provide step-by-step instructions on how to use *Tutor*. It also contains reference material, for example, menu layouts, for the *Tutor* interface. This manual is made up of seven chapters, which are described below:

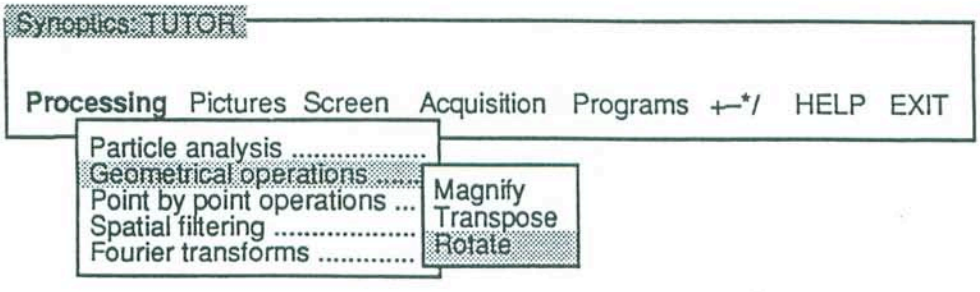
- *Chapter 2, StartingTutor*
  How to start *Tutor*, including a short tutorial to familiarise you with basic image processing operations.
- *Chapter 3, Image Processing with Tutor*
  A tutorial showing a real-life image processing application using *Tutor*.
- *Chapter 4, Particle Analysis with Tutor*
  A third tutorial showing you how to use Semper's powerful particle analysis features from *Tutor*.
- *Chapter 5, Tutor Menus*
  A blow-by-blow account of *Tutor* menus with diagrams of their layout and a list of the Semper commands that can be accessed from each menu.
- *Chapter 6, The Tutor Learning Tool*
  How to use *Tutor* to learn about the Semper image processing language, detailing *Tutor* help, command line access, command display and logging.
- *Chapter 7, Adapting Tutor*
  How to add new functions to *Tutor's* menus and some suggestions as to what to do next.

There are two appendices at the end of this manual, which are described below:

- *Appendix A, Installing Tutor*
  How to install *Tutor* if you have a non-standard configuration.
- *Appendix B, Troubleshooting*
  A list of *Tutor* error messages and suggested solutions.

Conventions used in this manual

Tutor menu icons are used to illustrate the instructions given in this manual. The relevant part of the menu is highlighted, together with its preceding menu so that you can follow the menu path. An example of a *Tutor* menu is given below:



Semper commands are shown in the following font:

```
rotate angle pi/3
```

unless they are embedded in the text, in which case they are shown in **bold**.

Computer output is shown in the following font, in bold:

```
Hello Tutor User
```

What else to read

For further details of the Semper command language, including a number of tutorials for the new user, read the following manual:

Beginners' User Guide

It is also useful to read this manual in order to gain an appreciation of some of the terms used in the Semper language.

If you would like to write your own application interfaces using the new Semper 6 *Plus* commands, consult the manual:

User Interface Guide

Both of these manuals are contained in the *Semper 6 Guide*.

A number of image processing operations are described in this manual. If you are new to the field of image processing, we recommend the following introductory text:

Digital Image Processing
by R.C. Gonzalez and Paul Wintz
published by Addison-Wesley, 1987
I.S.B.N. 0–201–11026–1

Chapter 2

STARTING TUTOR

Overview

This chapter describes how to start the demonstration user interface, *Tutor*, and explains what you see when you start it for the first time. It also contains a short tutorial to let you try out some basic image processing operations using *Tutor*.

Tutor start up

If you are the first person at your location to use Semper, please consult the appropriate *Semper 6 Installation Guide* for your machine.

Unless you are working on a PC, log onto your machine using your username and password or the username provided for Semper users. To start *Tutor*, change directory so that the Semper directory is current:

```
CD /SEMPER6P          (PCs)

cd /usr/semper        (UNIX workstations)
```

Note that you do not need to change to the Semper directory if you are working on a VAX installation. To run *Tutor*, type the following command:

```
semper /run=tutor
```

This causes Semper to start executing instructions from a file called *TUTOR.RUN* rather than reading from the standard *SEMPER.RUN* file (which is used when you start command-driven Semper by typing **semper**).

As Semper executes the *TUTOR.RUN* file various messages are printed on the display. Note that there is a difference between the first and subsequent start-ups of *Tutor*. The first time that you run *Tutor* it creates a library disk; the library disk contains a copy of the library programs (from the file *TUTOR.SPL*) in a format which can be accessed quickly by Semper. It also creates a file in which your images are stored, you'll be asked what size you want to make this disk but, by default, *Tutor* creates a file 1Mb in size. Finally, it rebuilds the *Tutor* user interface according to the programs

contained in the library disk and saves the interface to disk. Subsequently, when you run *Tutor*, there is no need for it to recreate these files. All *Tutor* has to do is to read in the stored user interface from the file *TUTOR.UIF*.

Some or all of the following messages are displayed on the screen at first start-up:

Device <*number*> assigned

A Semper 'device' has been assigned to this session. A 'device' may be a display area, a disk or magnetic tape.

Device <number> initialised with <number> slot directory

A new Semper disk (file) has been created (this may either contain images or programs).

Program '<name>' **inserted on device** <number>

The named library program has been added to the program disk. A library program contains sequences of Semper commands that you find convenient to use.

The disk in which you will store pictures does not exist yet. Please enter the size that you would like it to be in Mb (1,000,000 bytes). The disk should be at least 1Mb in size.

The *Tutor* picture disk is about to be created. If you plan doing a large amount of work with *Tutor* then it is a good idea to make the disk larger than the default size.

Unable to open file 'TUTOR'
Creating user interface.

A stored version of the user interface was not found, so a new one is generated.

help

If *Tutor* gives you a message other than these; refer to the explanation of *Tutor* error messages given in *Appendix B, Troubleshooting.*

If you start up *Tutor* and you don't have a mouse connected to your system, *Tutor* will automatically enable the **Command line** option. This option, which is explained in *Chapter 6, The Tutor Learning Tool*, lets you type in Semper 6 *Plus* commands in the conventional, line-orientated, manner. The automatic enabling of the command line is provided so that you can return to Semper 6 *Plus* easily: it is not possible to use the *Tutor* interface without a mouse.

What do you see?

If you start *Tutor* and are using a system with a separate terminal and display monitor (PC and VAX systems) the *Tutor* menu interface and help text is displayed on the terminal screen. The display monitor screen is blank for the display of images.

If you start *Tutor* and are using a workstation running a windowing system (VAX and Hewlett-Packard workstations, Sun or Silicon Graphics), three windows appear on the workstation screen:

- a Semper 6 *Plus* display window (for image display)
- a Semper 6 *Plus* menu window (for *Tutor* menus)
- a terminal window (for *Tutor* help and messages)

Note that you can move or re-size the workstation windows to suit your screen.

Figure 2-1 shows the initial *Tutor* screen, which contains the *Tutor* main menus.

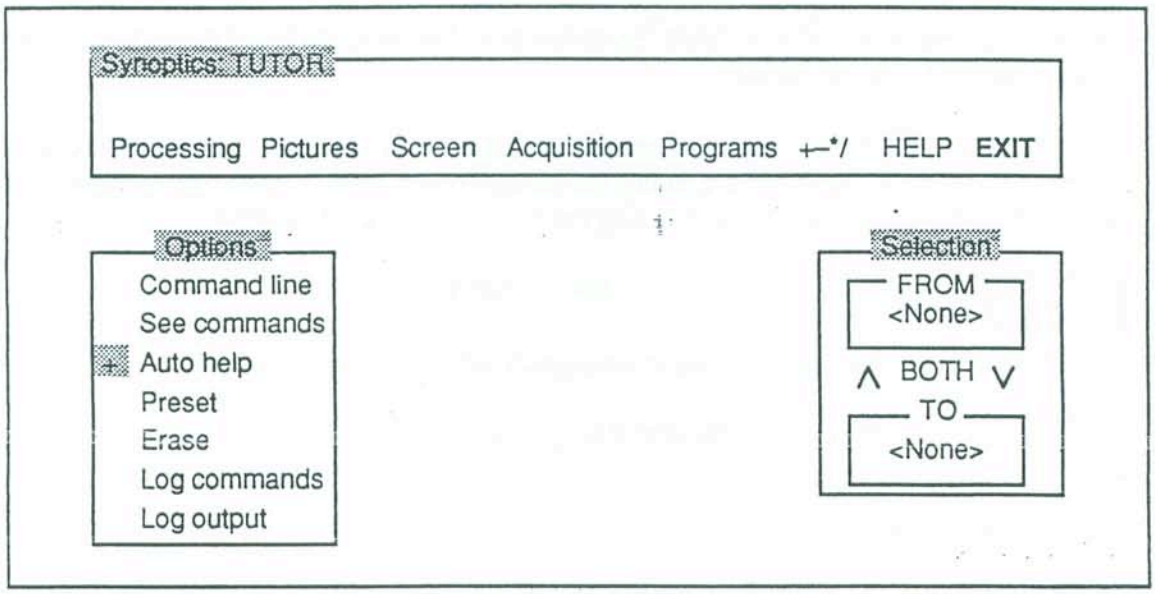


Figure 2-1. The Initial Tutor Screen

Figure 2-2 overleaf shows the *Tutor* help text. You will see this text below the *Tutor* panel, or in a separate text window if you are using a workstation.

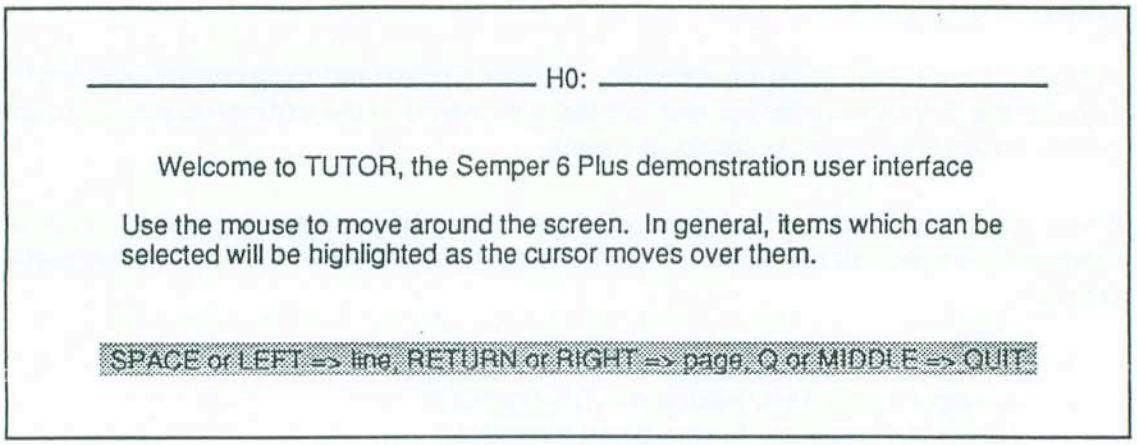


Figure 2-2. The Initial Tutor Help Text

Notice that there is a highlighted bar across the bottom of the screen or text window. This is *Tutor's* paging prompt and indicates that *Tutor* is waiting for you to press a key or click a mouse button before continuing to display some more text. Press and release the left hand mouse button (click the button). Some more text appears; this is *Tutor's* help text. When you have read through all the initial help the paging prompt disappears.

Next a small inverse block (or arrow if you are using a workstation) becomes visible – this is the cursor. As you move the mouse around, the cursor moves with it. Once the paging prompt disappears, you can start to use *Tutor*, using the mouse to select menu items.

Mouse Rules

To select a menu or option, click with the left mouse button.

To hide a menu, click the right mouse button.

A short tutorial

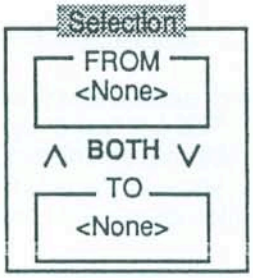
This tutorial gives you a brief introduction to the way that *Tutor* works. It shows you how to:

- Select a picture for display
- Display a picture on the screen
- Process the picture by sharpening it
- Magnify, transpose and rotate a picture

Tutor comes with a number of standard test images so that you can do some image processing without having to generate your own images. The first thing to do is to see what an image looks like. To do this you need to choose the image and the position on the screen where you would like it to be displayed.

Selecting an image

Move the cursor over to the area marked **Selection** and position it on top of the text **BOTH.**



As you do this, the letters **BOTH** are displayed in inverse video. This means that you can *select* this item. Click the left hand mouse button. After a short pause, as *Tutor* looks for images, the screen changes to the one similar to that shown in Figure 2-3 below.

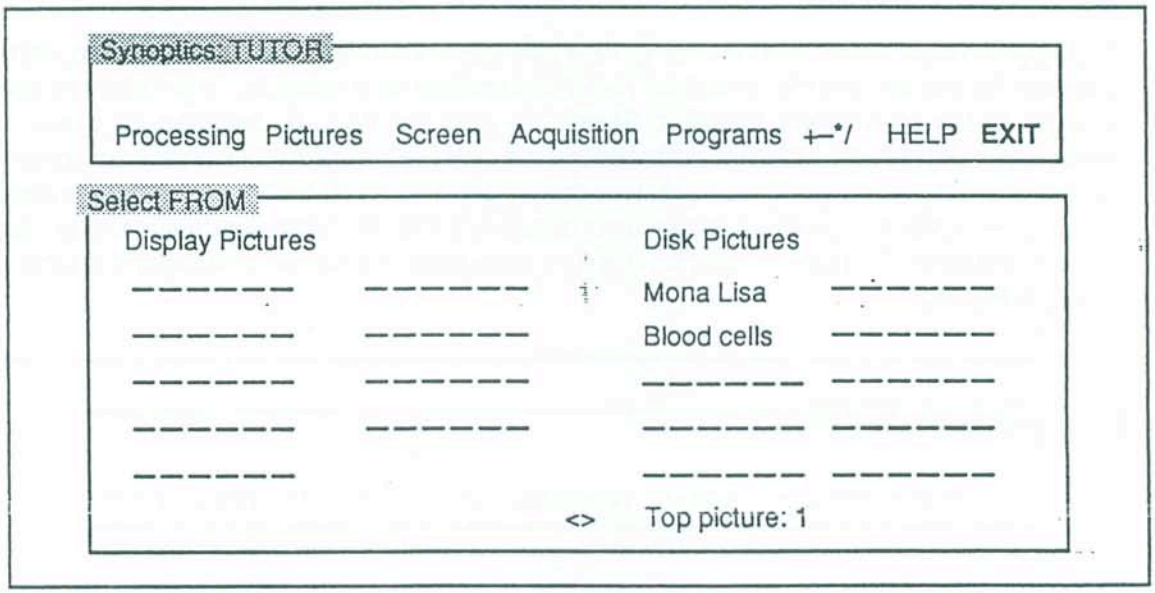


Figure 2-3. Tutor Picture Selection Menu Screen

Notice that *Tutor* displays some more help, explaining what this menu does, as shown in Figure 2-4 overleaf.

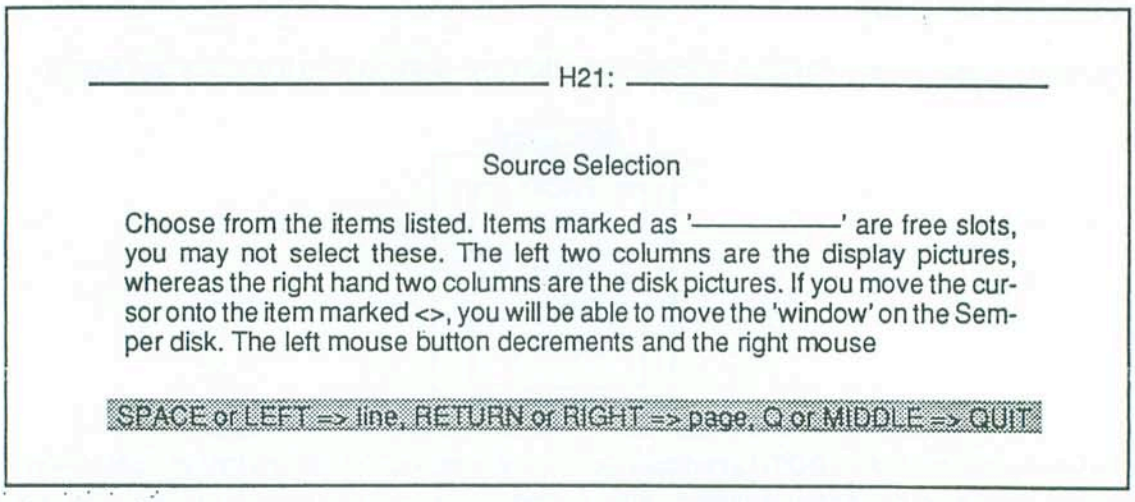


Figure 2-4. Tutor Picture Selection Help

Click the left mouse button until the paging prompt disappears. Move the cursor onto the text **Mona Lisa** and click the left hand mouse button. *Tutor* hides and then re-displays the large menu and also changes its title from **Select FROM** to **Select TO**. After the menu is displayed the cursor is positioned on **Mona Lisa**. Move the cursor so that it is over the top left line of dashes, immediately beneath the text **Display pictures**, so that the dashes are inverted. As before, *Tutor* displays more help, click the left button until the paging prompt disappears. Click the left mouse button to select the first line of dashes. The picture selection menu is hidden and the screen now looks like the one shown in Figure 2-5.

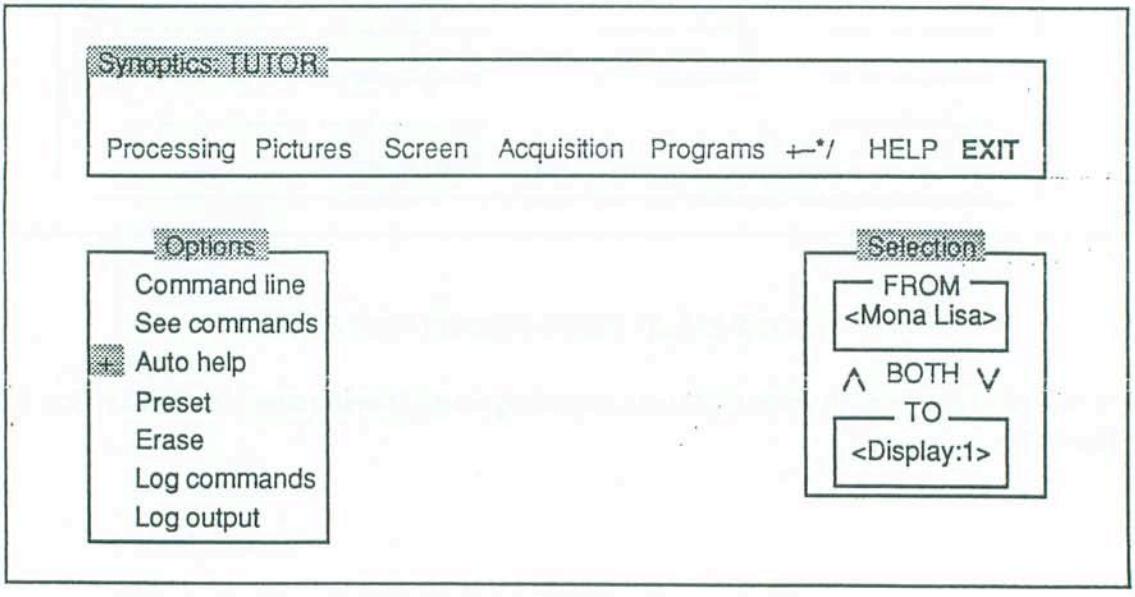
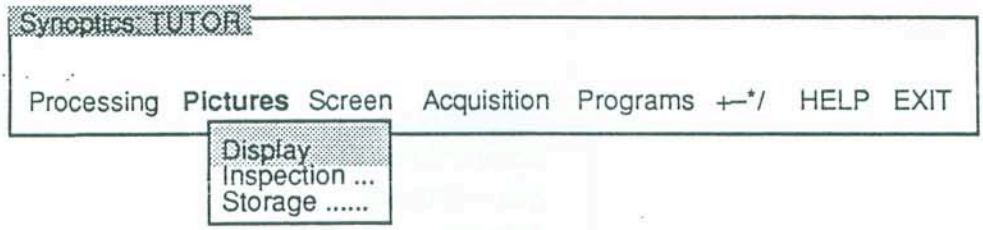


Figure 2-5. Tutor Screen After Picture Selection

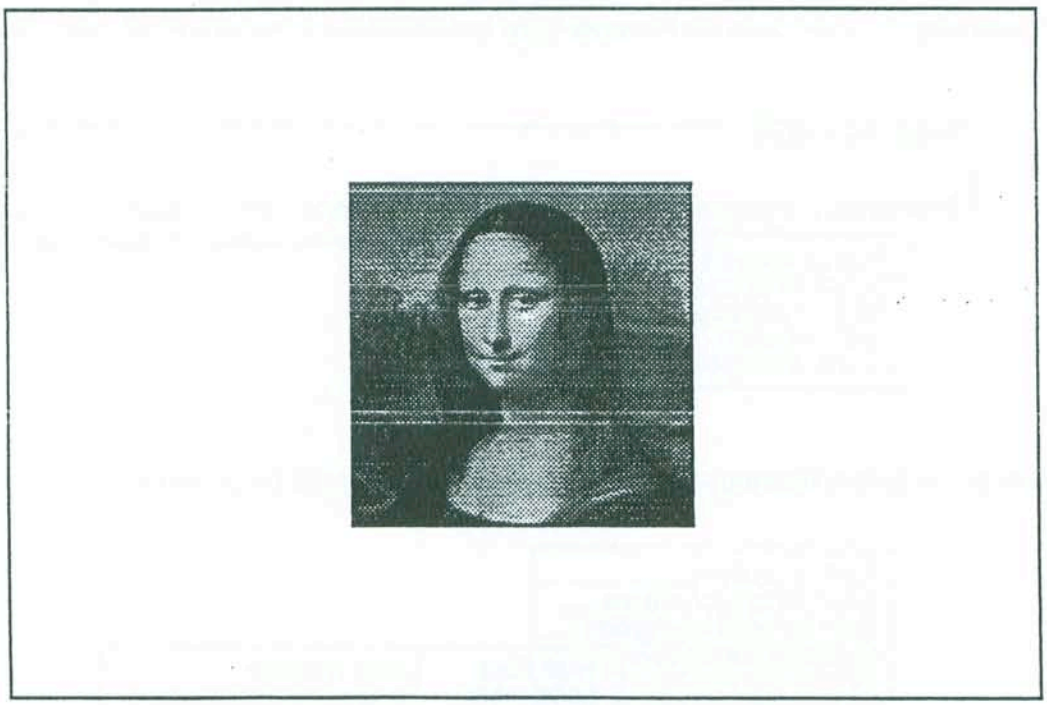
You have now selected where you would like the *Mona Lisa* to appear on the screen. *Tutor* defines a standard set of screen regions when it starts up and you have just chosen one of these. In fact, you can use another menu in *Tutor* to define these regions yourself, this is explained in the following chapter, *Chapter 3, Image Processing with Tutor.*

Displaying an Image

Now you have selected the picture you want to display and where you want to display it. All that remains is to display it. Move to the entry marked **Pictures** and click on it. A small menu appears with the following entries on it like this:



As you might expect by now, *Tutor* displays some help information about the new menu. When you are ready, click on the **Display** option. After a short delay, while *Tutor* scales the picture to suit the display, the picture appears, line by line, in the middle of your display screen or window, as is shown below:

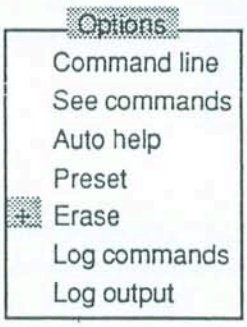


Sharpening an Image

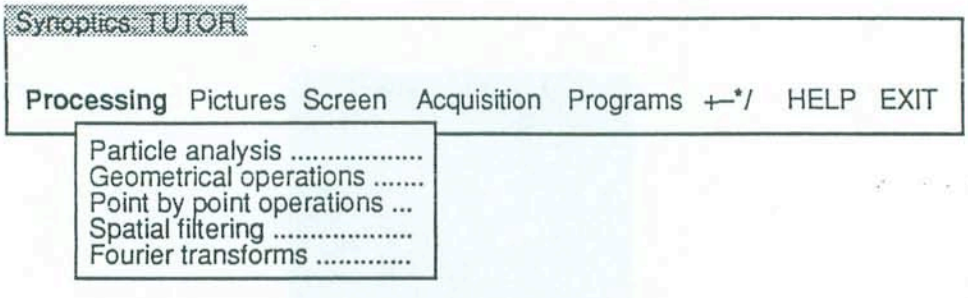
Before we begin to use image processing techniques on the test image, there are two preliminary steps to take:

1. Switch off the **Auto help** on the **Options** menu
2. Switch on the **Erase** option on the **Options** menu

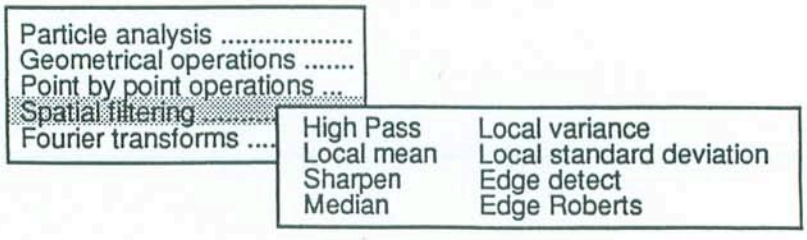
To do this, click the left mouse button on these options – this toggles the options on and off. Switching off **Auto help** saves times as **Auto help** is useful as a learning tool but slows you down when you are experimenting with *Tutor*. Switching the **Erase** option on ensures that you start with a clean display area each time you display a picture. The **Options** menu should now look like the one shown below:



Now you may want to process the image in some way perhaps sharpening it a little. Click on the text **Processing**, *Tutor* removes the **Pictures** menu and displays a new menu which looks like this:

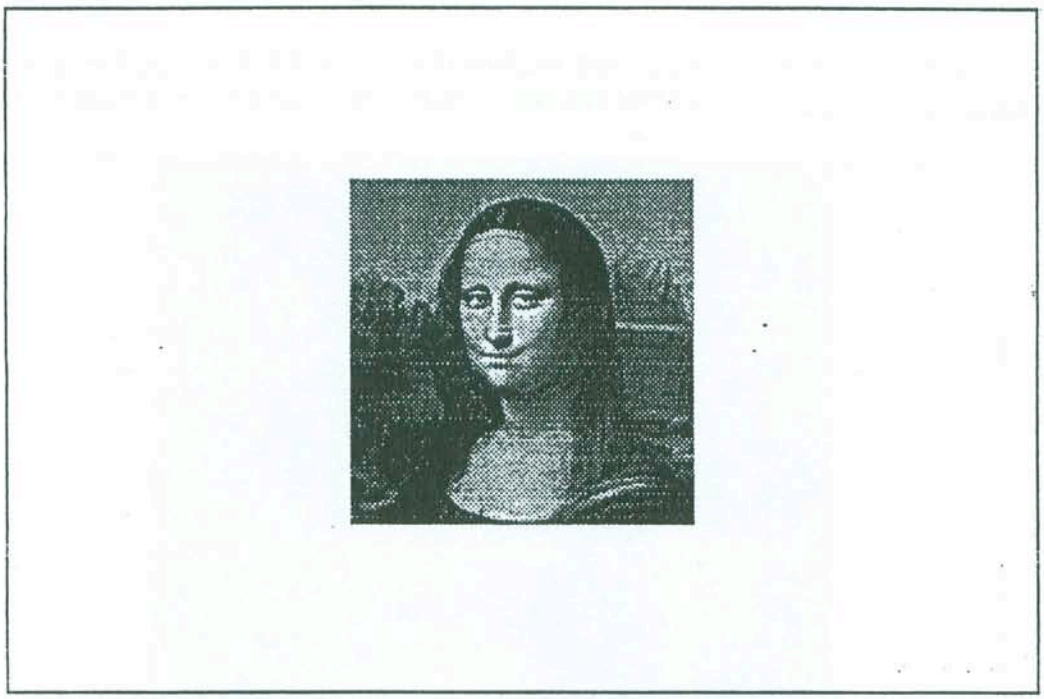


Click on the **Spatial filtering** option to display a further sub-menu on the screen:



Select the **Sharpen** option. A further menu appears which asks you for the size of the kernel over which this filter is to operate:

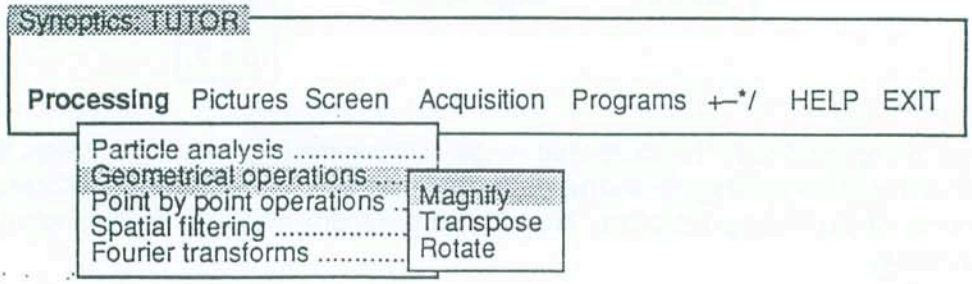
Click on the option **5 x 5**. The sharpened image replaces the original on the screen. Since you haven't changed the picture selection (using the **BOTH** option) *Tutor* still uses the picture on disk as the source for the filtering operation. The effect of the sharpening filter on the original image is shown below.



The effect of the **Sharpen** option is to double the high spatial frequencies in an image to emphasize the fine detail.

Magnifying an image

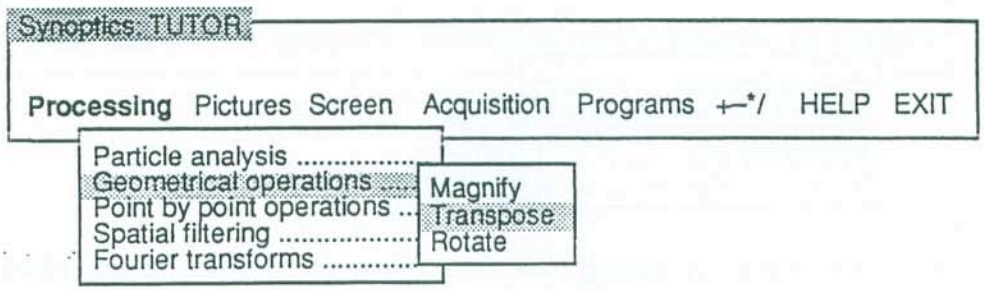
Tutor provides a simple way to magnify an image. Display the **Processing** menu and the **Geometrical operations** sub-menu. Select the **Magnify** option from this sub-menu.



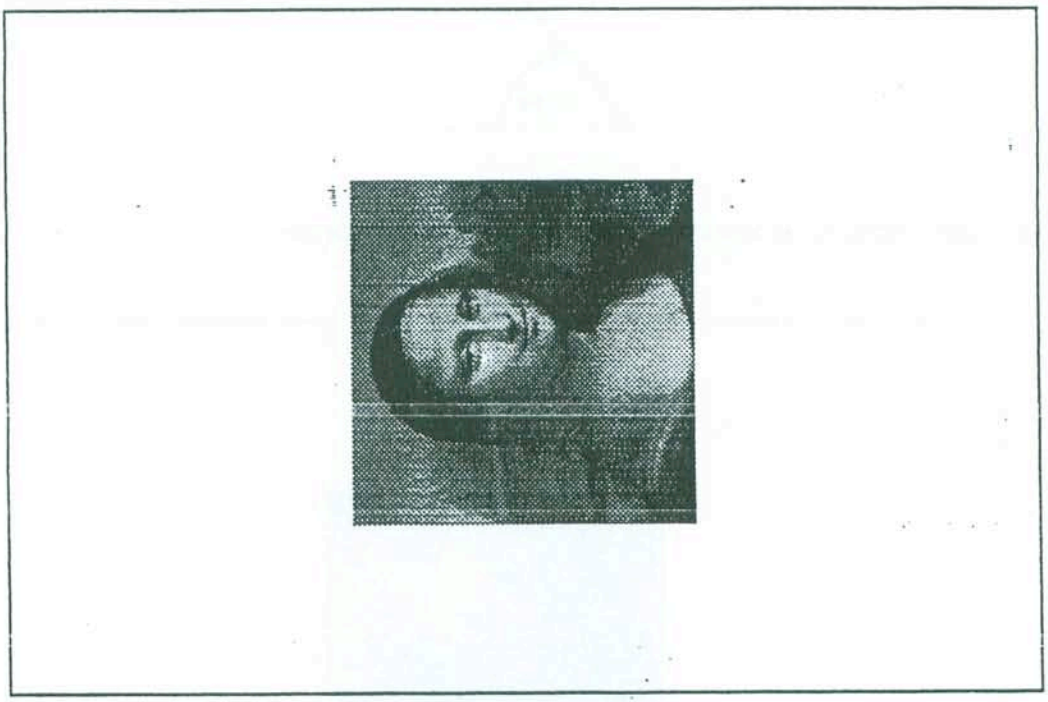
Your image is re-displayed on the screen, magnified by a factor of 2, as is illustrated below. Note that it is only the image on the display that has changed. The image on disk remains the same.

Transposing an Image

Tutor allows you to transpose a picture, that is, to interchange the *x* and *y* axes so that they are reflected in the top left/bottom right diagonal. To see the effect on the test picture, display the **Processing** menu and the **Geometrical operations** sub-menu and select the **Transpose** option.

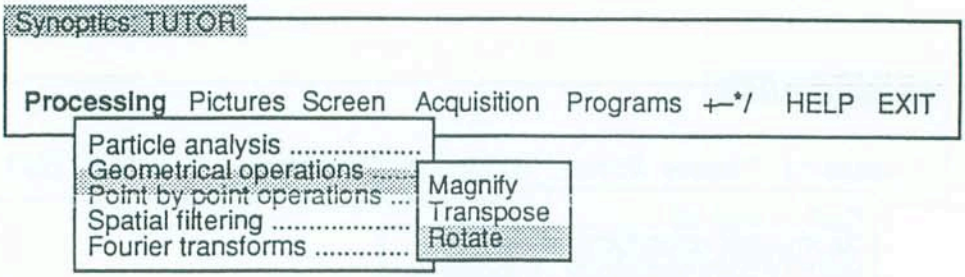


The transposed image is shown below.

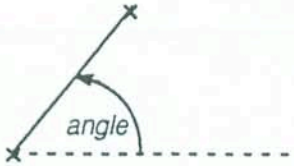


Rotating an Image

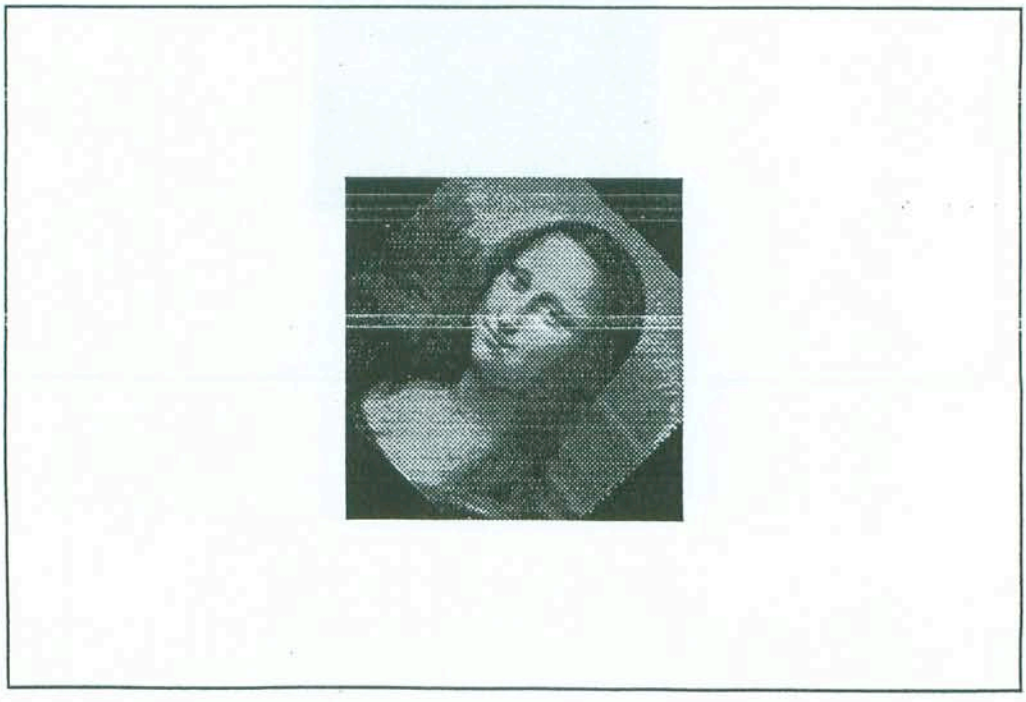
As a final exercise, you may like to rotate the test image. Display the **Processing** menu and the **Geometrical operations** sub-menu. Select the **Rotate** option.



Define an angle for rotation by drawing a line on the display picture, using the cursor (click the left mouse button to start the line and again to end the line). The angle from the horizontal is used as the angle of rotation. Note that *Tutor* only accepts angles between 30° and 150°.



The effect of rotating the test image by 45 degrees is shown below.



Leaving Tutor

To stop *Tutor* and return to the operating system, select the **EXIT** option from the main menu bar.

A new panel appears on the screen which asks you to confirm your decision, as shown in Figure 2-6 below:

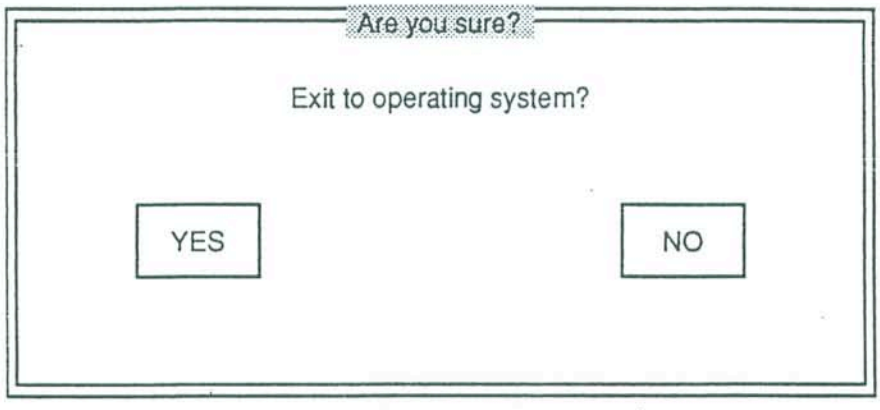


Figure 2-6. The Tutor Exit Panel

To leave *Tutor*, move the mouse to highlight the **YES** option and click the left mouse button.

What next?

The following two chapters contain further tutorials which introduce you to *Tutor's* screen layout, image processing and particle analysis features. If you are new to *Tutor* we recommend that you work through these tutorials. If you are a more experienced user, skip to *Chapter 5, Tutor Menus.*

Chapter 3

IMAGE PROCESSING WITH TUTOR

Overview

This chapter introduces some basic image processing features that you can access from *Tutor's* menus. It describes how to:

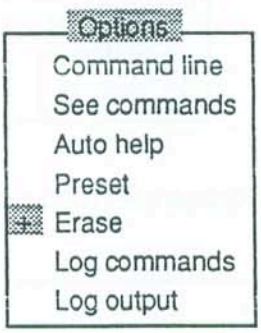
- divide the display into separate areas
- use a filtering technique on the test image
- change the viewing conditions
- perform an edge detection operation on the test image

Defining separate areas of the display

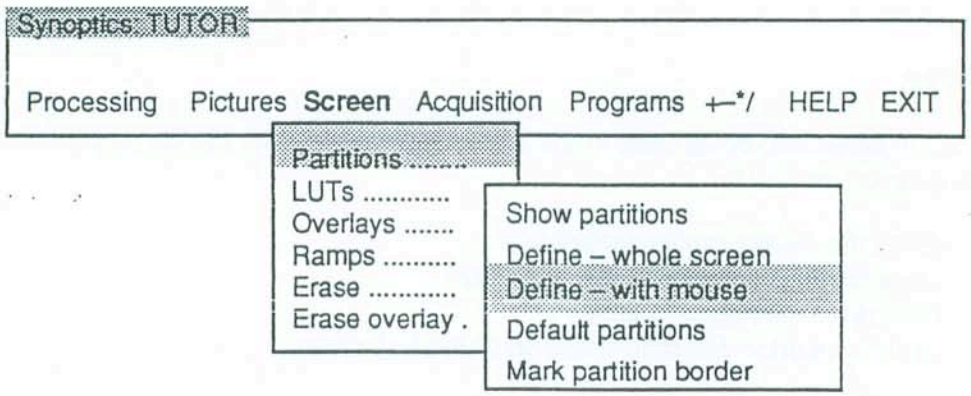
Start *Tutor* by typing the command:

```
semper /run=tutor
```

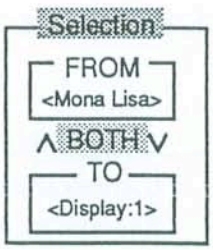
As a preliminary step, ensure that the **Auto-help** option is off and the **Erase** option is switched on, on the **Options** menu.



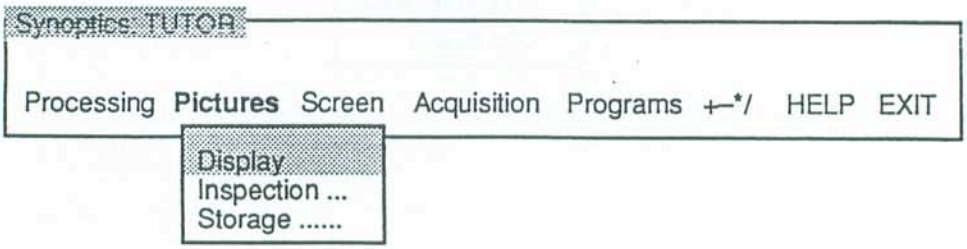
The next step involves defining two areas of the display, called *partitions*, to split up the display or display window horizontally. Select the **Screen** menu and the **Partitions** sub-menu and select the **Define – with mouse** option and the **Partition 1** option. A cross-hair cursor appears on the display. Click the left mouse button once to define one corner of the partition. You will see a rubber-band effect on the display which illustrates the potential area of the partition. Click the left mouse button again to define the opposite corner of the partition. Go through the same procedure to define **Partition 2.**



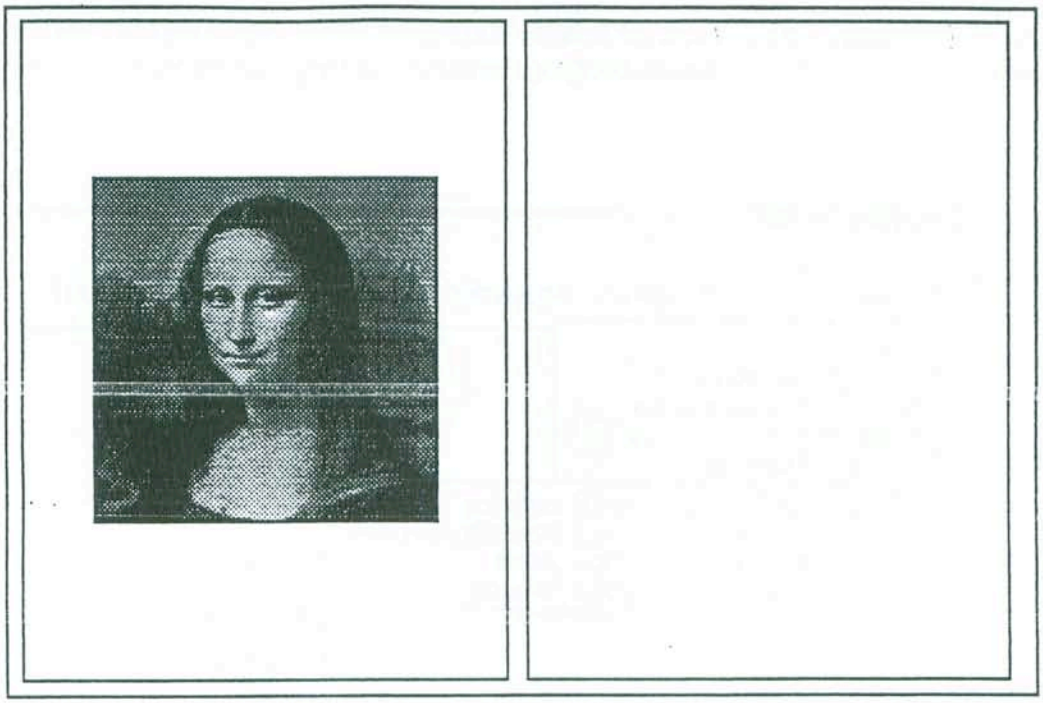
Display the first test picture in partition 1. To do this, select **BOTH** from the picture **Selection** menu and choose *Mona Lisa* as the source picture (**FROM**) and the first line of dashes under **Display Pictures**, *Display:1*, as the destination (**TO**).



To see the test picture, select the **Pictures** menu and click on **Display**.



Your display should now look similar to the one shown overleaf. The two partitions are defined so that you can display the results of image processing on an image in partition 2, whilst displaying the original image as a reference copy in partition 1.

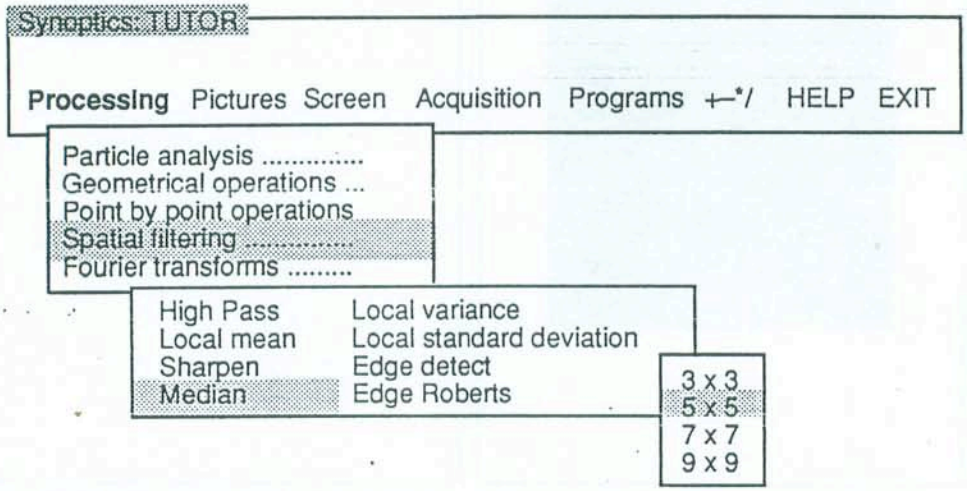


Performing a median filter

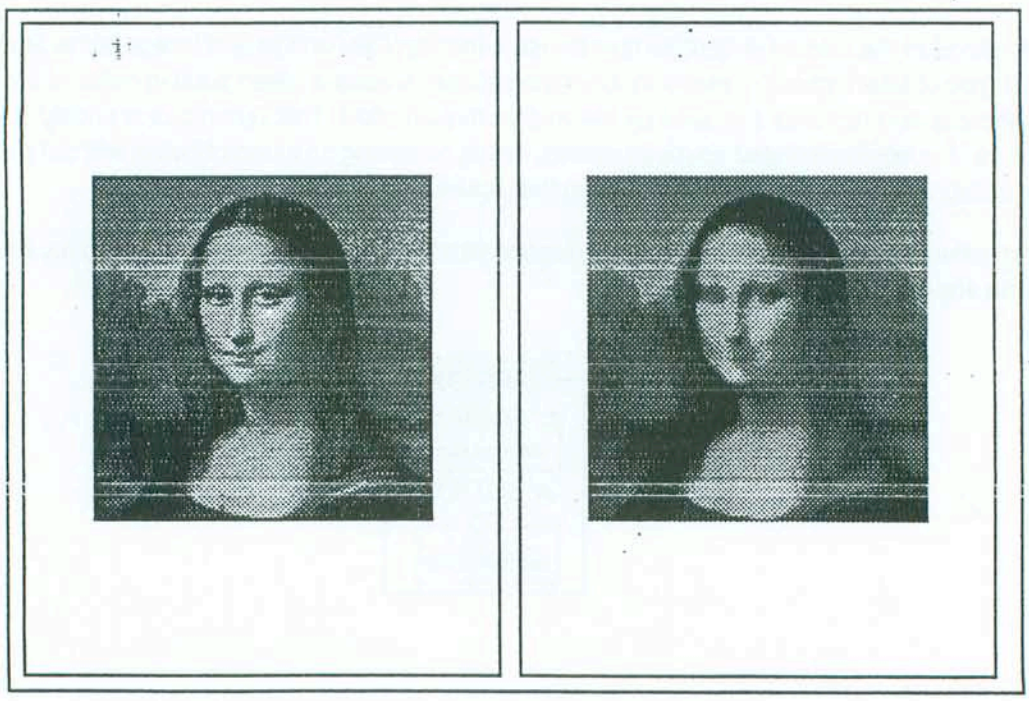
This stage of the tutorial describes how to use a median filter on the test image of the *Mona Lisa*. This type of filter ranks the pixels in a neighbourhood around a given pixel in order of increasing brightness and replaces the pixel by the middle ranked pixel. This technique is widely used as a means of removing isolated spurious values, that is, removing one kind of noise, without degrading the sharpness of the lines or the edges in the picture.

First, select partition 2 (*Display:2*) as the display area for the output of the filter, using the **Selection** menu and the **TO** option.

Display the **Processing** menu and the **Spatial filtering** sub-menu. Select the **Median** option and choose **5 x 5** as the filtering kernel (specifying the number of neighbouring pixels to be ranked).



The effect of using a median filter on the image is shown below:

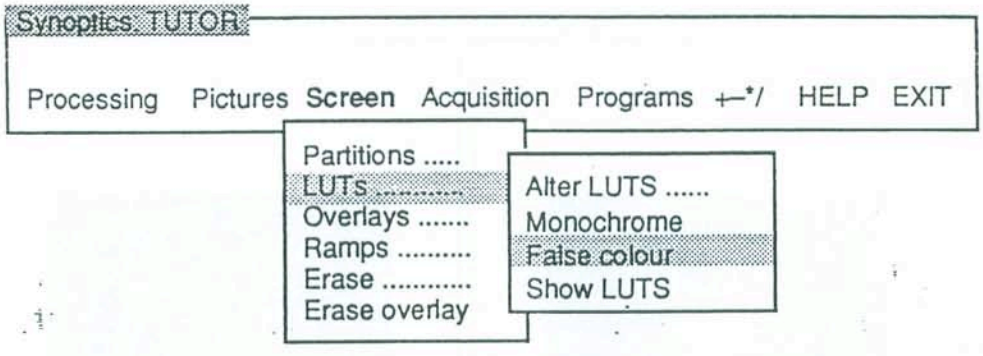


Changing the viewing conditions

Tutor allows you to change the viewing conditions of the display from the **Screen** menu. In this instance, we will use a false colour look-up table to change the display colours to emphasize the next image processing operation. Note that the human eye is more sensitive to variations in colour than gradations in grey-scale levels. Accordingly, *Tutor* has look-up tables which allow you to map digital data in a variety of ways:

- monochrome
- false colour
- full colour (true colour systems only)

To change to a false colour look-up table, display the **Screen** menu and the **LUTs** sub-menu. Select the **False colour** option.

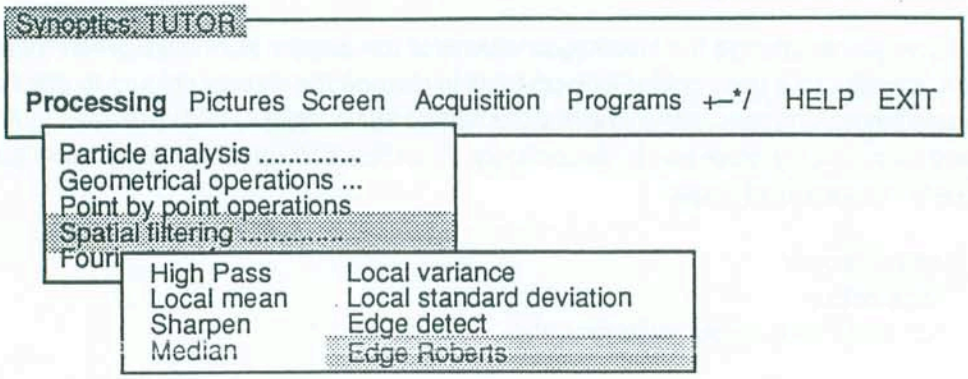


Performing an edge detection operation

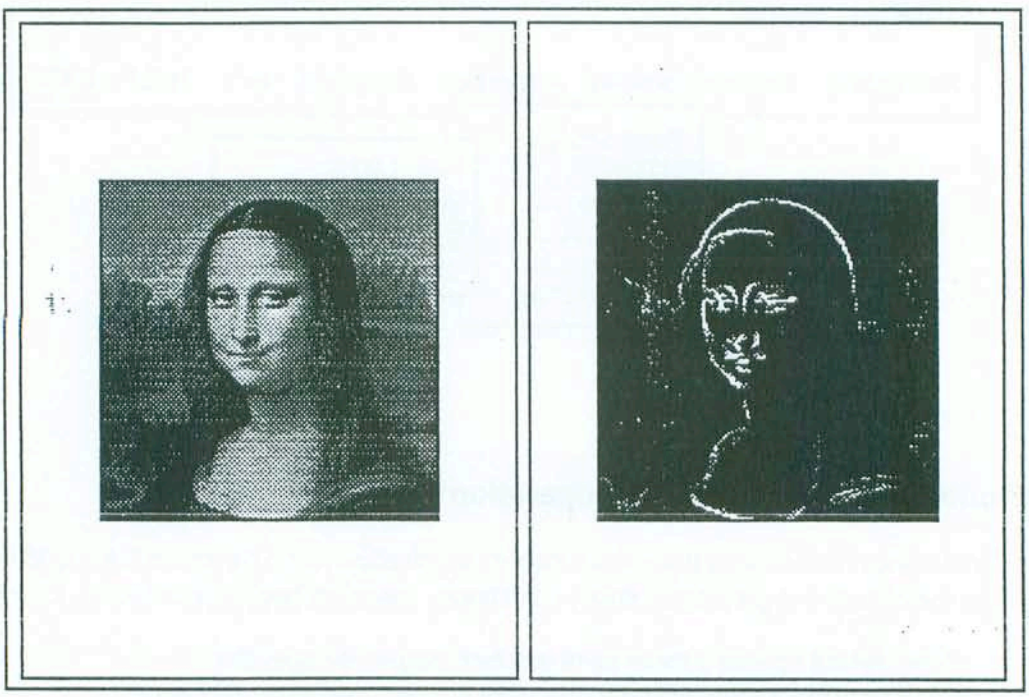
The *Tutor* menus contain edge detection functions which allow you to delineate the boundaries and highlight the changes in grey-scale level in an image. There are two edge detection functions:

- **Edge detect** applies a three point gradient magnitude operator
- **Edge Roberts** applies a larger four point absolute diagonal difference

In this instance, we shall apply the **Edge Roberts** operator on the test picture. To do this, select the **Processing** menu and the **Spatial filtering** sub-menu. Select the **Edge Roberts** option, as is shown overleaf.



The effect of the edge detection operation on the test image is shown below:



To leave *Tutor*, select the **EXIT** option from the main menu bar.

What next?

You may like to experiment with other features on *Tutor's* processing menus before progressing to the particle analysis tutorial in the next chapter. For example:

- **Point-by-point operations**, which allows you to perform arithmetical operations on images
- **Fourier transforms**, which allows you to create the transform and power spectrum of an image
- Other features of the **Spatial filtering** menu, for example, **Local mean** and **Local variance** filters

Chapter 4

PARTICLE ANALYSIS WITH TUTOR

Overview

This chapter introduces you to the particle analysis features of *Tutor*. Many image processing operations require differentiation and analysis of particles and *Tutor* is provided with a number of features to deal with this type of application.

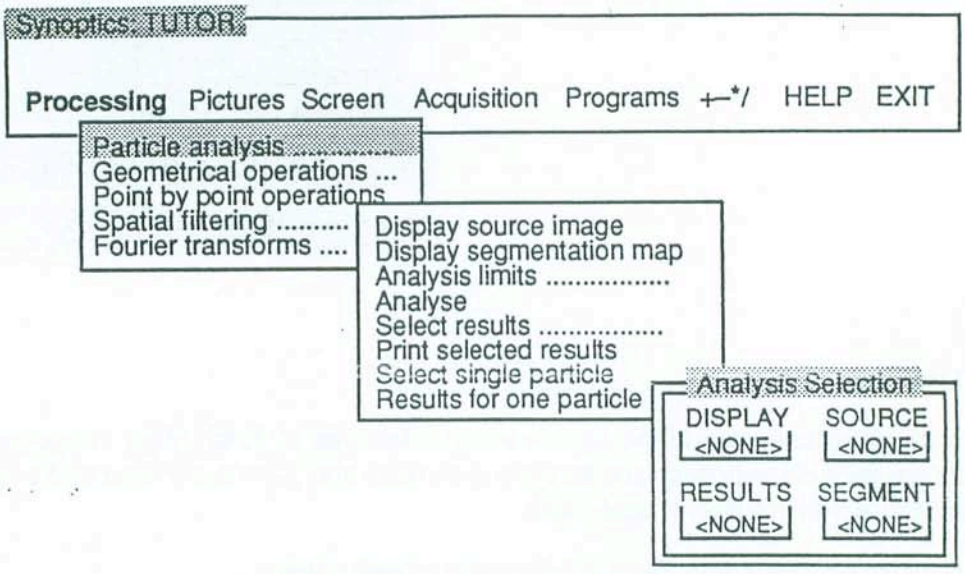
A typical particle analysis task might consist of the following steps:

1. Defining the source and output pictures for the analysis
2. Setting up a picture for analysis
3. Analysing a picture
4. Interpreting the analysis

Defining the source and output

As a preliminary step, start *Tutor* and switch the **Erase** option on and the **Auto help** option off on the **Options** menu.

Select the **Particle Analysis** option from the **Processing** menu. This displays an analysis sub-menu and a menu that lets you specify the display area and pictures used in the analysis procedure, as is illustrated overleaf.



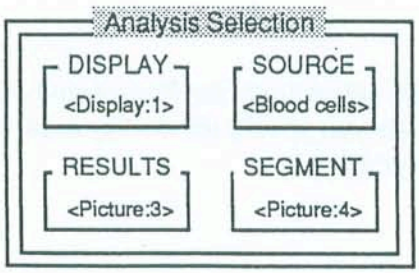
When you are performing particle analysis, picture selection takes a different form from that usually used in *Tutor*. This is because the analysis procedure produces two new forms of output:

- a segmentation map which is a visual representation of the analysed particles
- a particle parameter list (*ppl*) which contains measurements of the individual particles

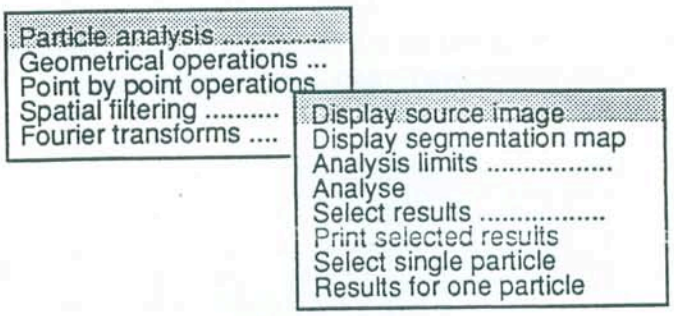
To select the source and output pictures for the analysis, follow the steps given below:

1. Select the **DISPLAY** option from the **Analysis Selection** menu and select the first display picture (the first line of dashes under *Display Pictures)*.
2. Select the **SOURCE** option and select the disk picture called *Blood cells*.
3. Select the **RESULTS** option and select disk picture 3 (the first line of dashes under *Blood cells*).
4. Select the **SEGMENT** option and select disk picture 4.

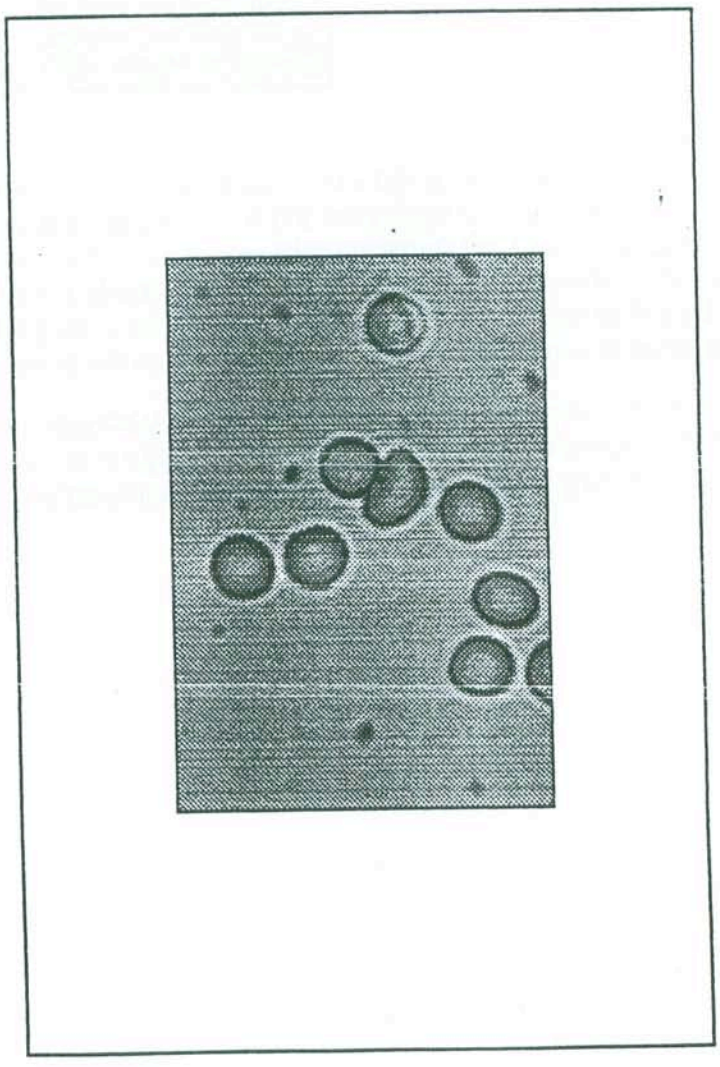
The **Analysis Selection** panel should now look like the one given below:



Next, select the **Display source image** option from the **Particle Analysis** sub-menu. An image showing blood cell particles appears on the screen:



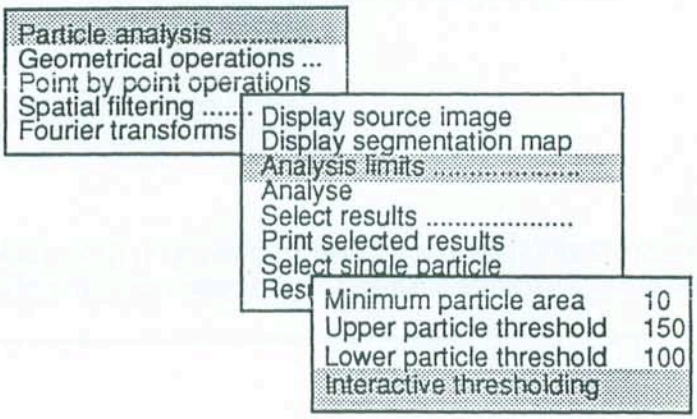
This image will be used as a source for the analysis procedure. *Tutor* is able to count the number of particles in a picture and measure a number of parameters for each particle.



Setting up a picture for analysis.

The second stage of the analysis procedure consists of defining the thresholds for analysis, using the display look-up table to highlight all pixels that you require *Tutor* to treat as particles.

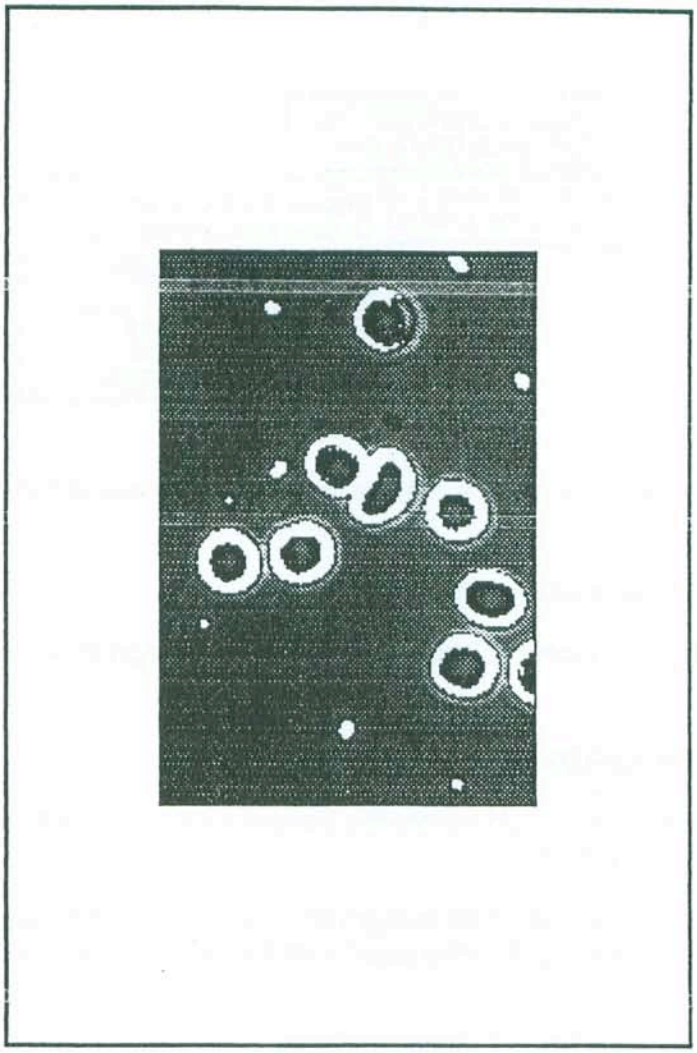
Select the **Analysis limits** option on the analysis sub-menu. You will see a further sub-menu. Select the option called **Interactive thresholding** from this menu.



The display look-up table will change and a slider appears at the bottom of the display or display window, which shows the range of highlighted grey levels. Move the mouse to adjust the look-up table so that only the particles of interest are coloured. At the end of the adjustment your image should look similar to the one shown overleaf. Click the left mouse button when you are satisfied with the display highlighting. *Tutor* sets the threshold limits, that is the greyscale range that will be used in the analysis, to correspond to the levels highlighted on the display.

Note that you can also specify an area threshold, using the **Minimum particle area** option. The other options on the **Analysis limits** menu allow you to specify numerical values for the greyscale threshold, instead of using the mouse to supply these values interactively.

The image below shows the effect of adjusting the look-up table to highlight the blood cell particles.



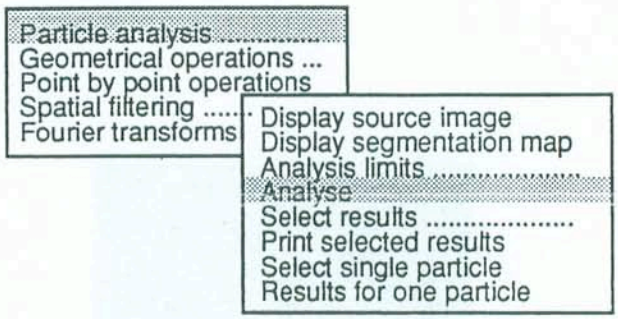
The first slider shown below illustrates the range of greyscale levels. The second slider shows the highlighted band used as a threshold for analysis.

(1)

(2)

Analysing the picture

To analyse the picture, select the **Analyse** option from the **Particle Analysis** sub-menu.



Tutor reports the number of particles that are found during the analysis and displays a message similar to the following:

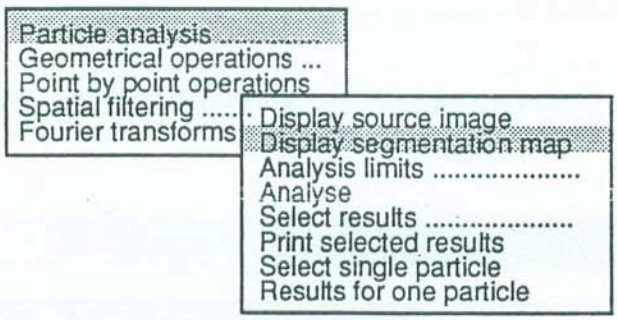
```
Number of particles found: 16
```

Note that your result may differ from the one given above if the greyscale thresholds differ in any respect.

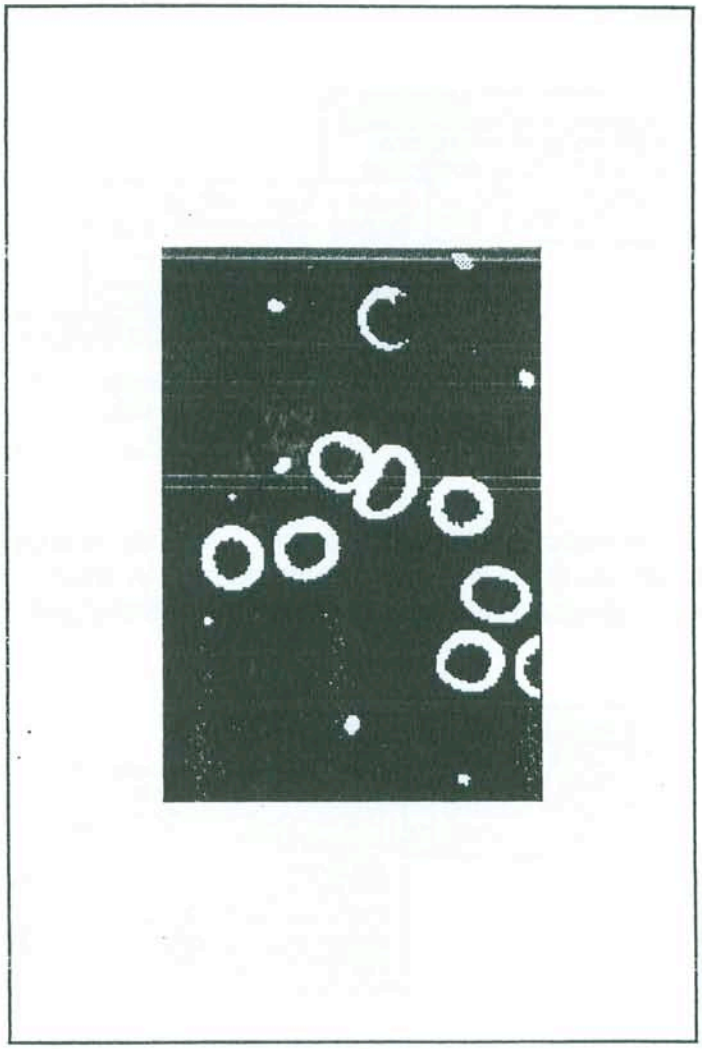
Interpreting the analysis

As was mentioned earlier, *Tutor* provides two types of output from the analysis; a segmentation map and a particle parameter list.

The segmentation map shows which particles were found during the analysis. To see this map, select the **Display segmentation map** option from the **Particle Analysis** sub-menu.

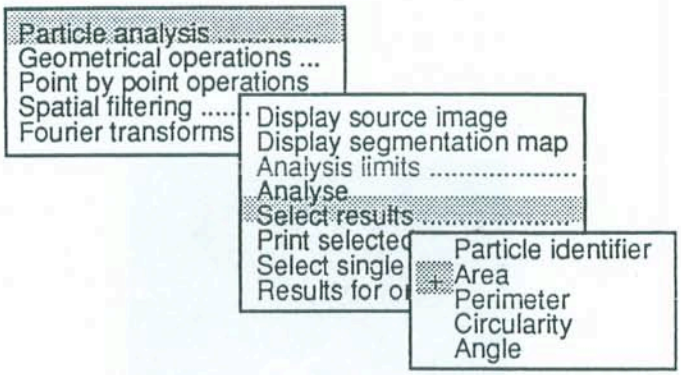


From this map we can see that the analysis procedure has found the blood cells particles and also the small particles in the background, as is shown below.

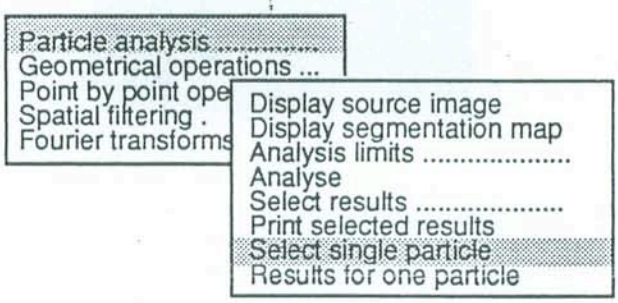


Tutor also produces a list of particle parameters which are stored in **RESULTS** picture 3. This picture stores a number of parameters found for each particle during the analysis, including the particle identifier, area, perimeter, circularity and angle.

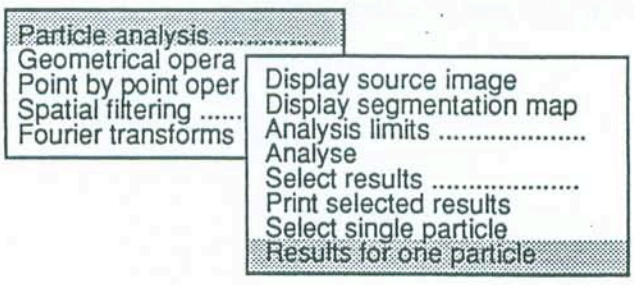
To display the results of the analysis, first click on the **Select results** option from the **Particle Analysis** sub-menu. A new menu appears on the screen or window, showing the analysis parameters. Select the **Area** option from this menu. A plus sign appears to the left of the parameter to show that it is selected.



Secondly, click on the **Select single particle** option. *Tutor* displays a cross-hair cursor on the display which you can use to select a small background particle. When you click the left mouse button, *Tutor* uses the display look-up table to highlight the individual particle.



Finally, if you select the **Results for one particle** option, *Tutor* displays a table showing the area for the highlighted particle.

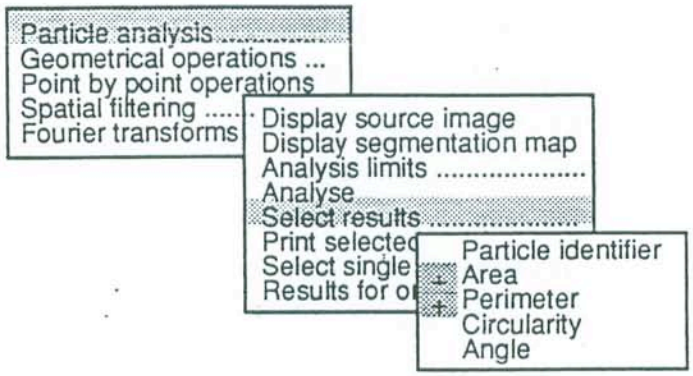


This information is retrieved from the particle parameter list stored in picture 3. You will see text on the screen or text window, similar to that given below:

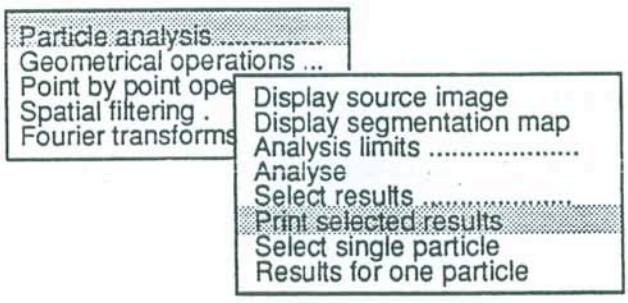
```
Particle id: 1
Index    Area
  1    186.00
```

Note that this information could be used to perform a more selective analysis on the test picture. It is possible to change the minimum area threshold from 10 to 250, for example, and analyse the image again. This would have the effect of eliminating the small background particles from the analysis results.

It is also possible to display the parameters for all the particles found during the analysis. To do this, click on the **Select results** option and select the **Perimeter** option. (You can also select any other parameter of interest).



Next, select the **Print selected results** option. *Tutor* displays a table showing the selected parameters for all particles.



The parameter result table is shown below.

Index	Perimeter	Area
1	43.882	136.0
2	35.194	95.0
3	207.471	556.0
4	39.059	115.0
5	50.675	101.0
6	20.353	31.0
7	906.395	2078.0
8	251.976	1217.0
9	377.971	1208.0
10	374.864	1186.0
11	538.919	879.0
12	20.890	35.0
13	298.962	1074.0
14	178.461	475.0
15	40.685	128.0
16	27.194	53.0

What next?

Tutor's particle analysis facilities are limited, but access to the Semper command line allows you to perform a number of analysis functions which do not appear on *Tutor's* menus. For example, Semper allows you to annotate and edit the segmentation map and to print 25 parameters for each particle. See *Chapter 6, The Tutor Learning Tool* for details of the Semper command line and refer to the manual:

Quick Reference List

for details of Semper's particle analysis commands.

Chapter 5

TUTOR MENUS

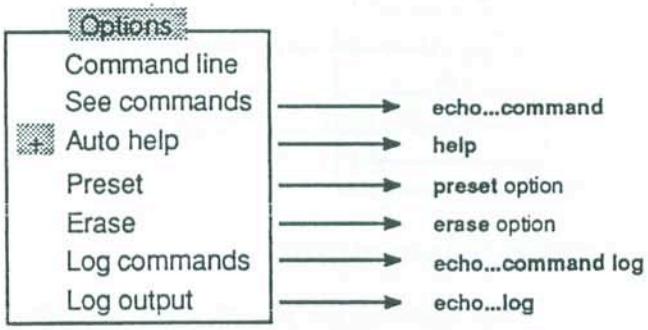
Overview

This chapter describes the functions that are contained in *Tutor*'s menus. The first two sections list the functions of the **Options** panel and the **Selection** panel. The third section describes the *Tutor* main menu bar.

The sections are broken down, on a menu by menu basis, until all the menus are explained. Each menu description is preceded by a diagram illustrating the menu and listing the Semper commands accessed by the menu. The descriptions do not cover any extensions that may be particular to one machine or, of course, any extensions that you might write. Some of the descriptions suggest other Semper commands which may be appropriate or useful if you develop your own programs.

The Options panel

The **Options** panel contains seven items as shown below:



To the left of one or more of these you may see a plus sign ('+') this indicates that the particular option is active. The options are described below:

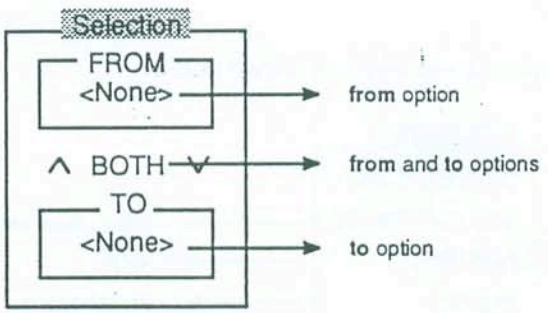
- **Command line** This option draws a field on the screen that allows you to enter conventional Semper commands.
- **See commands** This option draws a field on the screen that shows the commands that are used when you select a menu option. Note that you cannot select this option if you have already selected the **Log commands** option.
- **Auto help** This option causes help information to be displayed whenever you select a new menu. If this option is de-selected, the same help is available by clicking on the **HELP** button.

- **Preset** — This option sets the Semper variable *preset*. Semper uses the *preset* variable to decide whether or not to scale any picture according to the range of values in the picture. If *preset* is set, the values of the Semper variables *min* and *max* are used to scale the picture; values outside this range being set to *min* or *max* appropriately.
- **Erase** — This option sets the Semper variable *erase*. If *erase* is set then Semper clears the region of the screen in which a picture is to be shown, before displaying a picture.
- **Log commands** — This option causes *Tutor* to start recording the commands that you execute in a file called *TUTORC.LOG*.
- **Log output** — This option causes *Tutor* to start recording its output, including the help text, in a file called *TUTORL.LOG*.

For further detail of the **Command line**, **See commands**, **Auto help**, **Log commands** and **Log output** options refer to *Chapter 6, The Tutor Learning Tool.*

The Selection panel

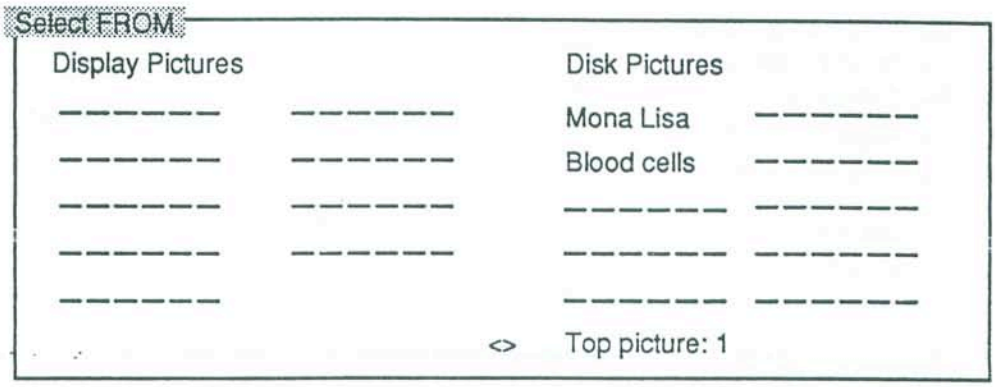
Tutor allows you to select source and output pictures from a **Selection** panel:



The selection panel is composed of several parts:

- **FROM** — This option allows you to select a source picture for your image processing operations.
- **TO** — This option allows you to select a destination picture for your image processing operations.
- **BOTH** — This option combines the functionality of both the **FROM** and **TO** buttons. You select the source picture and then the destination picture.
- ∧ — Copy up arrow. Makes the source picture the same as the destination picture.
- ∨ — Copy down arrow. Makes the destination picture the same as the source picture.

Selecting the **FROM, TO** or **BOTH** options displays the following panel:



The menu is given the title **Select FROM** for source picture selection or **Select TO** for destination picture selection. Note that there are two types of entry: one consists of a dashed line which indicates that a disk or display picture does not exist for a particular slot and the other gives the title of an existing picture. To select a source picture, click on a picture's name. To select a destination, click on a picture name or a dashed line, which effectively creates a new picture at the chosen slot.

Although Semper is capable of storing almost a thousand pictures on disk, the menu only shows ten pictures at any one time. You can move around to a different part of the disk using the following option:

<>

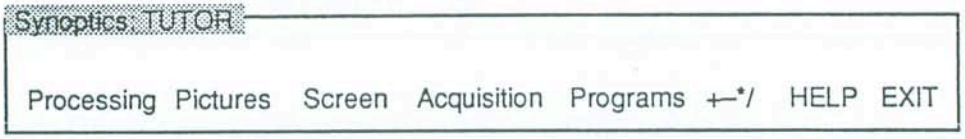
As you move the cursor over this option, a small inverse panel appears:

Click the left mouse button to add ten to the value of the panel. Click the right mouse button to subtract ten. It is not possible to produce a value of less than 1 or greater than 990. If you move the cursor outside the button (**<>**) *Tutor* recreates, if necessary, the menu showing the new window onto the Semper picture disk.

The main menu bar

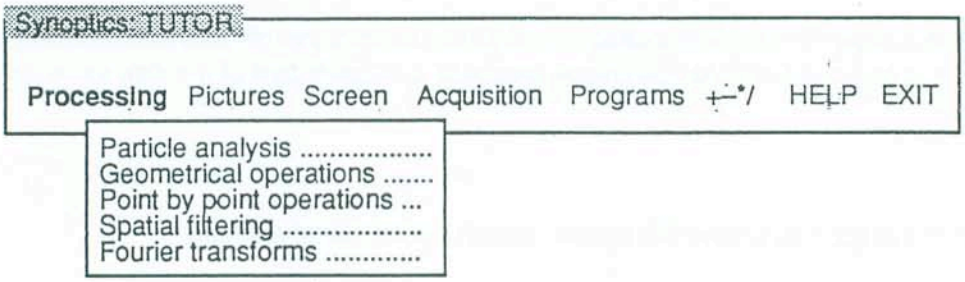
The main menu bar of the *Tutor* user interface consists of the following menus and functions:

- the **Processing** menu
- the **Pictures** menu
- the **Screen** menu
- the **Acquisition** menu
- the **Programs** menu
- the **Calculator**
- the **HELP** function
- the **EXIT** function



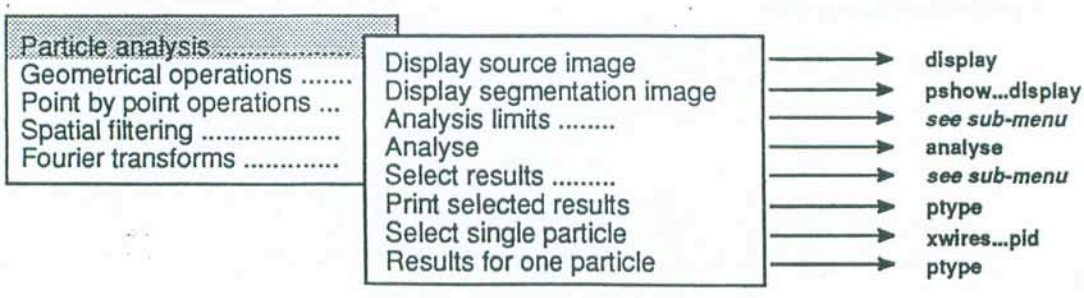
These menus and functions are described in order in the following sections.

The Processing menu

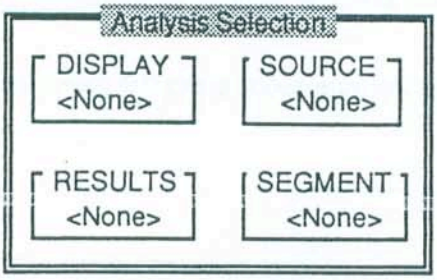


This menu covers some of Semper's image processing functions. Only a sub-set of Semper's commands are contained in this set of menus but they are sufficient to cover many of the commonly used image processing functions. The processing functions include particle analysis, geometrical operations, filtering, image arithmetic on a point by point basis and *Fourier* transforms.

The Particle analysis sub-menu



The **Particle analysis** sub-menu lets you use some of the particle analysis commands that are built into Semper, as shown in the diagram above. When you activate this menu, *Tutor* changes the picture selection panel to the one shown below. The conventional selection panel returns when you hide the **Particle analysis** menu.



A new picture selection panel is shown because the analysis of an image produces two output pictures specified by **RESULTS** and **SEGMENT**.

- The **RESULTS** picture is a particle parameter list which describes the characteristics of each particle that is found.
- The **SEGMENT** picture is a binary segmented image in which the original pixels are replaced by the particle number, with background pixels (that is, areas that do not contain a particle) set to zero.

The **RESULTS** image is, in fact, not a conventional image and it is not possible to display it directly. To see the particle parameters, use the **Print selected results** and the **Results for one particle** options which are described overleaf.

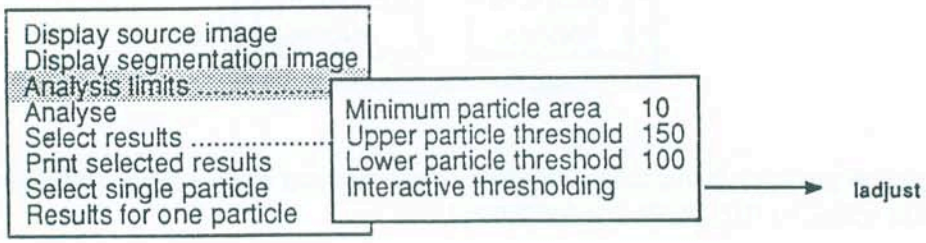
You can display the segmented image using the **Display segmentation image** option. This option changes the current look-up table to one that is appropriate to particle display.

The **DISPLAY** and **SOURCE** options on the **Analysis selection** panel allow you to specify the display picture and source picture respectively.

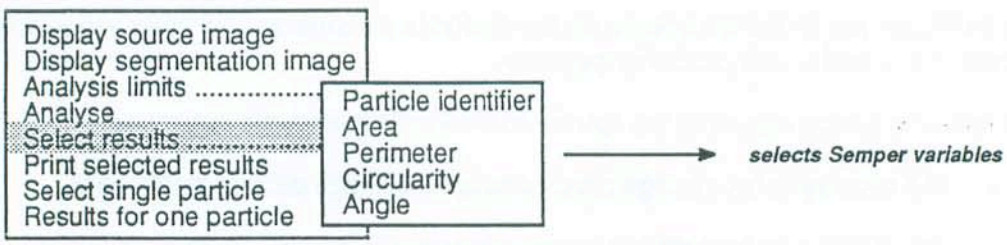
The following section describes the options on the **Particle analysis** sub-menu:

- The **Display source image** option displays the image defined in **SOURCE**.
- The **Display segmentation image** displays the segmentation map produced by the analysis, using a special look-up table to highlight the particles.
- The **Analysis limits** option pops up a sub-menu which lets you alter how the image is to be analysed. This sub-menu is described below.
- The **Analyse** option performs the actual analysis of an image.
- The **Select results** option displays a sub-menu that allows you to select the information that is displayed when you ask to see the results of a particle analysis. This sub-menu is described later.

- The **Print selected results** option prints out the parameters that you specified in the **Select results** sub-menu for all particles. If you did not select any parameters, no results are sent to the terminal.
- The **Select single particle** option lets you identify a particular particle from the segmented image so that you can print out some of its characteristics. Use the mouse to mark the particle of interest.
- The **Results for one particle** option prints the parameters for the particle you selected using the above option.

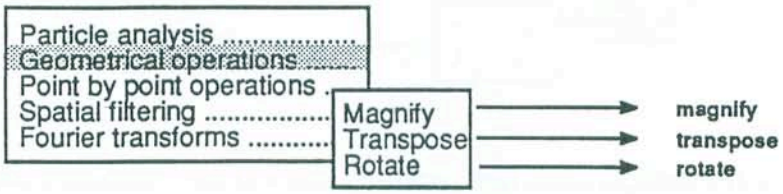


The **Analysis limits** sub-menu lets you change the thresholds for analysis. If you select any of the first three items by clicking on them, a calculator pops up and you can enter a specific numerical value for the entry. This is the same calculator that is accessible from the main menu via the +_*/ button. If you choose the **Interactive thresholding** option, a small ramp is displayed at the bottom left of the display screen. You can then use the mouse to move the lower and upper limits of the (green) highlighting band. Moving the mouse horizontally alters the upper limit and moving it vertically adjusts the lower limit of the highlighting band. When you are satisfied with the highlighted range click the left mouse button. The lower and upper particle thresholds are set according to the limits of the highlight bands. Whenever you alter any of the characteristics on this sub-menu, it is hidden and re-displayed to reflect the latest parameters.



The **Select results** sub-menu allows you to select some of the parameters stored by *Tutor* in the particle parameter list. Only five of a possible twenty-five parameters results for each particle can be selected from this sub-menu. The selected parameters are displayed when you use the **Print selected results** and **Results for one particle** options from the **Particle Analysis** sub-menu. Note that Semper calculates other parameters which are not shown on the menu, these are; *feret diameters, x and y limits, x and y position, centre of area, vertical and horizontal projections* and their values can be printed using the **ptype** command.

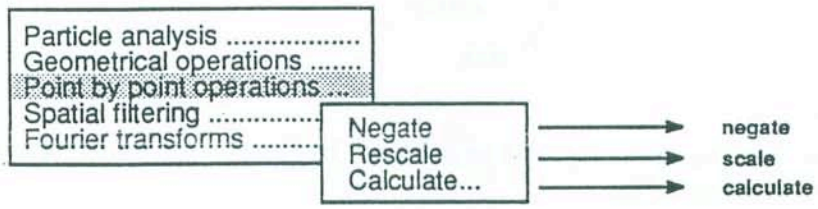
The Geometrical operations sub-menu



The **Geometrical operations** menu is a sub-menu of the **Processing** menu and is concerned with performing geometrical operations on images. The **Magnify** option magnifies an image by two; in fact the Semper command supports magnification by any integral factor.

Both the **Transpose** and **Rotate** menu options require that the image is both square and factorisable, Semper's **show sizes** command gives details of these factorisable sizes. The **Transpose** option transposes a picture (like matrix transposition). The **Rotate** option turns a picture through an angle (30 .. 150 degrees). When you rotate an image, *Tutor* asks you to draw a line with the mouse. The angle that this makes with the horizontal is the angle through which the image is rotated. Semper will display an error message if you give too small or too large an angle. Note that the Semper **extract** command allows you to rotate an image by an **arbitrary** angle.

The Point-by-Point operations sub-menu



The above menu is concerned with **Point by point operations** on pictures:

- The **Negate** option rescales a picture, interchanging its minimum and maximum values. This affects every pixel value in the image. Note that if you would like to *view* the image without affecting its contents, you can use the **negated** option to the Semper command **display**.

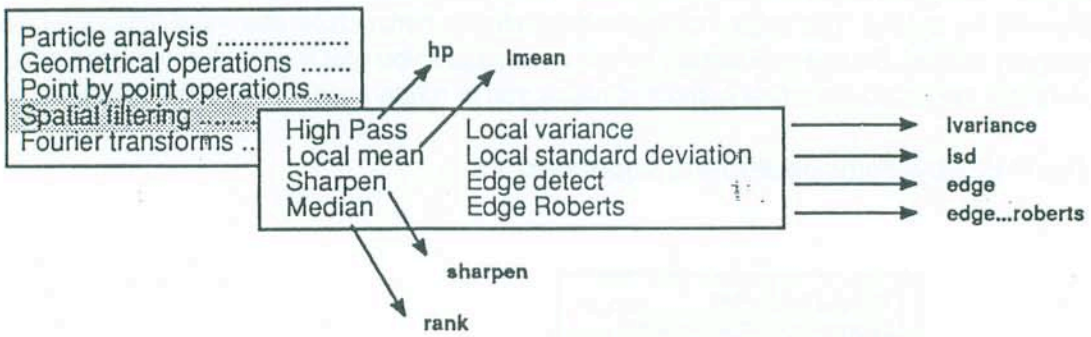
- The **Rescale** option alters the scale of an image to be in the range 0 to 255. The underlying Semper command, **scale**, allows you to specify any linear rescaling that you may require. In fact, you can use **scale** to perform the same function as the **Negate** option.

- The **Calculate** option allows you to perform simple arithmetic between pictures.

In fact, the Semper **calculate** command lets you use completely arbitrary arithmetic expressions on pictures. These include functions like sine, tangent and logarithm as well as the more conventional addition, subtraction and so on. *Tutor*'s **Calculate** option makes it easy to calculate the difference between images or the ratio of images. For a list of Semper functions and arithmetical operators, refer to *Appendix B, Semper Expression Syntax* in the following manual:

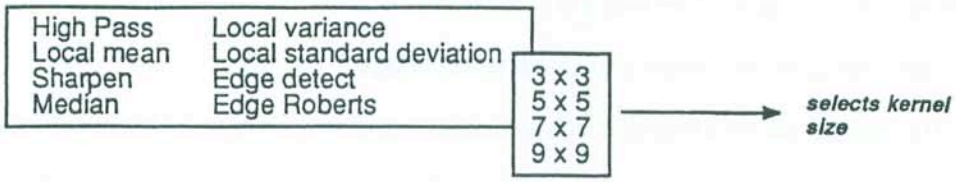
Semper 6 Command Reference Part 2

The Spatial filtering sub-menu



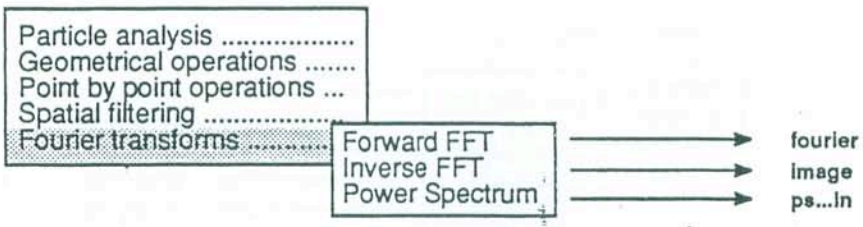
The **Spatial filtering** menu contains eight filters. The size of the kernel, over which the filter operates, is described below.

- The **High Pass** filter levels the local background of the picture, keeping the rapidly varying detail.
- The **Local mean** filter replaces each source pixel by the local mean over a square block and gives you an idea of the general brightness around a pixel.
- The **Sharpen** filter adds the difference between the original picture and a locally averaged version back to the original, making the fine detail stand out more.
- The **Median** filter removes isolated line and point objects on non-binary images, replacing pixels by the middle ranked pixel of a small region.
- The **Local variance** filter is similar to the local mean filter except that the pixel is replaced by the variance rather than the mean.
- The **Local standard deviation** filter replaces the pixel with the standard deviation rather than the mean or variance.
- The **Edge detect** filter finds edges on the basis of a three point gradient vector.
- The **Edge Roberts** filter uses a four point gradient vector to find any edges.



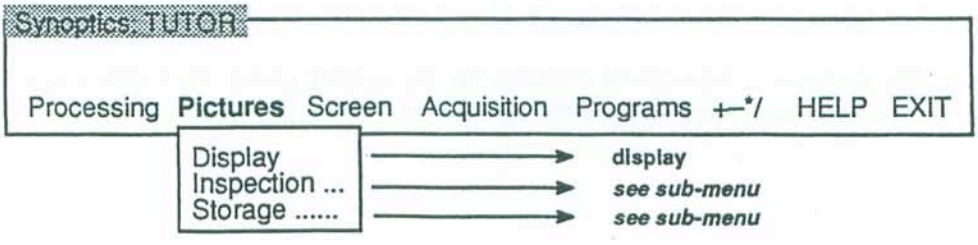
The above menu appears when you wish to filter an image. Not all of the filters can support all these different sized kernels. In some cases it is not useful to use all of the kernel sizes (for example, smaller kernel sizes are more suitable for the **Sharpen** filter). In some cases too, other filter sizes are supported by the underlying Semper command but are not given in the menu.

The Fourier transform sub-menu



The **Fourier transform** sub-menu lets you transform and back-transform images to and from *Fourier* space. The **Power spectrum** option forms the log *Power Spectrum* of a transformed image so that the wide dynamic range of the picture can be accommodated. This is a form suitable for sending to the display. Not shown, but present in the Semper language, are commands for filtering the transformed images (for example, the command **weight**). Any pictures to be transformed must have dimensions which are a power of two. If your image does not meet this criterion, you can use the **create** command to make an image which is the next power of two greater than your image size and **paste** your image into the picture. If you create a new image, use the **value 0** key to give the newly created picture a value of zero; this will, effectively, form a border to the pasted image.

The Pictures menu

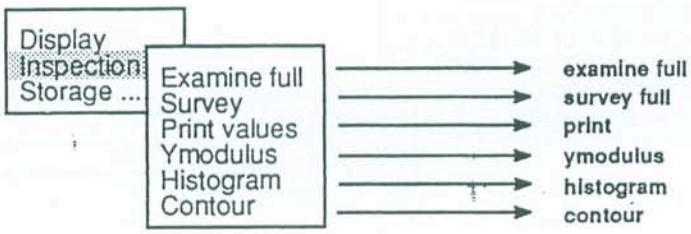


The **Pictures** menu is concerned with the manipulation of pictures:

- The **Display** option allows you to display an image.
- The **Inspection** menu lets you look at the information in an image in various different ways.
- The **Storage** menu allows you to rename pictures, copy them and also to output them as a *PostScript* file in a form suitable for dumping directly to a printer.

The **Display** option lets you show an image on the display device using, as the source the picture given by the **FROM** button and, as the destination the region given by the **TO** button (which must be a display picture/partition). Semper automatically scales the picture when it copies it in the display so that it makes the best use of the display's capabilities (in terms of the number of grey levels, for example). If you have set the **Preset** option on the **Options** panel, Semper uses the existing values of the variables *min* and *max*, not the values contained in the picture.

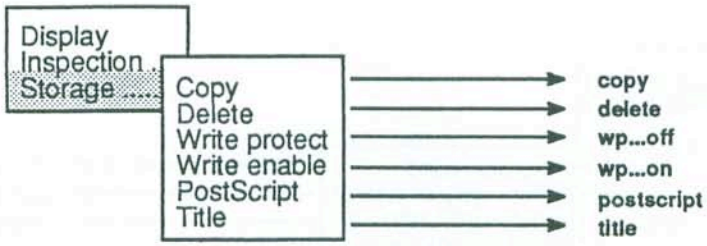
The Inspection sub-menu



The **Inspection** menu lets you find out about a picture and also produce a 3-D view of an image:

- The **Examine full** option describes the size and name of an image as well as the currently recorded range of the data in the image. On a display picture you will see an entry *sampling*, the value of this depends upon the size of the picture and the display partition; Semper automatically sub-samples a picture so that it will fit on the display.
- The **Survey** option reports the range of data within a picture.
- The **Print values** option prints picture values centred about the origin; in fact Semper lets you print out the values of an arbitrary rectangular region by specifying the appropriate keys and options.
- The **Ymodulus** option draws a wireframe perspective view of the image.
- The **Histogram** option produces a histogram of the entire image.
- The **Contour** option draws contours on the overlay plane. Note that another Semper command, **spc**, lets you draw the contours on the image directly.

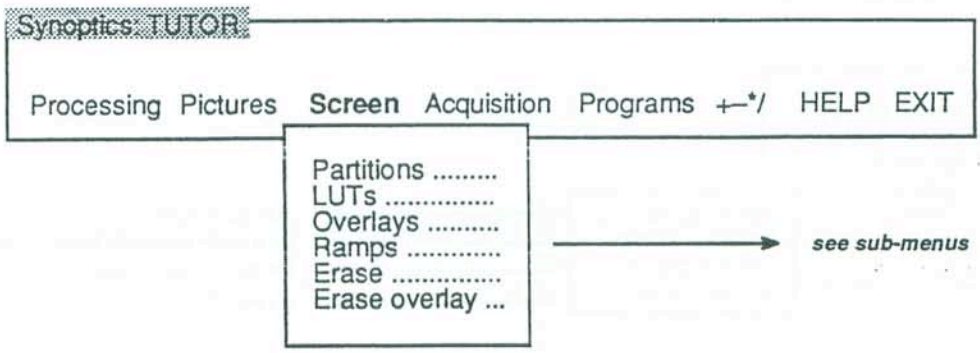
The Storage sub-menu



The **Storage** menu allows you to use some of Semper's storage functions. This includes copying pictures, in fact you can copy whole groups of pictures using the Semper **copy** command. Semper also allows you to write protect pictures so that they cannot be modified, which is useful if you have any valuable pictures.

You can use the **Title** option to change the name of an picture: *Tutor* only lets you give pictures titles up to ten characters long but Semper itself allows titles of about 100 characters. You type the new name of the image into a panel which appears approximately in the middle of the screen. If you are using the **See commands** option you will see that in the command line the title is given as *<Name>* rather than the title you type. This is because of the way in which Semper operates. The **PostScript** option lets you output a picture to a file, with the filename specified in the same way as for the **Title** option. You can copy these files to a *PostScript* printer to produce hardcopy.

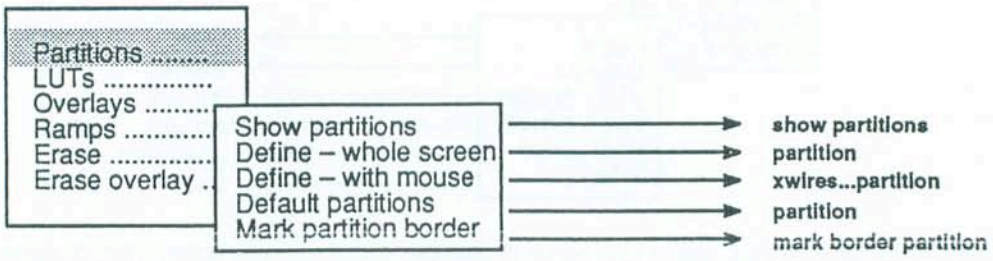
The Screen menu



The **Screen** menu is concerned with functions that manipulate the display.

- **Partitions** are Semper's name for regions of the display screen.
- **LUTS** determines how the displayed image is rendered, for example colour or grey scale.
- **Overlays** changes the colour of annotation of images and line drawing.
- **Ramps** tests the framestore or examines what a *lut* looks like.
- **Erase** lets you erase the image and overlay planes of the display screen.
- **Erase overlay** lets you erase the overlay plane only of the display. In fact, Semper itself allows you to erase the image plane, the overlay plane or both planes by specifying the appropriate options.

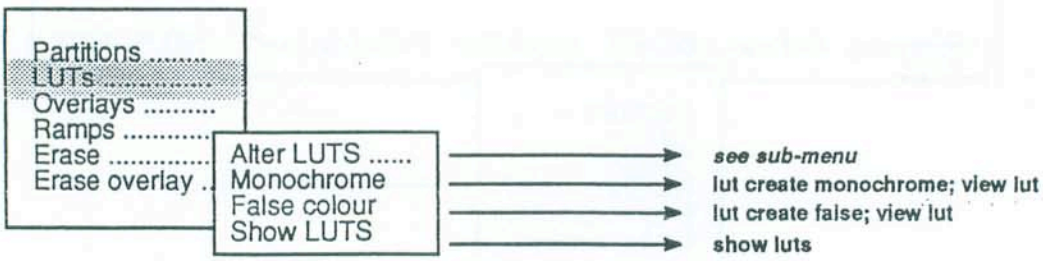
The Partitions sub-menu



Semper treats pictures stored in the display as if they were stored on a disk, keeping track of their size, dynamic range, scaling and so on. **Partitions** are used to sub-divide the display into logical storage areas. When a picture is displayed, a *display* picture is created, stored within the area defined by the partition. This menu allows you to define or determine the framestore areas associated with partitions 1 to 9. Many Semper installations allow you to have more partitions than this but, as with the picture selection menu, the limit of nine partitions is imposed to reduce the size of the menu.

If you define a partition using the mouse you need to mark the two opposite corners of the area you want to use for the partition, by clicking the left mouse button. A box is rubber-banded after you have defined the first point. The **Mark partition borde**r option marks the border of the selected partition in the current overlay colour.

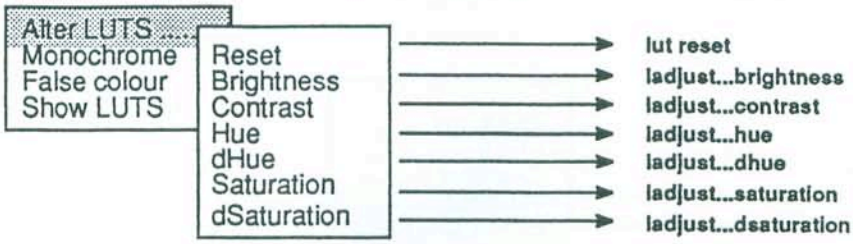
The Luts sub-menu



Semper understands three different types of look-up tables (*luts*):

- monochrome
- false colour
- true colour

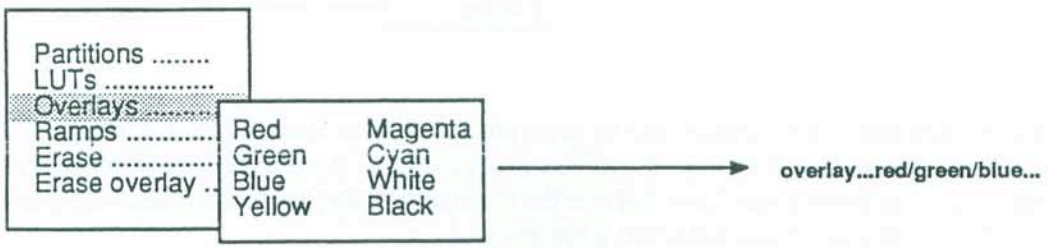
Whether or not your system supports all of these depends on the framestore you are using. If you are using the **See commands** option you will notice that you have to view the *lut* after creating it. This is because Semper allows you to have multiple look-up tables and you need to select the one to view at any one time. The **Alter LUTs** menu lets you alter various characteristics of a *lut* (for example, brightness or contrast) with the mouse. The **Show luts** option prints, on the terminal, details of the currently defined look-up tables.



The **Alter LUTs** sub-menu lets you change the contents of a look-up table interactively. You may use the mouse or keyboard, or both if you wish. The **Reset** option sets the *lut* to the default state, in case you want to return to the original look-up table. You will find that the **Hue** and **Saturation** options have no effect on a monochrome look-up table, for obvious reasons.

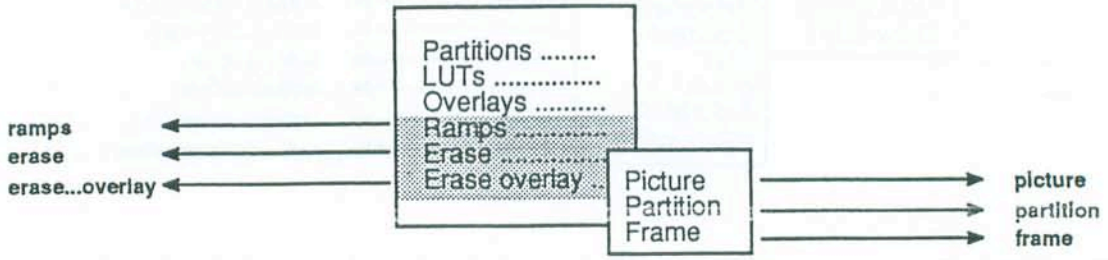
When you adjust the look-up tables, move the mouse vertically to change the selected parameter. Clicking the left hand mouse button, when you have finished adjusting the *lut*, returns you to normal *Tutor* operation. The Semper command, **ladjust**, which is used for these options is, in fact, more flexible than this and allows you to alter more than one characteristic at once. It also allows you to highlight particular bands (this option is used in setting the particle analysis limits).

The Overlays sub-menu



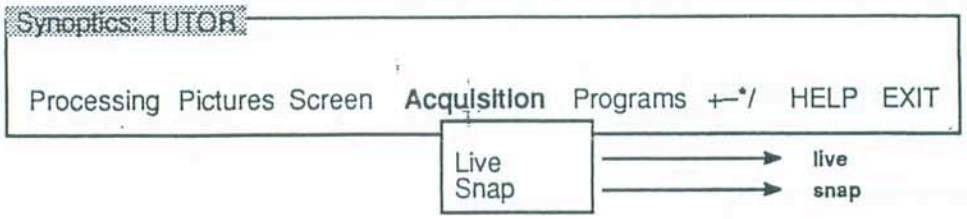
The **Overlays** sub-menu lets you choose the colour of the annotation used by Semper, that is the colour of the overlay plane. Note that some systems may not be able to support all the colours on this menu.

The Ramps, Erase and Erase overlay sub-menu



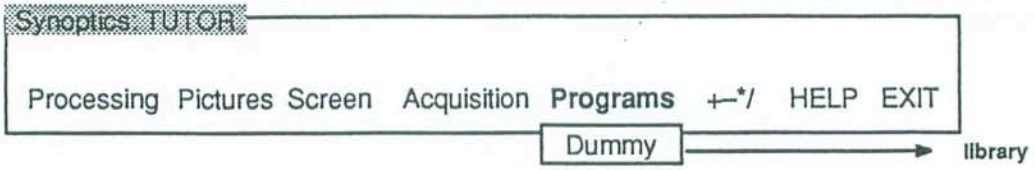
The **Ramps**, **Erase** and **Erase overlay** options all use a similar sub-menu to define which region of the screen you wish to modify. Semper lets you specify the region in terms of the area occupied by a picture, the area of a partition or the entire frame.

The Acquisition menu



The **Acquisition** menu allows you to grab images from a video camera provided that your framestore supports this facility. If you do have a camera, it provides a quick and easy way of capturing more images into *Tutor*. Refer to the manual *Installing Semper 6 Plus on a PC* for details of camera connections and capturing a live image.

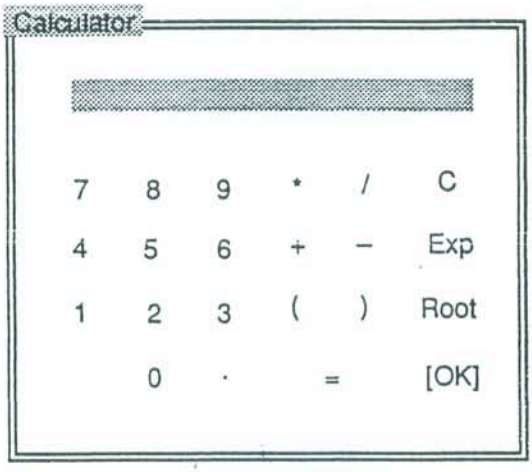
The Programs menu



The **Programs** menu allows you to run programs which are stored in a Semper library (or program) disk. This is where you can add your own library programs after, perhaps, logging commands and turning the file into a Semper program. For an example of Semper programming, see *Chapter 7, Adapting Tutor.*

The Calculator

The button marked +–*/ enables a pop-up calculator which looks like the one shown below. The calculator remains on the screen until you click on the button marked **[OK]**. The = and the **[OK]** buttons both evaluate the expression that you have entered. If you enter an invalid expression then *Tutor* will beep once. *Tutor* will also beep if you attempt to evaluate an empty expression, that is with nothing in the calculator display.



The HELP function

The button marked **HELP** prints help for the last menu that was displayed. This help is the same as that produced when the **Auto help** option is enabled on the **Options** panel.

The EXIT function

The button marked **EXIT** returns you to the operating system after confirming this action with you.

Chapter 6

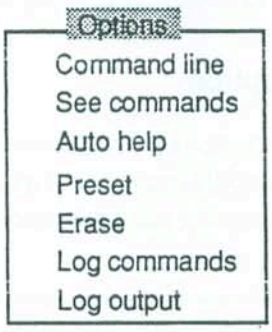
THE TUTOR LEARNING TOOL

Overview

Tutor is designed so that you can use it as a learning tool in the following stages:

- With the menus in conjunction with the **Auto help**. This allows you immediate access to the Semper image processing facilities without the need to learn much about Semper.
- With the **See commands** option enabled. This lets you see the actual Semper commands that *Tutor* uses and allows you to become familiar with Semper's syntax.
- With the **Log commands** and **Log output** options. These allow you to send information to a log file which can be printed. Sequences of Semper commands can be stored in this way and used as a basis for a Semper program.
- With the **Command line** option. This lets you enter your own commands and experiment with Semper itself.

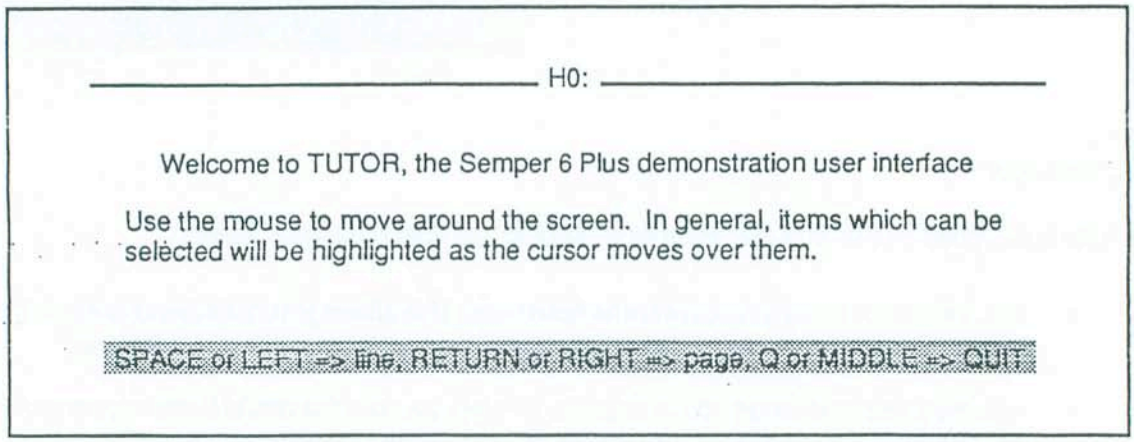
All of these options can be selected from the **Options** panel on the *Tutor* menu interface.



Using *Tutor* to learn about Semper means that you are able to progress from a structured, but limited, menu environment to using the Semper language itself – which provides flexibility, full image processing functionality and the framework for constructing your own specialised applications.

Using the help system

When you first experiment with *Tutor* you can enable the **Auto help** option on the **Options** panel. This provides you with help text which explains each *Tutor* menu. If the help text is too big for the scrolling area or text window you will see the *Tutor* paging prompt:

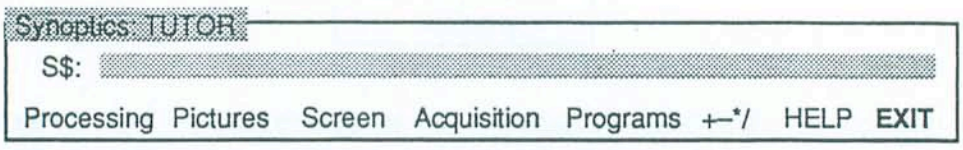


Use the mouse buttons or keyboard keys as indicated, to scroll the text. Note that while the paging prompt is displayed by *Tutor* the mouse does not perform its normal selection actions, that is, showing menus, selecting objects etc.

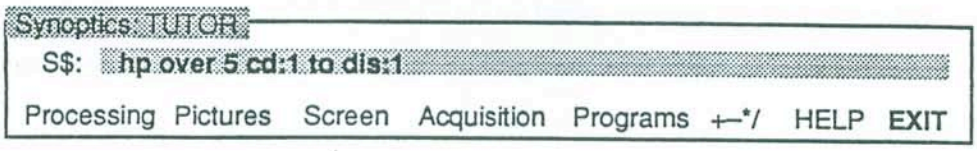
Semper 6 *Plus* help is also available, which is separate from *Tutor* help. Semper on-line help is extremely detailed and describes all the commands contained in the Semper language. It is quite feasible to have both sets of help available while *Tutor* is running: the mechanism for this is described in a later section called *Enabling the command line*.

Using the See commands option

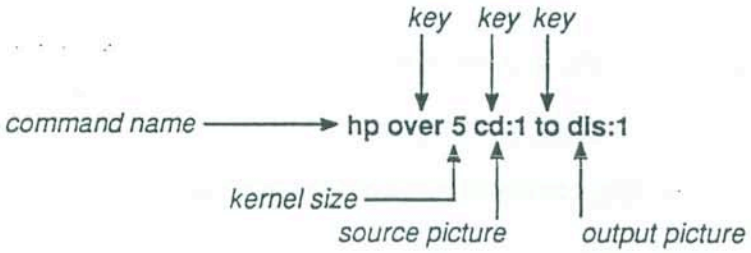
The **See commands** option on the **Options** panel allows you to learn Semper commands as you experiment with *Tutor*. When you select this option, a new field appears at the top of the main menu bar. Note that you cannot select this option if you have already enabled the **Log commands** option.



The Semper commands that correspond to selected menu items are shown in this field. For example, if you were to select the **High Pass** filter option (with a **5 x 5** kernel) from the **Spatial filtering** menu you would see the following command appear in the field:



The syntax of this Semper command can be broken down as:



Semper command syntax is explained in detail in the following manuals:

Beginners' User Guide
Advanced User Guide

These manuals are contained in the *Semper 6 Guide*.

Logging Information

Tutor allows you to log two types of information:

- Semper commands
- *Tutor* output (including help text and printed results from Semper commands)

When you enable the **Log commands** or **Log output** option, the information is sent by *Tutor* to a text file, which you can print out. Commands are logged in a file called *TUTORC.LOG*. Output is logged in the file *TUTORL.LOG*. When *Tutor* is logging information in this way, it writes to the end of the appropriate file so that previously stored information is not destroyed. We recommend that you delete the files *TUTORC.LOG* and *TUTORL.LOG* periodically so that they do not get too big and use too much space on disk.

Note that you cannot use the **See commands** and **Log commands** options at the same time. An example of command logging is given overleaf.

Example – logging commands

1. Turn off the **See commands** option on the **Options** menu.
2. Turn on the **Log commands** option on the **Options** menu.
3. Select the **Negate** option from the **Point-by-Point** menu of the **Processing** menu.
4. Exit *Tutor*.
5. You will see that you have two log files in your directory called *TUTORC.LOG* and *TUTORL.LOG*. *TUTORC.LOG* contains the command log. List this file on the screen, using the appropriate command:

   ```
   TYPE TUTORC.LOG      (PC and VAX installations)

   cat tutorc.log       (UNIX workstations)
   ```

6. You will see the following Semper commands listed on your terminal:

   ```
   display from cd:1 to dis:1
   negate cd:1 to dis:1
   ```

   These commands correspond to the menu selections that you made in *Tutor*.

7. Re-start *Tutor* by typing the command:

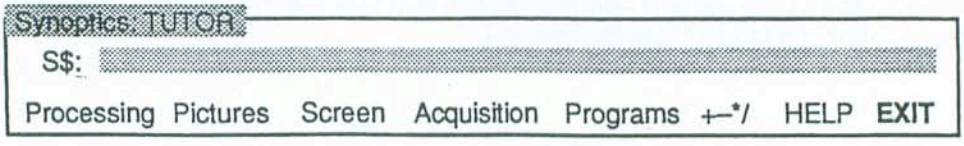
   ```
   semper /run=tutor
   ```

You can use the *Tutor* command logging feature to learn about the sequences of Semper commands used to perform an operation and also as a basis for Semper programming. For example, you can log a series of Semper commands and then incorporate the log file into an program file. For further details, refer to the following chapter, *Chapter 7, Adapting Tutor.*

The **Log output** option works on the same principle as the **Log commands** option. You can use this to obtain a printed copy of Semper output, for example, a list of particle parameters, pixel values or Semper help text.

Enabling the Command Line

To enable the Semper command line, turn on the **Command line** option on the **Options** menu. The command line is drawn at the top of the main menu bar, as shown below:



Before typing a command you need to move the cursor into the command line and click the left mouse button – this activates the command line.

The command line gives you access to all Semper commands. Some of these commands are not available from the *Tutor* menus. For example, try typing:

```
mask cd:1 to dis:1
```

This command produces a masked version of test picture 1 on the screen.

In an earlier section, *Using the help system*, mention was made of having both *Tutor* and Semper 6 *Plus* help available at the same time. You can do this by assigning the standard Semper help library, as follows:

```
assign help name 'semper'
```

If Semper is unable to find the help file it produces an error, otherwise you will see a message similar to the one below:

```
Device 5 assigned
```

With a help library assigned, you can obtain help on Semper commands. For example, try typing:

```
help mask
```

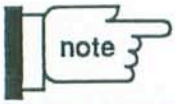
The following help information appears in a text window or in the Semper scrolling area:

```
-------------- Mask.command = mask --------------

You use MASK to reset to a constant all pixels inside or outside a given subregion –
for example, to eliminate unwanted features in an image, or to perform low or high
pass filtering of a Fourier transform.

Exx.   MASK
           resets all points more than two thirds of the way from the origin to the
           nearest picture edge to the mean picture value at that radius
       XWIRES CIRCLE; MASK INSIDE @CIRCLE VALUE 0
           resets to zero all points inside a circle indicated via 3 cursor points
       MASK OUTSIDE RADIUS 100 POSITION 20,30 WIDTH 10
           resets points more than 100 pixels from the point (20,30), fading
           smoothly over 10 or so pixels rather than abruptly
       XWIRES CURVE TO 51; MASK WITH 51
           resets all points outside an arbitrary region indicated via the cursor
[MORE VIA HELP /FULL...]
```

It is a useful learning tool to assign the Semper help in conjunction with the **See commands** option. By doing this, you can see which commands are used to perform a particular action and then read the Semper help for these commands.



The **Command line** option is not intended for very long command lines. If you enter a continued command line, with a plus sign (+) on the end, Semper will report an error because your command will be confused with *Tutor's* commands. If you need to enter long commands you can exit to the Semper command line using the following command:

```
uif exit
```

(To leave Semper, type **stop**).

Chapter 7

ADAPTING TUTOR

Overview

This chapter describes how to add a new image processing function to the *Tutor* menus. It also takes a look at a section of the program used to generate the *Tutor* interface itself. Before you start to adapt the user interface it is a good idea to experiment with the Semper command language so that you are familiar with its conventions. You may, for example, like to activate the **Command line** option on the **Options** panel. For further details about using *Tutor* as an introduction to Semper, refer to *Chapter 6, The Tutor Learning Tool.*

Analysing Tutor

Tutor itself is an example of quite a complicated user interface. It is made up of the user interface elements; panels, menus, cells and textfields. Figure 7-1 illustrates the elements that comprise the interface:

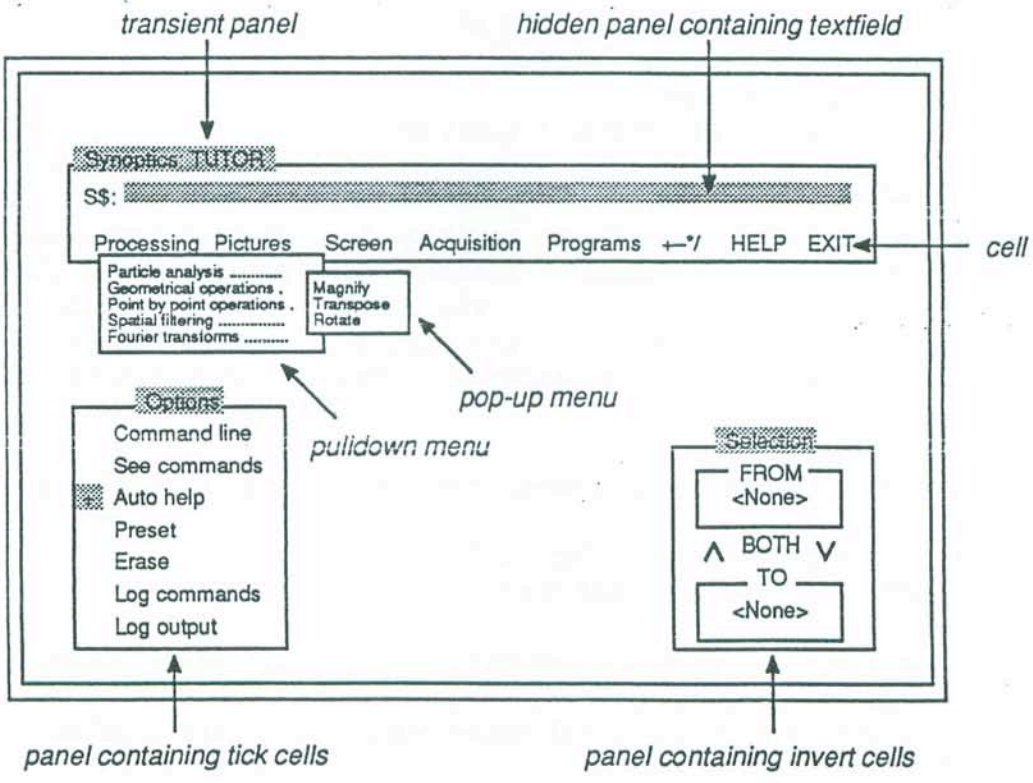


Figure 7-1. Components of the Tutor User Interface

An example of a section of the *Tutor* program is given below. This extract defines part of the first level of menus on the main menu bar at the top of the *Tutor* screen. Note that selecting a cell on the menu bar causes a call to one of a number of library programs which are not listed here.

```
! build$menus:
!
! Create the demonstration user interface from scratch
!
build$menus()
local r, c
!
justification top left
!
! Set variables for no menus showing (used by library programs). Also
! no mandatory panel visible.
!
$lv = 0
$tm = -1
m99 = -1
unset $mn
!
! Create a dropdown line
!
panel create name 'Synoptics: TUTOR ' size 80,4 position 0,0 transient +
   foreground $c0 background $c1
p00 = pno; invert = yes
cell create text 'Processing ' position  1,2 changes 'mu=3;lib bar'
cell create text ' Pictures  ' position 13,2 changes 'mu=2;lib bar'
cell create text '  Screen   ' position 25,2 changes 'mu=1;lib bar'
cell create text 'Acquisition' position 37,2 changes 'mu=4;lib bar'
cell create text ' Programs'   position 49,2 changes 'mu=5;lib bar'
cell create text '+-*/'        position 60,2 changes 'lib cal'
cell create text ' HELP '      position 65,2 changes 'lib hfn'
cell create text ' EXIT '  position 72,2 changes 'lib bye' fore $c2 back $c3
unset invert
!
! Now the menus which drop down from the bar, the first level menus.
!
menu create pulldown position  1,3
m3 = eno; invert = yes; add = m3
!
cell create text 'Particle analysis ..........' row 1 column 1 +
   changes 'mu=18;lib bar2'
cell create text 'Geometrical operations ......' row 2 column 1 +
   changes 'mu=16;lib bar2'
cell create text 'Point by point operations ...' row 3 column 1 +
   changes 'mu=17;lib bar2'
```

```
cell create text 'Spatial filtering ..........' row 4 column 1 +
   changes 'mu=15;lib bar2'
cell create text 'Fourier transforms..........' row 5 column 1 +
   changes 'mu=19;lib bar2'
unset invert, add
```

Since *Tutor* is defined as a text file (*TUTOR.SPL* in the *SEMPER6P* directory) you can list it to a printer and examine the entire source code at your leisure. There are also many useful library programs in *Tutor* which you may use when creating your own interface.

Adapting Tutor to your requirements

Once you have gained confidence with using the Semper image processing language, you may like to add some functions of your own to the Tutor menus. The **Programs** menu is free for you to use or, alternatively, you might decide that your application fits in with the functions on an existing menu, for example, **Processing.**

There are three defined stages to adding a function to the *Tutor* menus:

1. Write a Semper program defining the new image processing operation.

2. Add the function to an existing *Tutor* menu or create a new menu.

3. Delete two specified *Tutor* files so that *Tutor* can regenerate its interface to include the new function.

The first stage consists of defining the operation and writing the Semper statements to perform it. The operation must be written as a Semper 6 *Plus* library program, which is added to the library program file, *TUTOR.SPL*. (If you are unfamiliar with Semper programming, you may like to refer to *Tutorial 6, Programming with Semper* in the *Beginners' User Guide).* If you have a set sequence of image processing functions in mind and they all appear on the *Tutor* menus, you could use the **Log commands** option to record the actual Semper commands in the file *TUTORC.LOG*. You could then edit the appropriate parts of the file into a suitable Semper program, allowing for a user entering different picture numbers, as is described later.

The second stage consists of adding a new entry to an existing menu, or even creating a new menu. This also requires that you edit the file *TUTOR.SPL,* but in this case you must edit an existing function called *build$menus.*

The third stage involves deleting the files *TUTOR.PLB* and *TUTOR.UIF* which contain the compiled library programs and the saved user interface. This forces *Tutor* to build new files when it is next run.

These stages are described in detail in the following sections.

To avoid destroying the original *Tutor* files when adapting *Tutor*, we recommend that you make backup copies of the following files:

- *TUTOR.SPL*
- *TUTOR.PLB*
- *TUTOR.UIF*

These files may be required by another *Tutor* user.

1. Defining the image processing operation

In this example, we shall add a thresholding function to the **Point by Point operations** menu of the *Tutor* interface. This threshold converts the source picture into a binary image, with the dividing line falling at the middle intensity level. The first step is to write the Semper 6 *Plus* library program which performs this function, by editing the file *TUTOR.SPL*.

The program performs three main tasks:

1. Checks the source and destination pictures are valid
2. Finds the middle intensity level of the source picture (**survey** command)
3. Thresholds using a point by point operation (**calculate** command)

The *Tutor* interface uses the variables *$c8* and *$c9* to hold the picture numbers of the source and destination pictures. If either of the variables are set to –1, this indicates that no picture has been selected. (You can find out what other variables *Tutor* uses if you type **show variables** with the **Command line** option enabled). Therefore, the first stage of the library program is to check for invalid source or destination pictures:

```
if (($c8 = -1) | ($c9 = -1)) return
```

For the moment we shall assume that the program returns without giving any error message. The next stage is to establish the middle intensity level of the source picture. The **survey** command stores the range of values in the picture in the variables *min* and *max*, hence the midpoint:

```
survey $c8 noverify
midpoint = (max + min) / 2.0
```

You specify the **noverify** option of the **survey** command so that the output is not sent to the screen.

Finally, the image is thresholded using the **calculate** command:

```
calculate (:$c8 > midpoint) to $c9
```

A colon (:) is required before a picture number in the **calculate** command so that Semper can distinguish between a picture and a number that is part of an arithmetic expression.

The output from this command will either be 0 (for intensities less than the midpoint) or 1. Note that the **calculate** command is capable of handling more complex functions than this and you could, if you wanted, adapt the threshold command so that ranges of the picture are thresholded. Putting all these elements together, you end up with the final library routine.

Leave *Tutor* and add the following lines to the file *TUTOR.SPL*, using a text editor with which you are familiar. You can insert the new function at the end of any existing function in the file. Note that in a Semper program, an exclamation mark means that the text following is a comment and the **local** command declares that a variable, in this case *midpoint*, is local to the routine.

```
! Threshold:
!
! Threshold an image. This produces a binary image of the source.
!
threshold()
local midpoint; ! Middle intensity point of source picture
!
! Check that the source and destination are valid. The variables
! $c8 and $c9 give these.
!
if (($c8 = -1) | ($c9 = -1)) return
!
! Survey image to get mid point of intensity range but do not
! output the results to the display (only set the variables min
! and max).
!
survey $c8 noverify
midpoint = (max + min)/2
!
! Now threshold the image using the calculate command
!
calculate (:$c8 > midpoint) to $c9
return
end
```

You have now created a new *TUTOR.SPL* file.

2. Adding the function to the menus

The next step consists of adding a new menu item to the *Tutor* menus. This is achieved using a **cell** statement:

```
cell create text 'Threshold ' row 4 column 1 changes 'lib threshold'
```

The **cell** command creates a new interface element, defined by the **create** option. The cell is called 'Threshold' so that this name appears on the menu at the specified position (row 4, column 1). The **changes** keyword defines the operation that is to be performed when the cell is selected (by mouse click), in this case running the *threshold* library program.

Add the **cell** statement to the file *TUTOR.SPL* using a text editor, as follows. Find the function called *build$menus* in *TUTOR.SPL*. This function creates the *Tutor* user interface from scratch. Next find the definition of the **Point by Point** pop-up menu in *build$menus*, which looks like the one shown overleaf:

```
!
menu create popup position 27,6
m17 = eno; invert = yes; add = m17
!
cell create text 'Negate  ' row 1 column 1 changes 'lib a30'
cell create text 'Rescale  ' row 2 column 1 changes 'lib a31'
cell create text 'Calculate ...' row 3 column 1 changes 'lib a44'
unset invert, add
!
```

To add the new function to the menu you need to add just one line, as shown below:

```
!
menu create popup position 27,6
m17 = eno; invert = yes; add = m17
!
cell create text 'Negate  ' row 1 column 1 changes 'lib a30'
cell create text 'Rescale  ' row 2 column 1 changes 'lib a31'
cell create text 'Calculate ...' row 3 column 1 changes 'lib a44'
cell create text 'Threshold ' row 4 column 1 changes 'lib threshold'
unset invert, add
!
```

3. Deleting Tutor files

You have now made all the necessary changes to the source code of the software. You now need to delete the old copy of the program library, *TUTOR.PLB*, (*Tutor* recreates it when it starts running) and also remove the saved user interface file, *TUTOR.UIF* so that *Tutor* can build a new interface.

The next time that you start *Tutor* the threshold function is available for use from the **Processing** menu. Note that you will need to wait at start-up whilst *Tutor* adds its library programs to a new program library and recreates its user interface. The new **Threshold** option is illustrated below:

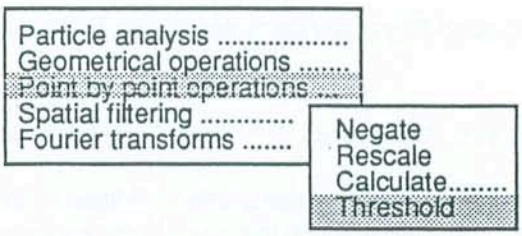


Figure 7-2. The Threshold Option

To use the **Threshold** option, first create a new disk picture, by typing:

```
create cd:5 size 512 byte
```

This is necessary as the **calculate** command requires a defined destination and a display picture may be unsuitable for its output (for example, the display may be undersampled). Then try thresholding using the test picture of the *Mona Lisa* as a source and disk picture 5 as a destination. As a final step, display picture 5. Figure 7-3 below illustrates the use of the thresholding function on the test image.

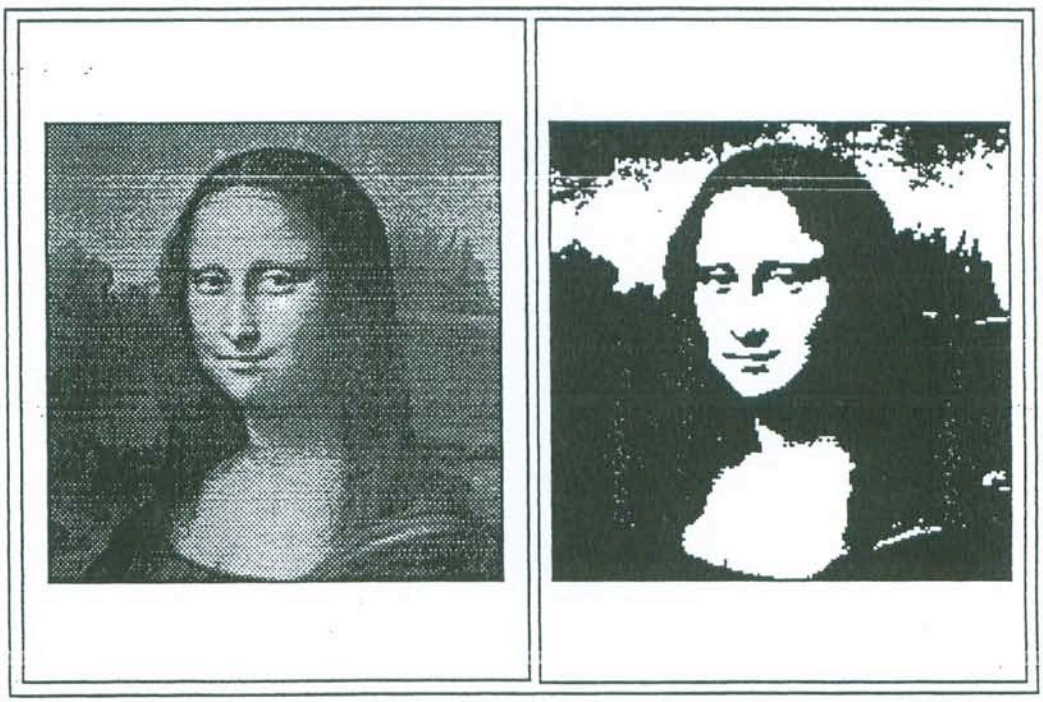


Figure 7-3. Original and Thresholded Image

What next?

After you have experimented with *Tutor* you may now like to add more of your own functions to it. Alternatively, you may like to create your own interface for a specific application.

Some useful documents to read are:

- *User Interface Guide*
- *Beginners' User Guide*

Both of these documents are contained in the *Semper 6 Guide*

Appendix A

INSTALLING TUTOR

Overview

This appendix describes the changes that you may need to make to non-standard framestore configurations before you start *Tutor* and details some of *Tutor's* software and hardware requirements.

Non-standard installations

Normally you should not have to make changes to your files as *Tutor* comes configured to suit the default settings for your framestore.

Depending on how your system is set up, you may need to edit the *Tutor* file called *TUTOR.RUN*, using a text editor with which you are familiar. This will only be necessary if you have your framestore in a non-standard configuration, in which case you will need to change how the framestore is assigned. The framestore must be assigned before Semper can access it and the Semper command to do this, called **assign**, takes several options. The Semper help for *framestore* will give you details of what these options are and what they do. For this information, type the following command:

```
help framestore
```

when running the Semper 6 *Plus* command language or check your installation notes. You can also type the command:

```
help assign
```

for details of the **assign** command.

Software requirements

Several files, over and above the standard Semper files, are required to run the *Tutor* user interface. These are:

- TUTOR.RUN — The start-up file
- TUTOR.SPL — The library programs
- TUTOR.HLB — The help file

These files are installed in the standard Semper directory of your machine on installation of Semper 6 *Plus*.

Memory requirements

Note that *Tutor* requires at least 1.5Mb of disk space as well as that required for the standard Semper installation. This is because *Tutor* creates the following extra files the first time that you run the software:

- TUTOR.UIF — The (saved) user interface (35Kb)
- TUTOR.PLB — The compiled library program (120Kb)
- TUTOR.DSK — The picture (image) disk (1Mb by default)

Other requirements

The only extra equipment that you need to run *Tutor* is a mouse with at least two buttons on it (if you don't have one already). Refer to your manufacturer's installation instructions for how to set up the mouse.

Appendix B

TROUBLESHOOTING

Overview

This appendix describes the type of warning and error messages that you may see when running *Tutor* and refers you to sources of help. It also lists and explains the start-up error messages that you may see when you run *Tutor*.

Tutor error and warning messages

Tutor shows the standard Semper error messages when most error conditions occur. For example, you may see an error message like the following:

```
?5 Bad size for picture 2001
```

displayed in a text window or scrolling area. These error messages are explained and possible solutions are given in *Appendix E, Error Messages* of the following manual:

Semper 6 Command Reference Part 2

Alternatively, you could assign the Semper on-line help (see *Chapter 6, The Tutor Learning Tool* for detaiis) and type **help** followed by a question mark and the error number, for example:

```
help ?5
```

Semper on-line help provides a full description of each error message.

In some circumstances, however, *Tutor* traps the error and displays an alternative form of error message. When *Tutor* does this, it shows an information panel, like the one shown in Figure B-1 overleaf. After displaying the information panel, *Tutor* positions the cursor over the **OK** button and waits for you to click on it to confirm that you have read and understood the message. While an information panel is showing, *Tutor* does not allow you to pop up any menus or select any items.

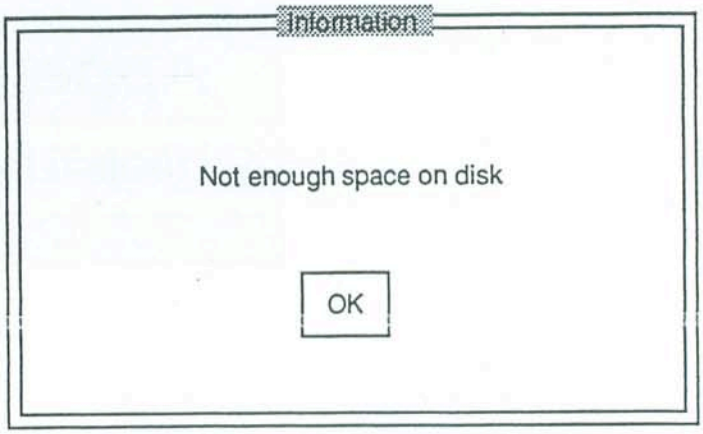


Figure B-1. An Information Panel

Tutor also displays warning messages as well as error messages. In some circumstances, to confirm an action, *Tutor* displays a dialogue panel like the one shown in Figure B-2. This operates in a similar way to the information panel, except that there are two buttons on the panel. One button is marked **YES** and the other **NO**. The cursor is positioned over the **NO** button so that if you are sure that you wish to proceed, you must move the cursor over to the **YES** button and click on it. Alternatively, if you don't want to continue with the operation, click on the **NO** button and the operation will be abandoned.

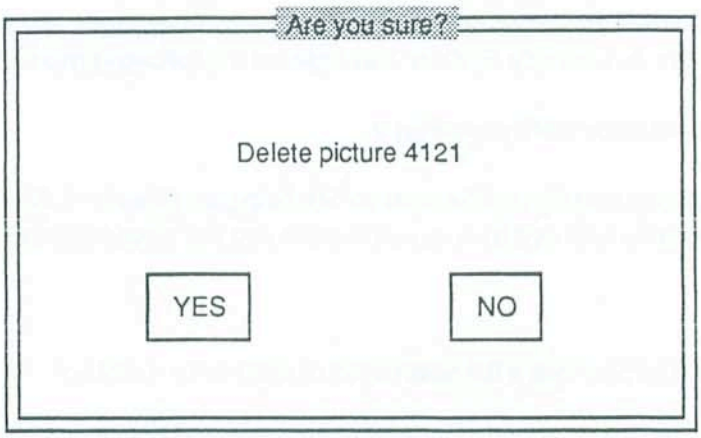


Figure B-2. A Dialogue Panel

Tutor start-up messages

The following error messages are displayed if *Tutor* is unable to perform one of its start-up functions:

Cannot assign the display. Are you running your PC too fast? Are the right startup files available for the display board?

This error message means that Semper could not initialise the framestore, that is used to display images. Is the **assign** command in the *TUTOR.RUN* file correct? Check your installation notes for other likely causes of failure. Some framestores have a configuration file which controls how the framestore behaves – is this file present? *Tutor* will stop running after printing this message and return you to the operating system.

Sorry, cannot generate the program file to run this program. Check that there is at least 120k free on your disk.

This error message means that Semper could not find enough space on your disk. Semper 6 *Plus* stores its programs on a library disk. This is a disk file which stores the programs so that they may be accessed quickly and efficiently. When *Tutor* is started for the first time it has to convert the file *TUTOR.SPL* into this format. Is the file *TUTOR.SPL* in the current directory? Is there enough space on your disk? Note that *Tutor* will stop running after printing this message.

Sorry, cannot find help file TUTOR.HLB.

This is a warning message that means that the *Tutor* help file was not found. Ensure that the file *TUTOR.HLB* is in the Semper directory. After displaying this message, *Tutor* will continue but no help will be available.

Sorry, cannot create picture disk for Semper 6 Plus operations. Check that you have at least 1 Mb free on your disk.

This error message means that the picture file *TUTOR.DSK* could not be created. Check that there is sufficient free space on your disk. *Tutor* will stop running after printing this message.

Sorry, could not read in test picture.

This is a warning message. When *Tutor* runs for the first time, it creates an empty picture file. After this file has been created *Tutor* attempts to read in a standard test picture so that you have at least one image with which to experiment.

Sorry, cannot assign file for command logging, this facility will not be available in this session.

When the command logging option is enabled, *Tutor* writes the commands that it executes into a file called *TUTORC.LOG*. In the start-up process *Tutor* was unable to access this file. This may be because the file is write-protected or, on a multi-user system, because your access privileges are incorrect. Since *Tutor* was unable to access this file, no command logging is allowed in your current session.

Sorry, cannot assign file for output logging, this facility will not be available in this session.

When the output logging option is enabled, *Tutor* writes the output from commands that it executes into a file called *TUTORL.LOG*. In the start-up process, *Tutor* was unable to access this file. This may be because the file is write-protected or, on a multi-user system, because your access privileges are incorrect. Since *Tutor* was unable to access this file, no output logging is allowed in your current session.